UNIVERSITY OF CALIFORNIA,
IRVINE

Robust Navigation via Measurement Integrity Monitoring and Learning Methods

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Mechanical and Aerospace Engineering

by

Changwei Chen

Dissertation Committee:
Professor Solmaz Kia, Chair
Professor Andrei Shkel
Professor Athanasios Sideris

2023

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

LIST OF ALGORITHMS

ACKNOWLEDGMENTS

This dissertation work is the output of the effort and support of several people to whom I am extremely grateful. First and foremost, I would like to thank my Ph.D. advisor, Professor Solmaz Kia, for the support and guidance on my work during this journey. Her continuous support has allowed me to freely explore the research topics. Her vision and motivation have inspired me to push my limit and become a better version of myself every day.

I would like to thank my Ph.D. committee members Professor Andrei Shkel and Professor Athanasios Sideris for sparing time out of their busy schedule to serve on the committee and for their helpful advice throughout both my Ph.D. Qualifying Exam and Final Defense.

I would like to thank the generous funding provided by National Institute of Standards and Technology (NIST) for my work. This work has been supported by NIST under Grant 70NANB17H192 and DOD Phase I STTR W31P4Q-20-C-0007.

I would like to thank all my friends and colleagues in the KCS lab, Amir-Salar Esteki, Donipolo Ghimire, Joan Vendrell, Minwon Seo, Mohammadreza Rostami, and Robert Palfini for their discussion and help in this dissertation work. It is a great honor to work with them and build the KCS together for the past three years.

I would also like to thank Professor Shkel again and his UCI MicroSystems Laboratory, specifically, Dr. Chi-Shih Jao, Dr. Danmeng Wang and Eudald Rafart for collaborating on the NIST project. I am more than grateful to be able to join their weekly meetings and receive guidance. I look forward to working with them again in the future.

I would also like to thank my parents for their love, patience, and sacrifice during this journey. They were always there for me to support whenever I was facing challenges. I gain stronger strength and determination every time I chat with them.

Last but not least, I would like to extend my heartfelt gratitude to my girlfriend, Ruochen, whose unwavering support and love have been a constant source of inspiration throughout this challenging academic journey. It would be impossible to finish my dissertation work alone without her companion. Her patience, understanding, thoughtfulness, and encouragement have been instrumental in sustaining my focus and motivation during my research. “Among the crowd with nowhere to stay, your appearance becomes an island and becomes concrete. I love your sharp scars and your mature innocence. Thank you for being so wonderful and dazzling, and becoming the star in my ordinary days.” I am truly fortunate to have her by my side, and I express my deepest appreciation for her invaluable contributions to both my personal and academic life.

VITA

Changwei Chen

EDUCATION

Doctor of Philosophy in Mechanical and Aerospace Engineering **2023**
University of California, Irvine *Irvine, CA*

Master of Science in Mechanical and Aerospace Engineering **2020**
University of California, Irvine *Irvine, CA*

Bachelor of Science in Automotive Engineering **2018**
Wuhan University of Technology *Wuhan, Hubei*

RESEARCH EXPERIENCE

Graduate Student Researcher **2020–2023**
University of California, Irvine *Irvine, California*

TEACHING EXPERIENCE

Teaching Assistant **2021-2023**
University of California, Irvine *Irvine, CA*

JOURNAL PUBLICATIONS

[J6] **Changwei Chen** and Solmaz S. Kia, "Self-localizing On-demand Portable Wireless Beacons for Coverage Enhancement of RF Beacon-based Indoor Localization Systems", *IEEE Journal of Indoor and Seamless Positioning and Navigation*, early access, 2023.

[J5] **Changwei Chen** and Solmaz S. Kia, "OptiTrack-aided Supervised Learning for Neural Network Based Ultra-wideband Ranging Bias Correction", *IEEE Sensors Journal*, under review, 2023.

[J4] Chi-Shih Jao, Danmeng Wang, **Changwei Chen**, Eudald Sangenis, Joe Grasso, Solmaz S. Kia, Andrei M. Shkel, "Augmented UWB-ZUPT-SLAM Utilizing Multi-Sensor Fusion", *IEEE Journal of Indoor and Seamless Positioning and Navigation*, early access, 2023.

[J3] Chi-Shih Jao, Ali A. Abdallah, **Changwei Chen**, Minwon Seo, Solmaz S. Kia, Zaher M. Kassas and Andrei M. Shkel, "PINDOC: Pedestrian indoor navigation system integrating deterministic, opportunistic, and cooperative functionalities", *IEEE Sensors Journal*, 22(14), 2022.

[J2] **Changwei Chen** and Solmaz S. Kia, "Cooperative Localization Using Learning-Based Constrained Optimization", *IEEE Robotics and Automation Letters*, 7(3), 2022.

[J1] **Changwei Chen**, Chi-Shih Jao, Andrei M. Shkel and Solmaz S. Kia, "UWB sensor placement for foot-to-foot ranging in dual-foot-mounted ZUPT-aided INS", *IEEE Sensors Letters*, 6(2), 2022.

CONFERENCE PUBLICATIONS

[C2] Chi-Shih Jao, Ali A. Abdallah, **Changwei Chen**, Minwon Seo, Solmaz S. Kia, Zaher M. Kassas and Andrei M. Shkel (2022), "Sub-Meter Accurate Pedestrian Indoor Navigation System with Dual ZUPT-Aided INS, Machine Learning-Aided LTE, and UWB Signals", *Proceedings of the 35th International Technical Meeting of the Satellite Division of The Institute of Navigation*, (Denver, CO), pp. 1108-1126. 2022.

[C1] **Changwei Chen** and Solmaz S. Kia (2021), "A Renyi divergence based approach to fault detection and exclusion for tightly coupled GNSS/INS system", *in Proceedings of the International Technical Meeting of The Institute of Navigation*, (Online), pp. 674-687. 2021.

RESEARCH PRESENTATION IN PEER-REVIEWED CONFERENCES

"Cooperative Localization Using Learning-Based Constrained Optimization", *The 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IROS 2022), Kyoto, Japan.

AWARDS AND HONORS

Paul and Beverly Holmes Endowed Fellowship 2021 - 2022

SOFTWARE

C++, Python, MATLAB, ROS2, OpenCV, TensorFlow

ABSTRACT OF THE DISSERTATION

Robust Navigation via Measurement Integrity Monitoring and Learning Methods

By

Changwei Chen

Doctor of Philosophy in Mechanical and Aerospace Engineering

University of California, Irvine, 2023

Professor Solmaz Kia, Chair

This dissertation focuses on the development of robust and reliable navigation algorithms for mobile autonomous systems. This work is applicable to inertial navigation systems (INS) aided by external measurements from the Global Positioning System (GPS) and/or ultra-wideband (UWB) sensors in a cooperative manner. To improve the robustness and continuity of precise navigation systems, this work looks at various factors, such as identifying and excluding erroneous sensor measurements, effective use of machine learning tools for navigation systems, and innovative navigation frameworks and infrastructures to enhance external signal access. On the integrity monitoring front, we develop an information-theoretic approach for fault detection and exclusion (FDE). This work also includes a statistical inference method to optimally estimate the probabilistic fault detection threshold. For robust navigation framework and infrastructure design, we propose innovative solutions based on collaborative navigation approaches such as UWB-based foot-to-foot ranging for dual-mounted INS for pedestrian localization, cooperative navigation for multi-agent systems, and self-localizing on-demand portable wireless beacons for coverage enhancement of RF beacon-based indoor localization systems. To induce robustness in these designs, we incorporate multiple measures. In UWB-based foot-to-foot ranging, we analyze the UWB ranigng measurement acuraccy and how proper relative placement of the UWB sensors can lead to measurement accuracy enhancement. To improve ranging accuracy of the UWB measurements, we also

study how to remove the bias in different measurement models, i.e., line-of-sight (LoS) and non-line-of-sight (NLoS), using an artificial neural network (ANN) approach. Our contributions span across the architecture design of ANNs, identifying the informative features of the UWB signal to train the ANNs, and employing an OptiTrack motion capture camera system to collect a diverse set of training data in various relative poses between the sensors. Our work also explores the use of ANNs to improve the computational complexity of loosely coupled cooperative navigation solutions. Our work uses a supervised machine learning approach to learn the solution of computationally expensive optimization processes of loosely coupled cooperative navigation solutions from off-line data. The result is a set of light-weight ANNs that can predict the solutions online in a computationally efficient manner. The innovation in our work is proposing ANN architectures that output feasible solutions, i.e., solutions that are guaranteed to satisfy the constraints of the optimization processes. Our last contribution towards robust navigation solutions is to extend the use of our loosely cooperative navigation method to design a framework to deploy portable on-demand beacons to extend the coverage/signal access of the RF beacon-based localization systems. Our solution addresses the challenge of how to localize these deployed beacons in an on-line and decentralized manner. The proposed solution is a robust deployment method in the sense that if a portable beacon is moved for any reason, it can automatically re-localize itself in the decentralized manner. Simulations and experimental studies demonstrate the results of this thesis work.

Chapter 1

Introduction

1.1 Motivation

Navigation has been an ancient art since the beginning of human history and has become increasingly essential in the tasks of controlling and monitoring the movement of a target, e.g., ground vehicle, pedestrian, aircraft, submarine, to reach the goal in both commercial and military applications [3, 4, 5]. The core of navigation, however, has never changed over millenniums and has been continuously enhanced, which is the location. Nowadays, almost all navigational techniques involve locating the navigator's position compared to known locations, which requires measurements taken from multiple sensors, e.g., global positioning system (GPS) receivers, ultra-wideband (UWB) transceivers, inertial measurement units (IMUs), and altimeters. In other words, the target location is inferred or decoded from the sensor measurements. However, measurements are always contaminated by noise or occasionally corrupted by unexpected and unjustifiable values, i.e., faults or jammings, but only reliable and accurate measurements contribute to the estimation of precise locations. Although measurement noise and regular bias can be modeled and thus compensated

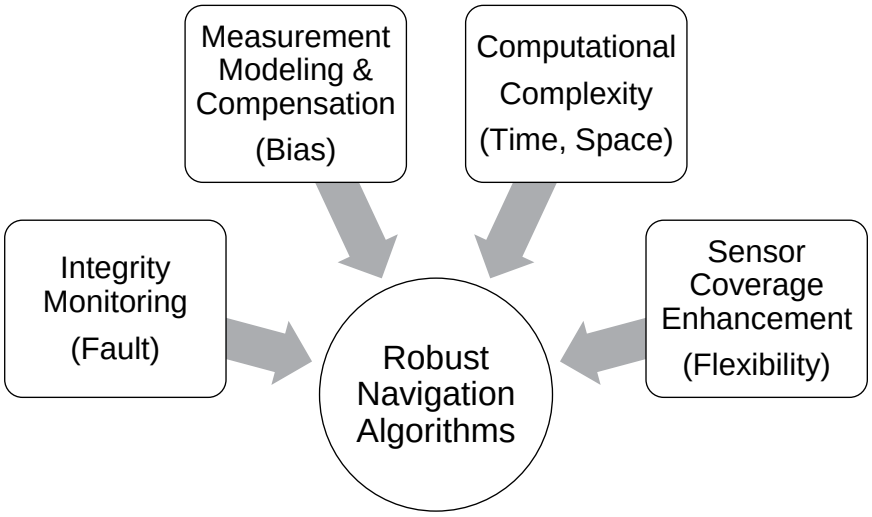


Figure 1.1: Robustness enhancement in navigation algorithms

systematically to mitigate their effects in some cases [6, 7], using incorrect measurements degrades the localization accuracy [8] and can even lead to catastrophic results such as air crash. Therefore, given the fact that measurement modeling, processing and integrity monitoring all play important roles in location awareness applications, it is essential and critical to develop efficient fault-tolerant mechanisms that provide precise measurements. For example, GPS obtains satellite pseudorange measurements with clock bias mitigation methods to triangulate the target position and the receiver autonomous integrity monitoring (RAIM) algorithm is implemented to detect and exclude faulty GPS measurements [9]. Thus, the research interest of this dissertation work is to develop and improve such mechanisms to enhance the effectiveness, efficiency and robustness of navigation.

We focus mainly on four main components for robust navigation, which are integrity monitoring, measurement modeling and compensation, computational complexity, and sensor coverage enhancement, as shown in Fig. 1.1. In what follows, we give a short overview of these components and the desired specifications for them for our application of interest.

Integrity Monitoring: Integrity monitoring of navigation systems is the ability to provide timely warning to users when the system should not be used for navigation [10]. To imple-

ment integrity monitoring and ensure the navigation continuity, the navigation systems must detect faulty measurements before resulting in unacceptable performance. Consequently, integrity monitoring of multi-sensor navigation systems has been pursued and enhanced for decades, contributing to a rapid growth in the research on the subject of Fault Detection and Isolation (FDI) or Fault Detection and Exclusion (FDE). In general, integrity monitoring can be applied internally to any navigation system that provides redundant measurements or externally by comparing the outputs of multiple navigation systems [11]. And the key challenge has been developing a mechanism with adaptive criteria functions to increase the fault detection rate while decrease the miss detection rate.

Measurement Modeling and Compensation: Given that measurement is essential to navigation systems, a correct mathematical model for measurement is of necessity. Besides the fault which can be removed via integrity monitoring methods, the bias is another error resource that degrades the quality of sensor measurements. For example, for radio frequency (RF) signals, such as GPS signals, measurement bias is mainly due to multipath [9]. The main effort has been put into modeling and inference the bias and ensure the accuracy in navigation solutions.

Computational Complexity: Algorithm robustness calls for less computational complexity in the sense that the longer the execution time, the more likely errors will occur in the systems. Also, for robust navigation algorithms in real-life applications, we need to consider rigorously the computational complexity to make the algorithm run in real-time. Fast and accurate response with respect to the user's request is in high demand nowadays. Meanwhile, given that the learning methods are adopted as well, it is crucial to have the safety-critical guarantee to execute the navigation algorithms to enhance the robustness.

Sensor Coverage Enhancement: Lastly, to ensure robust navigation, sufficient connections to the sensor network are critical. Deploying additional sensors as beacons enhances the robustness of the navigation system. On the contrary, localization using fewer beacons

requires more robust navigation algorithms. If the hard requirements or assumptions of the navigation algorithms can be relaxed or even removed, they become more general and more widely applicable. In other words, the navigation algorithms are more robust against different scenarios than they used to be.

In the remainder of this chapter, an overview and literature review of integrity monitoring, measurement modeling and compensation and bias mitigation methods, computational complexity, and sensor coverage enhancement are presented. Then, the objective of this research is presented in more details given the context provided by our reviews.

1.2 Literature survey

In this section, we review the state-of-the-art integrity monitoring techniques in the literature. Then we move to the review of measurement bias modeling and compensation methods. We finalized the section with a review of approaches to reduce computational complexity and sensor beacon localization.

1.2.1 Integrity Monitoring

There are generally two main categories of the integrity monitoring methods in the literature, model-based and model-free. Considering there is also a massive quantity of work on integrity monitoring in the data science and computer science field, in this dissertation work, we only focus on that in the navigation domain, specifically GPS and inertial navigation system (INS) applications.

The leading workhorse of model-based FDE in the literature has been residual-based detectors, where the residual is the difference between the actual output (measurement) acquired

by the sensors and the estimated output. The theoretical structure for residual-based detectors is statistical detection theory. These techniques calculate certain statistics of the residual and compare the statistics with a predefined threshold based on false alarm probability as an indicator to determine which hypothesis is true given current observations, i.e., non-faulty or faulty. In the GPS applications, the measurements from the satellites (pseudorange, Doppler) can be obscured or degraded due to different phenomena such as the multipath and ionosphere interference. The conventional residual-based methods for GPS integrity monitoring such as RAIM [9], Pseudorange Comparison Methods [12], are widely adopted. However, these methods have limits; for example, to perform fault detection, at least 5 satellites should be in view, and to perform FDE, the number increases to 6. Furthermore, they can only deal with single-fault cases, i.e., the detection of one fault at one time since given only the information that the pre-designed threshold is exceeded, one cannot distinguish between the single-fault case and the multiple-fault case. Also, these methods are batch processing methods that have strong dependency on history, so they have limited capability to deal with incipient faults, i.e., faults with small magnitude and increasing slowly.

The performance of the traditional FDE methods has been enhanced by combining multiple navigation systems which are not limited in the same category. Advanced RAIM (ARAIM) [13] is proposed to use redundant pseudorange measurements obtained from multi-constellation and multi-frequency global navigation satellite system (GNSS). [14] investigates combining filtering with dynamical models to enhance the performance of a chi-square detector and leverage the frequency content of the residual signal to make the detection problem easier with noise. [15, 16, 17] focus on the FDI for distributed systems using the communications between the agents preserving local navigation. In such cooperative settings, [18] developed a cooperative integrity monitoring (CIM) algorithm using a decomposition method to isolate the fault. Besides, information techniques have been applied and improved for FDE, which is a quite success. [19] minimizes integrity risk for RAIM detector. Further-

more, a more systematic analysis of information theoretic approach to detection problem can be found in [20]. [21, 22, 23] carry out the FDE scheme for GNSS/INS tightly coupled systems with the use of information entropy and divergence such as mutual information, Kullback-Leibler divergence (KLD) and Rényi Divergence (RD) for the synthesis of detection residuals. The scheme is based on the local test (LT) and global test (GT) with optimal thresholding which can isolate multiple faults and exclude them recursively. One main advantage of the GNSS/INS tightly coupled systems is that they reduce the minimum number of observable satellites to 1.

1.2.2 Measurement modeling and bias compensation

The measurements acquired by multiple sensors contain noise and bias besides faults. In most of the cases, measurement noises are modeled as Gaussian random variables. The sources of bias, however, vary between sensors. For example, [24] models and compensates for the bias in vibrating gyroscopes. [25] focuses on the systematic bias of GPS due to atmospheric properties. In addition, the UWB range suffers from significant bias due to the none-line-of-sight (NLoS) scenarios [26]. It is important to correct for these bias in order to carry out data fusion in multi-sensor navigation systems. In this dissertation work, we focus on the UWB measurement bias modeling and compensation, whereas it can be easily generalized to any RF signal based ranging technologies.

Among various RF signals for ranging, the UWB signal has received considerable attention for indoor localization applications [27, 28]. UWB signals, because of their short impulse, are less susceptible to interference with each other or with other coexisting radio signals such as WiFi and Bluetooth [29]. UWB signals are also energy efficient due to their low power consumption [29]. Compared to radar-based techniques, UWB has a better accuracy ratio and a higher signal-to-noise ratio [30, 31, 32]. Other favorable attributes of UWB

include low cost, high time-resolution, and obstacle-penetrating NLoS ranging [30, 33, 34, 35]. However, NLoS and multipath radio propagation can lead to biased measurements due to the time increment for the signals to penetrate through obstructions or travel a longer non-direct path [36], [37]. It is also observed that even the line-of-sight (LoS) UWB range measurements exhibit spatially varying bias due to the relative pose and orientation among UWB sensors [38] and their antenna radiation patterns [39]. Therefore, to ensure UWB localization accuracy, it is critical to account for bias in the UWB range measurements before using them. Bias correction/compensation in the UWB ranging is not fully resolved and is an active area of research.

NLoS UWB ranging bias is often significant. To deal with UWB NLoS bias, some localization approaches rely on identifying NLoS signals and avoid using them for ranging [40, 41, 42, 43]. However, excluding NLoS range measurements and employing a LoS-only UWB ranging system means that in complex and cluttered indoor environments where most of the measurements are obtained in the NLoS cases, we should deploy a large number of beacon transceivers to increase the probability of taking LoS measurements. Still, NLoS scenarios may not be fully avoidable because of mobile obstacles in the environment, such as pedestrians.

There are mainly two approaches in the literature to handle the bias in UWB range measurements. The first is the model-based bias correction, where the bias is either modeled with a known analytical expression [44, 45, 46] or considered as a parameter in the state vector to be jointly estimated with other state parameters [47, 48, 49, 6, 50]. In practice, however, the prior information about the bias may not be available accurately in advance. The second approach is the model-free bias correction using ML techniques. The main idea is using ML methods, e.g., support vector machine (SVM) [51, 52], k-nearest neighbor (k-NN) [53], deep neural network (DNN) [54, 55], convolutional neural network (CNN) [56], long short-term memory (LSTM) network [57] for LoS and NLoS classification and to learn the bias in NLoS scenarios. The corresponding features for classification and learning the bias include the

channel impulse responses, the energy of the received signal, the maximum amplitude of the received signal, and the mean excess delay. However, the channel impulse response features are not readily available in low-cost UWB devices [58]. Moreover, complex learning models with a large number of features also come with high computational complexity, which may limit the use of these models in embedded systems. A lightweight learning method to estimate the UWB ranging bias in NLoS was proposed in [59]. But this method uses the relative pose as a learning feature, which may not easily be available in every localization application. The method in [59] in fact is proposed for UWB time-difference-of-arrival (TDOA)-based localization which involves three UWB radios instead of two, which is used in the ToF-based localization applications.

In most existing supervised learning-based bias compensation/correction work such as [60, 61], the acquisition of ranging bias values is often assumed trivial and is usually obtained by measuring the ground truth distance between UWB sensors with the tape ruler or the laser tracker and computing the difference between the measured true range and the UWB range measurement. The accuracy of the measured true range directly affects the learning performance whereas measuring the true range (ground truth) between UWB sensors by a human can introduce extra errors. The training data for the ML algorithms are often also obtained from stationary sensors. However, in real localization applications, the UWB sensors attached to moving targets are maneuvering, which induces potential data mismatch issues due to the additional bias caused by the relative pose of UWB sensors. The relative pose and the antenna radiation patterns of UWB sensors are not often considered in the literature only until some recent work [62, 59]. Even so, the existing methods require bias modeling with respect to the relative pose of the UWB sensors or the access to the relative pose when learning and predicting the bias. And they can only deal with pose-dependent bias due to the limited feature selection. The NLoS scenarios are also not included. The use of a motion capture camera in UWB ranging is reported in [59] for UWB bias estimation in LoS and in [63] to generate aiding measurements to improve the localization of an integrated

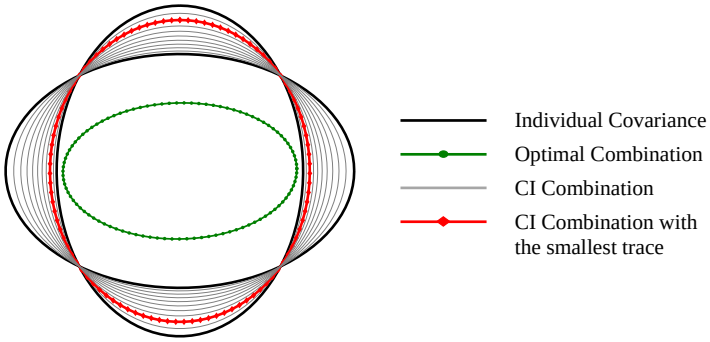
UWB/IMU system.

1.2.3 Computational Complexity

Reducing the computational complexity and enhancing the robustness of algorithms are critical and essential in real-time navigation systems. In this dissertation work, we focus on light-weighting the cooperative localization (CL) algorithm and generalizing it to relax the connectivity requirement for the absolute navigation solution.

In the multi-agent sensing systems, to reduce communication cost and distribute computation, agents compute and maintain their own local estimates whose accuracies are then improved by fusing the estimates of neighboring agents (in the case of data fusion) or using relative measurement feedback between neighboring agents (in the case of, e.g., CL). Such acts, however, create strong correlations among the state estimates of the agents. Neglecting the correlations often leads to excessively optimistic estimates and even the inconsistency of the estimator [64]. In sensor networks, accounting for correlations among the state estimates of the estimation nodes continues to be a challenging task. The approaches to solving this problem can be classified into explicit and implicit methods. Explicitly accounting for the correlations requires a higher computational complexity and demands frequent communication among correlated agents or a server [65, 66, 67, 68]. In contrast, implicit treatments, either via the use of conservative uncorrelated estimate upper bounds to guarantee estimation consistency or by estimating the unknown correlations, eliminate persistent inter-agent communications but come with higher communication costs.

The prime example of the implicit fusion algorithms that use conservative uncorrelated estimates is the Covariance Intersection (CI) method [69], see Fig. 1.2. CI has been used in various sensor network fusion problems [70, 71, 72] and also CL problems [73, 74, 75, 76]. Alternatively, for CL, [48] proposes the discorrelated minimum variance (DMV) relative



$$\omega^\star = arg \min_{\omega} trace[\omega(\mathbf{P}_1^-)^{-1} + (1-\omega)(\mathbf{P}_2^-)^{-1}]^{-1}$$

$$\mathbf{P}^+(\omega^\star) = [\omega^\star(\mathbf{P}_1^-)^{-1} + (1-\omega^\star)(\mathbf{P}_2^-)^{-1}]^{-1}$$

$$\hat{\mathbf{x}}^+ = \mathbf{P}^+(\omega^\star)[\omega^\star(\mathbf{P}_1^-)^{-1}\hat{\mathbf{x}}_1^- + (1-\omega^\star)(\mathbf{P}_2^-)^{-1}\hat{\mathbf{x}}_2^-]$$

Figure 1.2: CI produces the fused estimate $(\hat{\mathbf{x}}^+, \mathbf{P}^+(\omega^\star))$ from two individual estimates $(\hat{\mathbf{x}}_l^-, \mathbf{P}_l^-)$, $l \in \{1, 2\}$ whose cross covariance $\mathbf{P}_{12}^-$ is unknown.

measurement processing method, which does not directly use the CI but is reminiscent of the CI's approach in using uncorrelated upper bounds on the joint uncertainty matrix of any two estimation nodes. To reduce the conservatism of the CI method, alternative approaches such as Ellipsoidal Intersection (EI) [77] and Inverse Covariance Intersection (ICI) [78] have been proposed in the literature. CI, as well as its alternative approaches, EI and ICI, and also the DMV method, have a scalar parameter which is chosen using an optimization process that minimizes the total uncertainty of the fused states so that the best among the conservative estimates is chosen, see Fig. 1.2 for the graphical presentation of the CI fusion method. The second class of implicit methods to deal with state estimation in the absence of correlation information trades in extra computation for a better fusion performance by constructing the unknown cross-covariance matrix instead of conservatively over-bounding the joint covariance matrix of the estimates. The results in this class include the Maximum Allocated Covariance (MAC) of [79] and game-theoretic method of [80] for track-to-track fusion and the practical estimated cross-covariance minimum variance (PECMV) of [48] for CL. All three methods estimate the unknown cross-covariance matrix of any two nodes using a matrix optimization framework. They deliver a less conservative estimate than CI and its variants but computationally are expensive, to the point that they may not be solved online,

especially on an embedded computing system.

To reduce the computational cost of the consistent implicit track-to-track fusion and CL methods, we propose using machine learning (ML) based solvers to solve the optimization problems used in these methods. ML-based solutions for optimization problems have been successfully used in applications such as combinatorial optimization [81], wireless network optimization [82], and supply chain management [83]. ML-based approaches have also been used in CL problems but for different purposes. For example, [84] uses a DNN to assist a CL for vehicular networks, where the DNN is designed to solve the chronic nonlinear approximation problem. Or, [85] uses an ML-based surrogate model as a measurement scheduling merit function.

1.2.4 Sensor Coverage Enhancement

In a typical scenario, the infrastructure beacon installment and the survey of their position coordinates are performed manually via specialized equipment (e.g., motion capture cameras and laser distance measurers) after the installation [86, 87], which is human error-prone and effort-demanding, especially for large scale networks. Furthermore, in a large scale deployment, measurement station must be moved frequently in the target area to ensure the line-of-sight towards all beacons, which requires multiple re-calibration of the ground truth along with converting all measured positions to a common global coordinate frame [88]. Such a method cannot be applied for portable beacons.

The demand for automatic computing of the beacon positions via senor measurements and providing flexibility in beacon deployment has led to the development of an alternative approach, self-localization, which can be formulated as a parameter or state estimation problem. In the literature, most of the work focus on trilateration method to localize beacons using at least two linear measurements or three non-linear measurements from the pre-

installed beacons with accurate known positions nearby [89, 90, 91, 92], which is the strict requirement of the trilateration method even though the workload of manually positioning has been significantly reduced. If the beacon to be deployed has been placed in the area lacking sufficient connections to known beacons, the trilateration method fails. To relax the connectivity, mobile beacons are adopted to provide extended coverage of sensor networks [93, 94, 95]. The mobile beacons can be any portable mobile robot or vehicle carrying the wireless sensors. Then the mobile beacons can be controlled and assigned to the area with enough measurements for self-localization. However, there still exist the scenarios that the areas of interest are not covered yet by the sensor networks. As depicted in Fig. 6.1, the beacons are deployed at some target locations but only beacon 2 has three connections to known beacons, which constrains the self-localization process. Moreover, recent works conduct beacon self-localization in a centralized manner, where the correlation between the deployed beacons are completely ignored or mishandled [88], [96, 97, 95]. And due to the mobility of the deployed beacons, centralized self-localization cannot suitably capture the dynamical changes of the network.

1.3 Objective

The objective of this dissertation work is design of the robust and light-weight estimation process that simultaneously enhance the navigation accuracy and monitor the sensor performance. The main application of this work is the integrity monitoring and measurement enhancement solution for INS aided by external measurements obtained from GNSS and/or UWB sensors in a cooperative manner. This work is end-to-end spanning algorithm design, theoretical modeling and analysis, and experimental demonstration/validation.

In an effort to develop practical integrity monitoring algorithms, in this dissertation, we develop an information-theoretic based FDE method for tightly coupled GNSS/INS system

that can deal with multiple-fault case with fast alarm ability and less iterations. We then upgrade the method with an probabilistic threshold function to autonomously choose the FDE threshold according to the measurements, which enhance the robustness of the algorithm further.

Another objective of this work is the inference of the bias in UWB measurements. We also study the UWB sensor placement for foot-to-foot ranging for pedestrian inertial navigation. The aim is to find out the intrinsic factors that induce the bias in UWB measurements to remove the measurement bias and provide bias-free output for the demands of robust navigation.

The third objective of this dissertation work is to reduce the computational complexity of navigation algorithms with an elaborate design to respect the constraints and thus enable real-time implementation and generalization of navigation algorithms.

Lastly, we desire to propose an algorithm to localize portable wireless beacons that are deployed to tackle the limited connectivity/coverage of the infrastructure beacons.

1.4 Outlines

In Chapter 2, a residual-based FDE algorithm is developed to enhance the performance of the tightly coupled GNSS/INS system. Specifically, the Rényi divergence (RD) is introduced as a measure to evaluate the divergence between the state estimate of propagation and update steps to indicate whether there exists a fault in the system. The properties of this indicator mechanism are discussed analytically in detail. The FDE algorithm is constructed based on this divergence measure. Appropriate methods to select the parameters of the proposed divergence measure to increase the probability of the fault detection and decrease the false alarm rate of the FDE algorithm are proposed. The results are demon-

strated and validated in the Computer Aided Design of Aerospace Concepts (CADAC++) flight simulation platform. Furthermore, the probabilistic threshold method is proposed and validated via a set of UWB/INS navigation simulations. In Chapter 3, we study the impact of the UWB sensor placement for a Zero-velocity-update (ZUPT)-aided pedestrian inertial navigation solution that uses a foot-to-foot ranging feedback to improve its localization accuracy. Several sensor configurations are investigated in both static and dynamic cases. The Root Mean Square Error (RMSE) is used as the performance metric for the static measurement accuracy while the Circular Error Probable (CEP) is chosen to evaluate the sensor configurations in dynamic experiments. We also conduct research on the UWB LoS/NLoS identification and bias compensation and propose a learning-based probabilistic identification and compensation method in Chapter 4. Chapter 5 presents two data-driven approaches to generate the solutions of two different constrained optimization procedures contained in implicit CL algorithms. For the first approach, the artificial neural network (ANN) technique is used to predict the scalar solution of the problem with a single inequality constraint while in the second method the ANN works as a matrix predictor to learn the solution of a matrix optimization containing linear matrix inequality (LMI) constraints. We discuss the design of the ANNs in detail to respect the constraints. The effectiveness and the generosity of the two methods are demonstrated via CL experiments. And the experimental results show that both approaches reduce the computational cost significantly without sacrificing the localization accuracy and the estimation consistency. In order to generalize the CL for beacon-based localization, in Chapter 6, we report a decentralized beacon self-localization algorithm which relaxes the essential requirement of sufficient connections to the beacons with known positions and avoids the labor-intensive and error-prone manual measurement of the beacon positions. Finally, Chapter 7 gives our conclusions and future work.

Chapter 2

A Rényi Divergence Based Approach to Fault Detection and Exclusion for Tightly Coupled GNSS/INS Systems

In this work, an information theoretic based FDE method for tightly coupled GNSS/INS system is developed which can deal with multiple-fault case with fast alarm ability and less iterations. This method uses a modified form of the so-called Rényi Divergence (referred to as MRD) to monitor how much the propagated state estimate's probability distribution changes after it gets updated with the satellite pseudorange measurements. The faults are detected when the MRD value is beyond a threshold. Also, the concept of individual MRD is proposed to isolate the faulty channel and exclude the erroneous measurements. Furthermore, a ratio test based on the individual MRDs is designed to improve the ability of detecting incipient faults and at the same time separate the abnormal MRD jump caused by the abrupt change in system dynamics and the one associated with faults, which helps with decreasing the false alarm rate. Lastly, the selection of α, the parameter of the divergence measure for MRD is discussed in detail. The results are demonstrated via an extensive set of simulation study

in the 6 degree-of-freedom (6-DoF) CADAC++ environment [98], which is a high-fidelity flight simulator used by industry and the U.S. Air Force to simulate aerospace vehicles in all flight environments. A full Bayesian method for the threshold is adopted to improve the algorithm further and presented in the last subsection.

2.1 Problem Formulation and Objective Statement

In our study we use the CADAC++'s GNSS/INS model, which we briefly review below; for more details see [99]. Let $\mathbf{x} \in \mathbb{R}^{n_x}$ be the state of the vehicle, where $\mathbf{x} = [\hat{s}_1, \hat{s}_2, \hat{s}_3, \hat{v}_1, \hat{v}_2, \hat{v}_3, \hat{b}, \hat{f}]^\top$. Here, $\hat{s} = [\hat{s}_1, \hat{s}_2, \hat{s}_3]$ is the position error and $\hat{v} = [\hat{v}_1, \hat{v}_2, \hat{v}_3]$ is the velocity error. $\hat{b}$ is the user clock bias and $\hat{f}$ is the user clock frequency error. The motion of the vehicle is described by

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{G}\mathbf{u}_k + \boldsymbol{\nu}_k, \tag{2.1}$$

where $\mathbf{u}_k$ is the user acceleration that drives the INS dynamics. $\boldsymbol{\nu}_k$ is the process noise. The noises are assumed to be zero-mean white Gaussian with invariant covariance matrices given

by $E[\boldsymbol{\nu}_k \boldsymbol{\nu}_k{}^\top] = \mathbf{Q}_k = \mathbf{Q} > \mathbf{0}$. The fundamental matrix $\mathbf{F}$ and control matrix $\mathbf{G}$ are

$$\mathbf{F} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/T_f \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

where T_f is the correlation time constant.

The GPS measurement taken at time k is described by

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \boldsymbol{\omega}_k + \mathbf{f}_k, \quad \mathbf{z}_k \in \mathbb{R}^{n_z}, \tag{2.2}$$

where $\mathbf{z}_k = [\mathbf{z}_{k,1}(1), \cdots, \mathbf{z}_{k,N}(1), \mathbf{z}_{k,1}(2), \cdots, \mathbf{z}_{k,N}(2)]^\top$ with $\mathbf{z}_{k,i} = \left[\Delta\rho_i, \Delta\frac{\mathrm{d}|\rho_i|}{\mathrm{d}t}\right]^\top$, $i \in \{1, \cdots, N\}$, is the residual measurement vector of pseudorange and range-rate at GPS channel i. $\mathbf{f}_k$ is the fault vector whose entries are non-zero if and only if the corresponding measurement channels are faulty. The measurement noise $\boldsymbol{\omega}$ is white and Gaussian with predefined bias and invariant covariance matrix $E[\boldsymbol{\omega}_k \boldsymbol{\omega}_k^\top] = \mathbf{R}_k = \mathbf{R} > \mathbf{0}$. The linearized observation matrix, which should be computed every step, is CADAC++ simulation uses 4

GPS channels, therefore, in our simulation study $N = 4$.

$$\mathbf{H} = \left[\begin{array}{c|c|cc} & & 1 & 0 \\ [\mathbf{u}_{SB}]^I_{4\times 3} & \mathbf{O}_{4\times 3} & 1 & 0 \\ & & 1 & 0 \\ & & 1 & 0 \\ \hline & & 0 & \Delta\tau \\ & & 0 & \Delta\tau \\ \mathbf{O}_{4\times 3} & \Delta\tau[\mathbf{u}_{SB}]^I_{4\times 3} & 0 & \Delta\tau \\ & & 0 & \Delta\tau \end{array}\right],$$

where each column of matrix $[\mathbf{u}_{SB}]^I_{4\times 3}$ is the unit vector pointing from the user B to the satellite S, expressed in the inertial coordinates. $\Delta\tau$ is the GPS update interval which is set to 1 second.

By use of the conventional extended Kalman filter (EKF), the state estimation process is expressed as follow. The propagation equation is described by

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{H}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{G}\mathbf{u}_k, \tag{2.3a}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}\hat{\mathbf{P}}_{k-1|k-1}\mathbf{F}^\top + \mathbf{Q}_k. \tag{2.3b}$$

The estimate is updated according to

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\left(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}\right), \tag{2.4a}$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_{k|k-1}, \tag{2.4b}$$

where $\mathbf{K}_k$ is the Kalman gain, which can be obtained by

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}^\top(\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^\top + \mathbf{R}_k)^{-1}. \tag{2.5}$$

The faulty measurements corrupt the updated state estimates by corrupting the innovation term $(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1})$ in (2.4a). If the measurements are continuous and not faulty, the updated and propagated estimates $(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$ and $(\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$ respectively, are expected not to differ from one another significantly. We use the close form of RD measure (2.6) under Gaussian assumptions to compute the difference between the updated and the propagated estimate distributions $f(\hat{\mathbf{x}}_{k|k-1}|\mathbf{z}_{1:k-1})$ and $f(\hat{\mathbf{x}}_{k|k}\mathbf{z}_{1:k})$, which at each time step k instantiate as

$$\begin{aligned}
\mathrm{RD}(f(\hat{\mathbf{x}}_{k|k-1}|\mathbf{z}_{1:k-1})||f(\hat{\mathbf{x}}_{k|k}\mathbf{z}_{1:k})) &= \frac{\alpha}{2}(\hat{\mathbf{x}}_{k|k}-\hat{\mathbf{x}}_{k|k-1})^\top(\mathbf{P}^*_{k|k})^{-1}(\hat{\mathbf{x}}_{k|k}-\hat{\mathbf{x}}_{k|k-1}) \\
&\quad -\frac{1}{2(\alpha-1)}\ln\frac{\det(\mathbf{P}^*_{k|k})}{\det(\mathbf{P}_{k|k-1})^{1-\alpha}\det(\mathbf{P}_{k|k})^{\alpha}} \\
&= \frac{\alpha}{2}(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1})^\top\mathbf{K}_k^\top(\mathbf{P}^*_{k|k})^{-1}\mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}) \\
&\quad -\frac{1}{2(\alpha-1)}\ln\frac{\det(\mathbf{P}^*_{k|k})}{\det(\mathbf{P}_{k|k-1})^{1-\alpha}\det(\mathbf{P}_{k|k})^{\alpha}},
\end{aligned} \tag{2.6}$$

where $\mathbf{P}^*_{k|k} = \alpha\mathbf{P}_{k|k}+(1-\alpha)\mathbf{P}_{k|k-1}$. The faulty measurements cause a significant deviation of the updated state and it is expected that the RD measure (2.6) when measurements are faulty to be a large value. Our objective is in this dissertation work is to monitor the integrity of the tightly coupled INS/GPS to detect faulty GPS measurements by developing a RD-based FDE algorithm. That is we apply RD as a measure of the deviation or distance between the propagated and updated estimates. Intuitively, the distance of the two estimates should be within certain range while the outliers of the distance are regarded as faulty according to some statistical criteria. In what follows, we carefully construct the statistical criteria that can be used for fast detection of the GPS faulty measurements.

2.2 A modified Rényi Divergence Based FDE Algorithm

$\mathrm{RD}(f(\hat{\mathbf{x}}_{k|k-1}|\mathbf{z}_{1:k-1})||f(\hat{\mathbf{x}}_{k|k}\mathbf{z}_{1:k}))$ given by (2.6) provides a measure to compare the distribution of the updated and propagated state estimate of the navigation filter for fault detection. However, careful inspection of (2.6) reveals that the existence or lack of fault does not make any difference in the value of the second compound, $\frac{1}{2(\alpha-1)}\log\frac{\det(\mathbf{P}^*_{k|k})}{\det(\mathbf{P}_{k|k-1})^{1-\alpha}\det(\mathbf{P}_{k|k})^{\alpha}}$, of (2.6). This is because the only way that the update equations (2.4) are affected by the fault is due to the innovation feedback $(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1})$ in (2.4a). Therefore, in our FDE algorithm design below we only use the modified version of the RD measure where only the first term of (2.6) is considered, i.e.,

$$\mathrm{MRD} = (\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1})^\top \mathbf{K}_k^\top \mathbf{P}^{*-1}_{k|k} \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}), \tag{2.7}$$

where $\mathbf{P}^*_{k|k} = \alpha\,\mathbf{P}_{k|k} + (1-\alpha)\mathbf{P}_{k|k-1}$. Figure 2.1 shows RD and MRD plots for a simulation scenario in ROCKET6G where 10 impulsive faults are introduced every 30 seconds starting at $t = 50$ seconds in one of the GPS channels when $\alpha = 0.1$ is used. As we can see in the red plot, the difference between the RD and MD stays flat and shows no effect in detection the fault.

Given the measurement model (2.2), we observe that MRD is a quadratic term of the fault vector $\mathbf{f}_k$. For healthy measurements, the fault vector $\mathbf{f}_k$ is a zero vector so that the innovations $(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1})$ form a zero-mean white sequence with covariance $\mathbf{S}^I_k = \mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^\top + \mathbf{R}_k$. Then, the hypothesis that the filter is consistent and the healthy $\mathrm{MRD} = (\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1})^\top \mathbf{K}_k^\top \mathbf{P}^{*-1}_{k|k} \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1})$ has a chi-square distribution with n_z degrees of freedom.

Our objective is to design a FDE algorithm that at each time step k given a set of GPS

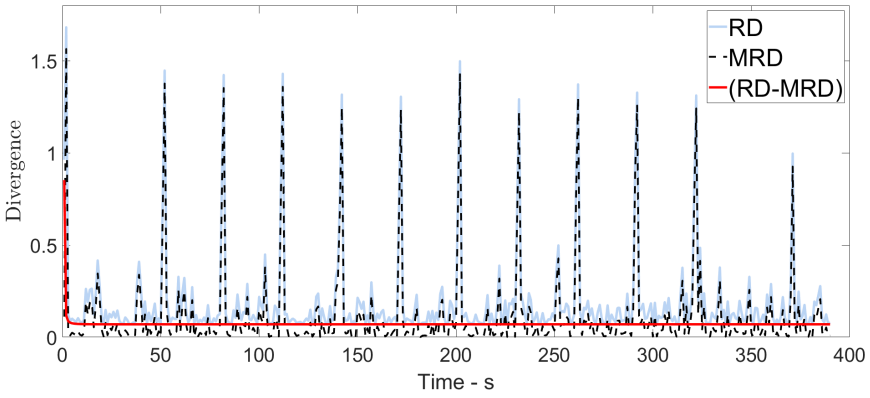


Figure 2.1: RD and MRD during a simulation scenario in ROCKET6G where 10 impulsive faults are introduced every 30 seconds starting at $t = 50$ seconds in one of the GPS channels.

measurements from N channels, can identify the set of non-faulty (healthy) measurements $\mathcal{V}_h \subset \{1, \cdots, N\}$ and use them to update the INS propagated state estimates. To preform the FDE algorithm in a computationally efficient manner, it is preferable to use the information filter (IF) form of the EKF to compute the updates. The IF deals with the information vector and information matrix obtained from the state vector and the covariance matrix as follows

$$\mathbf{Y}_{k|k-1} = \mathbf{P}_{k|k-1}^{-1} \tag{2.8a}$$

$$\mathbf{y}_{k|k-1} = \mathbf{Y}_{k|k-1}\mathbf{x}_{k|k-1} \tag{2.8b}$$

$$\mathbf{Y}_{k|k} = \mathbf{Y}_{k|k-1} + \sum\nolimits_{i\in\mathcal{V}_h} \mathbf{H}_i^\top \mathbf{R}_{i,k}^{-1}\mathbf{H}_i \tag{2.8c}$$

$$\mathbf{y}_{k|k} = \mathbf{y}_{k|k-1} + \sum\nolimits_{i\in\mathcal{V}_h} \mathbf{H}_i^\top \mathbf{R}_{i,k}^{-1}\mathbf{z}_{i,k}. \tag{2.8d}$$

Here, $\mathbf{y}_{k|k-1}$ and $\mathbf{Y}_{k|k-1}$ are, respectively, propagated information vector and propagated information matrix, while $\mathbf{y}_{k|k}$ and $\mathbf{Y}_{k|k}$ are, respectively, updated information vector and updated information matrix. Using IF, the updated state estimate and corresponding error

covaraince then is to (2.3), where

$$\mathbf{P}_{k|k} = \mathbf{Y}_{k|k}^{-1}, \quad \mathbf{x}_{k|k} = \mathbf{Y}_{k|k}^{-1}\,\mathbf{y}_{k|k}. \tag{2.9}$$

Without an FDE algorithm, $\mathcal{V}_h = \{1, \cdots, N\}$, where N is the number of the available GPS channels. We let the updates due to use of individual measurement channels be

$$\mathbf{Y}_{k|k}^{i} = \mathbf{Y}_{k|k-1} + \mathbf{H}_i^{\top}\mathbf{R}_{i,k}^{-1}\mathbf{H}_i \tag{2.10a}$$

$$\mathbf{y}_{k|k}^{i} = \mathbf{y}_{k|k-1} + \mathbf{H}_i^{\top}\mathbf{R}_{i,k}^{-1}\mathbf{z}_{i,k}. \tag{2.10b}$$

Then, the updated estimates after implementing the FDE algorithm and identifying the healthy measurement set $\mathcal{V}_h$ is

$$\mathbf{Y}_{k|k} = (N - |\mathcal{V}_h|)\,\mathbf{Y}_{k|k-1} + \sum\nolimits_{i\in\mathcal{V}_h} \mathbf{Y}_{k|k}^{i} \tag{2.11a}$$

$$\mathbf{y}_{k|k} = (N - |\mathcal{V}_h|)\,\mathbf{y}_{k|k-1} + \sum\nolimits_{i\in\mathcal{V}_h} \mathbf{y}_{k|k}^{i}, \tag{2.11b}$$

where the final updated estimate is given by (2.9). Note that MRD (2.7) can also be compute equivalently as

$$\begin{aligned}\text{MRD} &= (\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1})^{\top}\mathbf{K}_k^{\top}\mathbf{P}_{k|k}^{*-1}\mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}) \\ &= \left(\sum\nolimits_{i\in\mathcal{V}_h} \mathbf{H}_i^{\top}\mathbf{R}_{i,k}^{-1}\mathbf{z}_{i,k}\right)^{\top} \mathbf{P}_{k|k}^{*-1} \left(\sum\nolimits_{i\in\mathcal{V}_h} \mathbf{H}_i^{\top}\mathbf{R}_{i,k}^{-1}\mathbf{z}_{i,k}\right),\end{aligned} \tag{2.12}$$

where $\mathbf{P}_{k|k}^{*} = \alpha\,\mathbf{Y}_{k|k}^{-1} + (1-\alpha)\mathbf{P}_{k|k-1}$.

2.2.1 Selection of parameter α

The value of the RMD measure depends on the choice of parameter α. To ensure that $\mathbf{P}^*_{k|k}$ is positive definite, we choose $\alpha \in (0, 1)$. Use of $0 < \alpha < 0.5$ puts more weight on the propagated uncertainty level whereas $0.5 < \alpha < 1$ puts more weight on the uncertainty level of the updated estimates. In a recent work [23] where Rényi Divergence divergence is used for design of an FDE algorithm, $\alpha = 0.5$ is used so there is no particular emphasise on either of propagated or updated uncertainty level. Here, we employ a different approach. As we can see in Fig. 2.1 the faults result in spike in the divergence measure. However, healthy measurements also can lead to spikes when there is drastic changes in the system dynamics. For example in case of ROCKET6G model, at 371 seconds there is a spike in the divergence measure value due to the drastic change in dynamical model. Our proposed method to choose α is to select a value for which the average ratio of the MRD caused by faulty measurements and the largest "healthy" MRD is at its largest value so that the distinction between the faulty and non-faulty case is pronounced, i.e., the larger the ratio is, the more separable the faulty MRD is. Figure 2.2 shows MRD measure for different values of α in a simulation scenario in ROCKET6G where 10 impulsive faults of magnitude 10m are introduced every 30 seconds starting at $t = 50$ seconds in one of the GPS channels. Table 2.1 shows the average value of the divergence spike value due to faulty measurements to the natural spike of the system when measurements are healthy at $t = 371$ seconds. Table 2.1 shows also this ratio for when KLD measure is used. As Table 2.1 shows, the best value for α, where we have the most distinction is $\alpha = 0.1$. Interestingly the worse result is obtained for the KLD measure. Fig. 2.3 shows a statistical study that is conducted to investigate the separability of faulty and non-fault cases when KLD, MRD with $\alpha = 0.5$ and MRD with $\alpha = 0.1$ are used as divergence measure. As we can see in this figure, the best separation is achieved when MRD with $\alpha = 0.1$ is used as divergence measure.

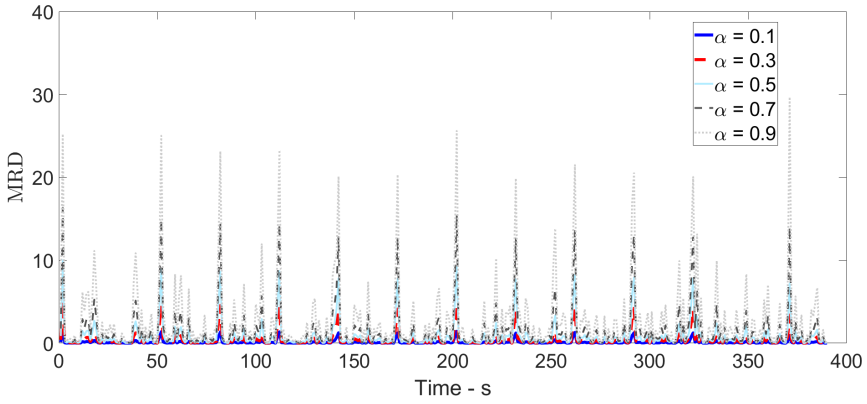


Figure 2.2: MRD of different α values for a simulation scenario in ROCKET6G where 10 impulsive faults of magnitude 10m are introduced every 30 seconds starting at $t = 50$ seconds in one of the GPS channels.

Table 2.1: The average ratio between the two spike values given different measures.

α	average spike ratio (non-healthy to healthy)
0.1	1.4051
0.3	1.2875
0.5	1.1480
0.7	0.9742
0.9	0.7393
KLD	0.6617

2.2.2 Fault detection threshold design

To design our fault detection threshold value, we use a standard hypothesis testing approach, following [100]. Let H_0 denotes the hypothesis that there is no fault in the measurements and H_1 otherwise. According to Neyman-Pearson Lemma, the likelihood ratio is

$$\Lambda(H_1, H_0) = \frac{p(\text{MRD}|H_1)}{p(\text{MRD}|H_0)},$$

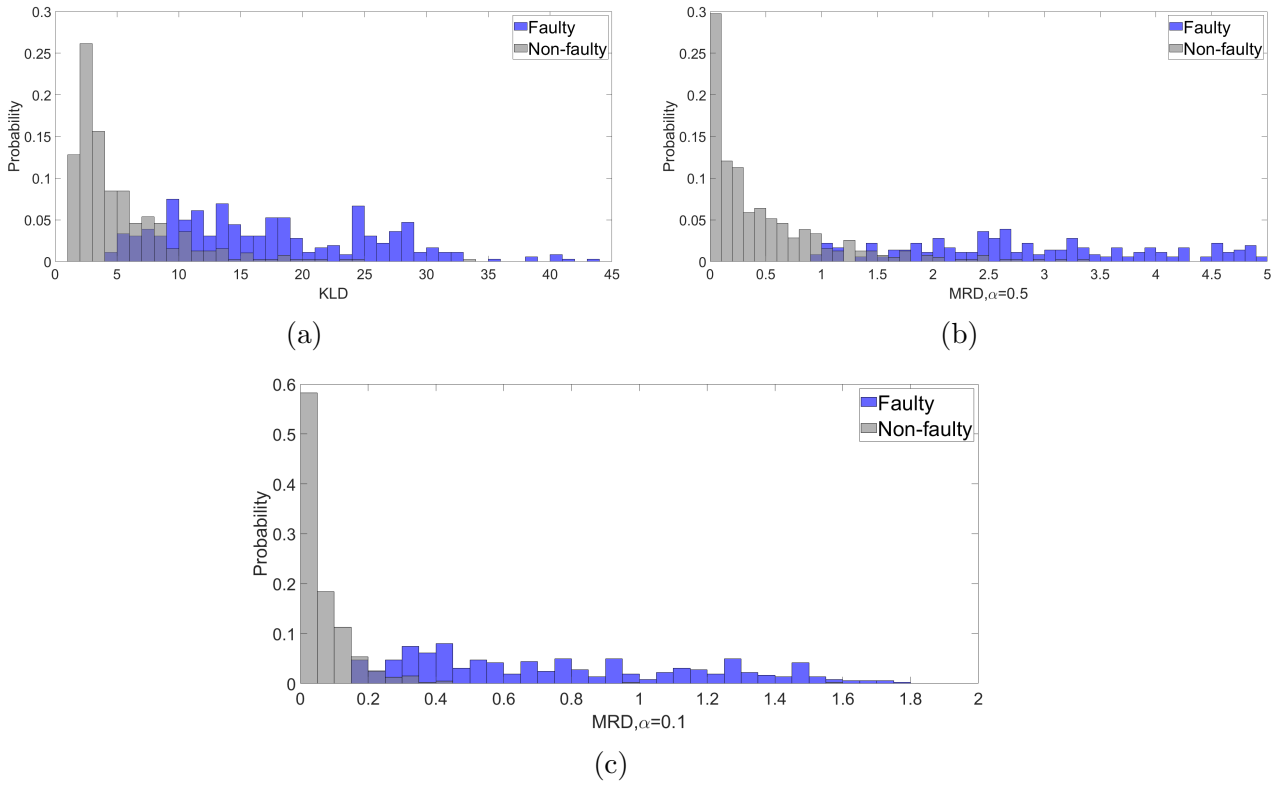


Figure 2.3: Test statistics distributions in the faulty and non-faulty cases when KLD, MRD with $\alpha = 0.5$ and MRD with $\alpha = 0.1$ are used as divergence measure.: the histogram in grey represents the probability distribution of the non-faulty divergence measure values, and the one in blue represents that of the faulty divergence measure values. For the faulty cases, impulsive faults, selected uniformly randomly from $5-15$ meters are added to one of the GPS channel measurements.

where the distributions of $p(\text{MRD}|H_0)$ is obtain by Monte Carlo runs. This expression is equivalent to

$$\text{MRD} \underset{H_0}{\overset{H_1}{\gtrless}} \lambda. \tag{2.13}$$

The threshold λ for false alarm probability β follows from

$$P\{\text{MRD} \in [0, \lambda]|H_0\} = 1 - \beta, \tag{2.14}$$

where the interval $[0, \lambda]$ is the one-sided $1-\beta$ probability concentration region or confidence interval for MRD and β is also called the tail probability. If MRD $> \lambda$, then H_0 is rejected while H_1 is accepted and vise versa. And the threshold λ is obtained by solve for the chi-square cumulative probability function given β in (2.14).

2.2.3 Ratio test to increase the detection accuracy

Faulty measurement is not the only reason that causes spikes in MRD. Rapid change in system dynamics also has the same effect on MRD in terms of the deviation in propagated estimate, which consequently results in false alarms in the detection of fault if only total MRD is considered. Therefore, to reduce false alarms and improve the accuracy of fault detection, we propose a ratio test to utilize the individual MRD computed according to

$$\begin{aligned} \mathrm{MRD}_j &= (\mathbf{H}_i^\top \mathbf{z}_{k,j} - \mathbf{H}_j \hat{\mathbf{x}}_{k|k-1})^\top \mathbf{K}_{k,j}^\top \mathbf{P}_{k|k}^{j*-1} \mathbf{K}_{k,j} (\mathbf{z}_{k,j} - \mathbf{H}_j \hat{\mathbf{x}}_{k|k-1}) \\ &= (\mathbf{H}_j^\top \mathbf{R}_{k,j}^{-1} \mathbf{z}_{k,j})^\top \mathbf{P}_{k|k}^{j*-1} (\mathbf{H}_j^\top \mathbf{R}_{k,j}^{-1} \mathbf{z}_{k,j}), \end{aligned} \tag{2.15}$$

$j \in \{1, \cdots, N\}$, where $\mathbf{P}_{k|k}^{j*} = \alpha\, \mathbf{Y}_{k|k}^{j-1} + (1-\alpha)\mathbf{P}_{k|k-1}$.The idea of the ratio test is considering the case that not all the measurements are faulty simultaneously, after obtaining the individual MRDs, a fault is detected in a particular measurement channel j if and only if the ratio between the corresponding individual MRD_j and the smallest individual MRD in this update step is greater than a predefined threshold γ, i.e.,

$$\frac{\mathrm{MRD_j}}{\min\{\mathrm{MRD_i}\}_{\mathrm{i=1}}^{\mathrm{i=n_z}}} > \gamma, \qquad j \in \{1, \cdots, N\}. \tag{2.16}$$

In healthy measurement scenarios (fault free measurements), all the individual MRDs should be relatively within the same scale despite the possible maneuver of the vehicle. In other words, the measurement channel j is said to fail the ratio test if its corresponding individual

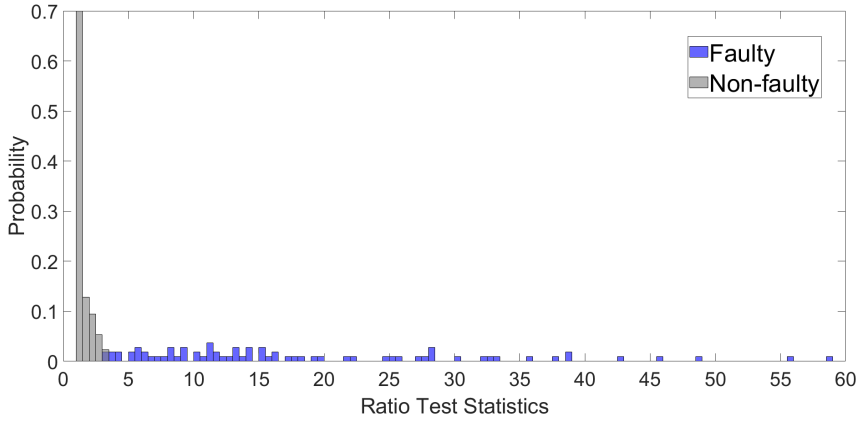


Figure 2.4: Ratio test statistics distribution histogram for 10 Monte Carlo runs with MRD with $\alpha = 0.1$ and fault magnitudes from $5 - 15$ meters.

MRD_j is significantly greater than the remaining ones in the sense that it is γ times larger than others. γ is selected statistically via Monte Carlo runs with introduced faults to make it as the lower bound so that no miss detection exists. We conducted a simulation study in ROCKET6G to determine distribution of the ratio test statistics for impulsive faults with magnitude from $5 - 15$ meters. The result is shown in Fig. 2.4. The low bound of the ratio test statistics, which is 3.2434, is chosen for the value of γ to minimize miss detection rate in the simulation of MRD based FDE algorithm in Section 2.3.

2.2.4 The proposed FDE algorithm

Our proposed MRD-based FDE algorithm is shown in Algorithm 1. In our algorithm, when N satellites are detected, the measurements are all used to obtain the updated state. Then, the MRD measure is calculated for this updated estimate. If the value is above the threshold value λ, the existence of a fault in GPS measurements is declared. Then, for the fault isolation stage, where the INS propagated estimate is updated in n parallel IF filters, which each corresponds to one of the GPS channels (IF_j, $j \in \{1, \cdots, N\}$). Next, the MRD is computed for each update (MRD_j, $j \in \{1, \cdots, N\}$). A ratio test is conducted for these individual MRDs which is explained in the later section. The GPS channels corresponding

Algorithm 1 MRD based FDE Algorithm.

Input: $\{\mathbf{H}_i^\top \mathbf{R}_{i,k}^{-1}\mathbf{H}_i, \mathbf{H}_i^\top \mathbf{R}_{i,k}^{-1}\mathbf{z}_{i,k}\}_{i=1}^N$, $\mathbf{P}_{k|k-1}$, $\hat{\mathbf{x}}_{k|k-1}$, λ, γ
Init: $\mathcal{V}_h = \{1, \cdots, N\}$, $\mathcal{V}_h = \{\ \}$,
Compute total MRD from (2.12)
Compute $\{\text{MRD}_j\}_{j=1}^N$, from (2.15)
Compute $\min\{\text{MRD}_j\}_{j=1}^N$
if MRD $\leq \lambda$ **then**
    No fault detected
else
    for $j \in \mathcal{V}_h$ **do**
        if $\frac{\text{MRD}_j}{\min\{\text{MRD}_j\}_{j=1}^N} > \gamma$ **then**
            $\mathcal{V}_f \leftarrow \{j\}$
        end if
    end for
    $\mathcal{V}_h \leftarrow \mathcal{V}_h \backslash \mathcal{V}_f$
end if
Output: $\mathcal{V}_h$

to the MRD_j that fails the ratio test are declared faulty. These GPS channels are removed from the batch before the fusion step and the remaining GPS measurement are used to check whether the corrected total MRD value is reduced below the predefined threshold value, whose selection procedure is explained later. With the help of ratio test, the isolation and exclusion of the fault can be done in one step instead of being a recurrent process which might potentially have computational issues when the number of faulty channels increases.

2.3 Demonstration Study

In what follows we conduct several simulations study in ROCKET6G in CADAC++ to demonstrate the performance of our proposed FDE Algorithm 1. For these simulations, we use the following parameters for our RMD measure: $\alpha = 0.1$, $\beta = 0.05$ with the corresponding $\lambda = 0.2289$ and $\gamma = 3.2434$. In the first simulation, a total number of 36 impulsive faults with magnitudes randomly selected from $5-15$ meters are added to the pseudorange measurement of GPS channel 2 every 10 seconds starting from $t = 10$ second. The simulation results are

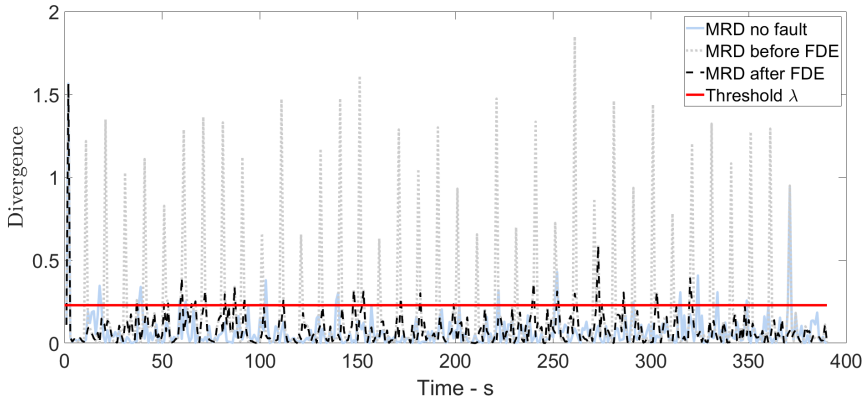


Figure 2.5: MRD plot for the FDE simulation of impulsive faults.

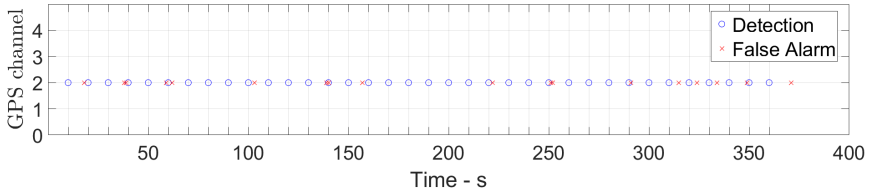


Figure 2.6: Performance of MRD based FDE method for impulsive faults without ratio test aided. The blue circles represent the successful detection of fault while the red crosses represent the false alarm

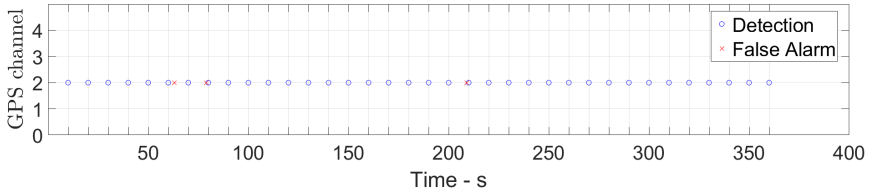


Figure 2.7: Performance of MRD based FDE method for impulsive faults with ratio test aided. The blue circles represent the successful detection of fault while the red crosses represent the false alarm.

shown in Fig. 2.5, Fig. 2.6 and Fig. 2.7. As shown in the figures, without the ratio test, there exist 18 false alarms. The implementation of the ratio test reduces the false alarm significantly according to Fig. 2.6 and Fig. 2.7. The spikes due to dynamics change exceeding threshold γ in Fig. 2.5 are not declared as faulty with the aid of ratio test while the faulty ones are detected.

In the second simulation, three step faults (with duration of 10 seconds each) are added to the pseudorange measurement of GPS channel 2 at $t = 200$ second, $t = 230$ second and

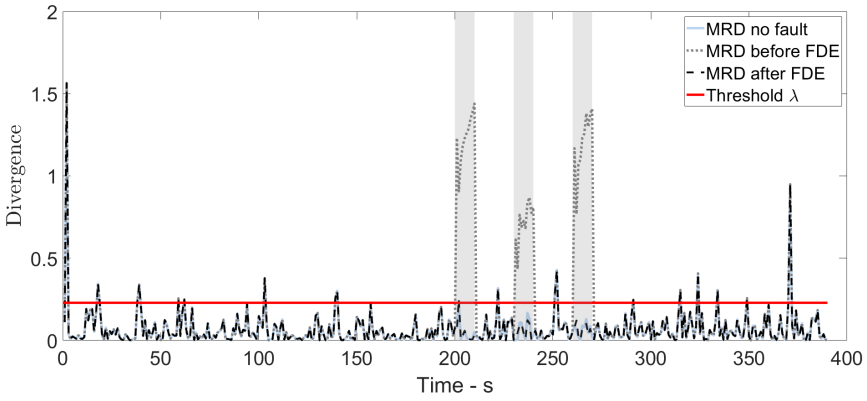


Figure 2.8: MRD plot for the FDE simulation of step faults.

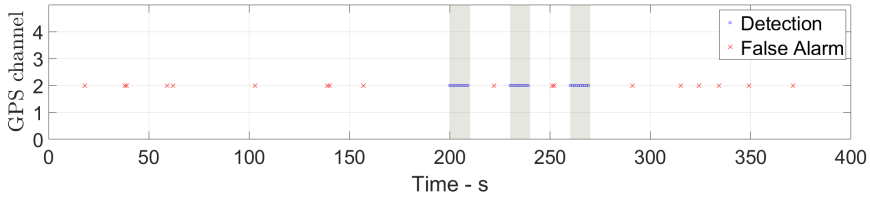


Figure 2.9: Performance of MRD based FDE method for step faults without ratio test aided. The blue circles represent the successful detection of fault while the red crosses represent the false alarm.

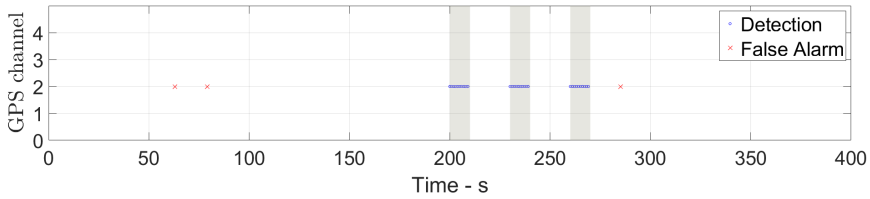


Figure 2.10: Performance of MRD based FDE method for step faults. The blue circles represent the successful detection of fault while the red crosses represent the false alarm.

$t = 260$ second. Fig. 2.8 - Fig. 2.10 demonstrate how the MRD based algorithm behaves in this case. Note that the gray areas in the figures represent the time period when the faults are introduced. As shown, all three step faults are detected and excluded by monitoring MRD with reduced false alarm in Fig. 2.10 and without miss detection after applying ratio test. The fault detection alert is triggered every consecutive step during the fault period because the previous fault is excluded while the upcoming measurements still contain faulty components which make the MRD exceed threshold.

The last simulation demonstrates the performance of our proposed algorithm in detecting

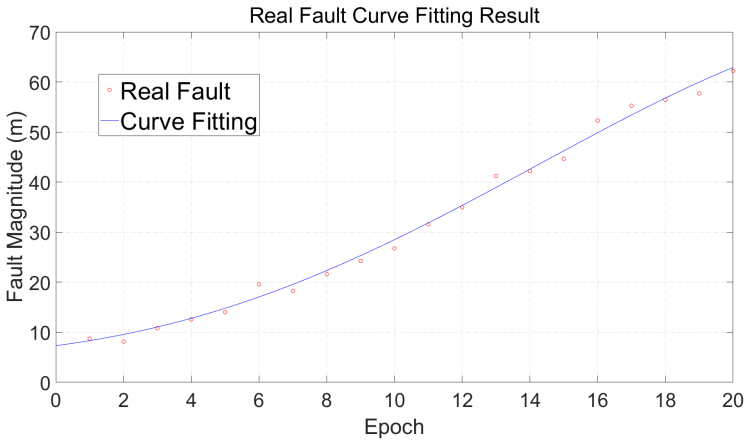


Figure 2.11: A real fault model based on the observations from satellite PRN 23 in Jan. 1st 2004 [1].

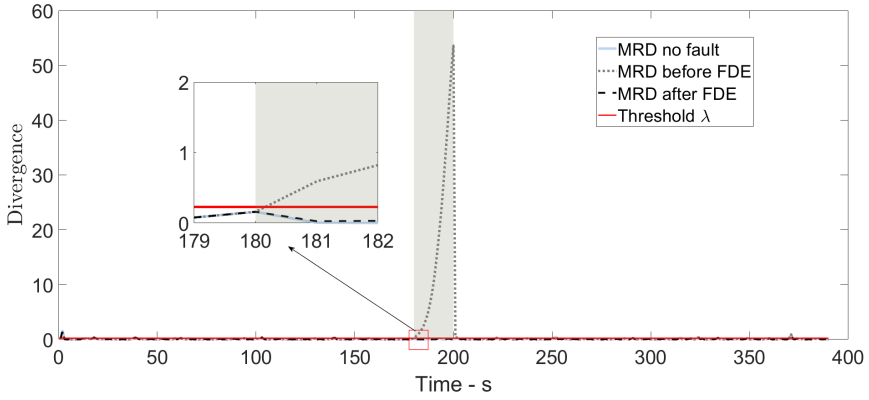


Figure 2.12: MRD plot for the FDE simulation of incipient faults.

incipient faults, which are also called slowly-varying faults. This fault is "deceptive" to many FDE algorithms because if it increases slowly enough, it contaminates the entire state estimate without being detected. An incipient fault model from [1] shown in Fig. 2.11 is used for testing the MRD based FDE algorithm. This incipient fault is added at $t = 190$ second. The behavior of the algorithm is demonstrated in Fig. 2.12 and Fig. 2.13. The incipient fault is detected immediately after it appears, which prevents the state estimate from diverging due to the accumulation of faults in measurements.

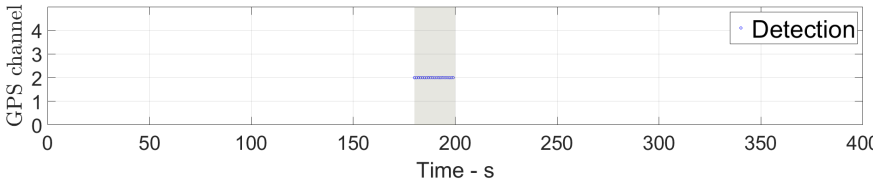


Figure 2.13: Performance of MRD based FDE method for incipient faults. The blue circles represent the successful detection of fault.

2.4 Probabilistic Threshold Estimation

In order to perform the proposed FDE algorithm, after obtaining the prior information (data collection) of the faulty and non-faulty the detection threshold needs to be predetermined and the detection result with respect to the threshold is deterministic and does not fully utilize the information of all the prior information, which makes the FDE algorithm less flexible and less robust. Also, the criteria of determining the threshold varies among the navigation applications, which also limits the performance of the FDE algorithms. Therefore, to enhance our proposed FDE framework in the previous sections, we propose the probabilistic threshold concept with full use of the statistic information via a full Bayesian approach.

2.4.1 Problem Formulation

In the context of robust navigation, we want to eliminate the faulty measurements received by the sensors and thus increase the localization accuracy. In such scenario, the two hypotheses H_0 and H_1 represent "Non-faulty" and "Faulty" measurements respectively. The observation Z is the modified Rényi divergence, MRD, between the propagated and updated estimates of the navigation filter as we introduced in the previous sections. The overall goal is to determine the true hypothesis H between H_0 and H_1. In previous section, we use the standard likelihood ratio test to design the threshold λ. In this section, we propose to handle the FDE threshold $\lambda(\boldsymbol{\theta})$ as a threshold function of the random variable parameter $\boldsymbol{\theta}$

and then use Bayesian inference to estimate $\boldsymbol{\theta}$. Later we will show this is a special parametric case of the Gaussian process classification with linear decision boundary.

2.4.2 Bayesian Inference of FDE Threshold

Assume $\boldsymbol{\theta}$ follows a multivariate Gaussian distribution, i.e., $p(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with unknown mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, then the threshold is also probabilistic and has a distribution $\lambda \sim p(\lambda)$. The objective is to estimate the $\boldsymbol{\theta}$ conditioned on all given observation $\mathbf{Z}$, aiming to obtain the posterior distribution $p(\boldsymbol{\theta}|\mathbf{Z})$, where $\mathbf{Z}$ represent the stack of all non-faulty and faulty MRDs. As such, we first choose intuitively a linear decision boundary function $f_{DB} = \boldsymbol{\theta}^\top \mathbf{Z}$ as our threshold function with λ to be the solution of $f_{DB} = 0$. Then the corresponding probability of H_1 hypothesis is true given the MRD observation $\mathbf{Z}$ is

$$p(H = H_1|\boldsymbol{\theta}, \mathbf{Z}) = \sigma(-\boldsymbol{\theta}^\top \mathbf{Z}) = \frac{1}{1 + \exp(-\boldsymbol{\theta}^\top \mathbf{Z})}, \tag{2.17}$$

where σ is the standard sigmoid function. Consider the likelihood model for MRD observation as

$$p(H|\mathbf{Z}, \boldsymbol{\theta}) = \prod_i \big(H_i p(H = H_1|\boldsymbol{\theta}, Z_i) + (1 - H_i) p(H = H_0|\boldsymbol{\theta}, Z_i)\big), \tag{2.18}$$

where $H_i = 1$ if $H = H_1$ and $H_i = 0$ if $H = H_0$. According to the Bayes rule, the posterior distribution of parameter $\boldsymbol{\theta}$ is obtained as

$$p(\boldsymbol{\theta}|H, \mathbf{Z}) = \frac{p(H|\mathbf{Z}, \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(H|\mathbf{Z})}. \tag{2.19}$$

Since the likelihood function $p(H|\mathbf{Z}, \boldsymbol{\theta})$ is non-Gaussian, the posterior is also non-Gaussian. Thus there is no close-form solution to calculate the posterior $p(\boldsymbol{\theta}|H, \mathbf{Z})$. However, we can approximate the posterior using various methods such as Metropolis–Hastings algorithm

of Markov chain Monte Carlo (MCMC methods [101] and variational inference (VI) [102] methods. Once the posterior is well captured, given a new observation of MRD Z^* the probability of the new measurement is faulty, i.e., $H = H_1$ is predicted by the following formula

$$p(H = H_1|Z^*) = \int p(H = H_1|\boldsymbol{\theta}, Z^*)p(\boldsymbol{\theta}|H, \mathbf{Z})d\boldsymbol{\theta}. \tag{2.20}$$

Practically, the integral can be calculated using the sampled mean as

$$p(H = H_1|Z^*) = \mathbb{E}_{\theta\sim p(\boldsymbol{\theta}|H,\mathbf{Z})}[p(H = H_1|\boldsymbol{\theta}, Z^*)] \approx \frac{1}{N}\sum_{i}^{N} p(H = H_1|Z^*, \boldsymbol{\theta}_i). \tag{2.21}$$

After making the threshold a Gaussian random variable, detection based on the single deterministic threshold is replaced by considering a distribution of the threshold for comparison and the prediction result is an average of all possibility of the threshold values, which in general makes full use of the data information. The graphical interpretation of the Bayesian estimation of the threshold (parameterized linear decision boundary) is shown in Fig 2.14.

In fact, the technique we applied is a special case of the Gaussian process classification, where we choose a parameterized linear decision boundary f_{DB} with parameter $\boldsymbol{\theta}$. And this parameter has a geometric interpretation in the likelihood ratio test. To generalize the probabilistic threshold method, we can discard the parameter and linear assumption. In other words, the decision boundary of faulty and non-faulty can be any arbitrary functions f and even non-parameterized ones whereas we can no longer interpret their geometric meaning. The general inference steps are quite similar to (2.17)$\sim$(2.20).

Prior of f:

$$p(f) = \mathbf{N}(\mathbf{0}, \mathbf{K}), \tag{2.22}$$

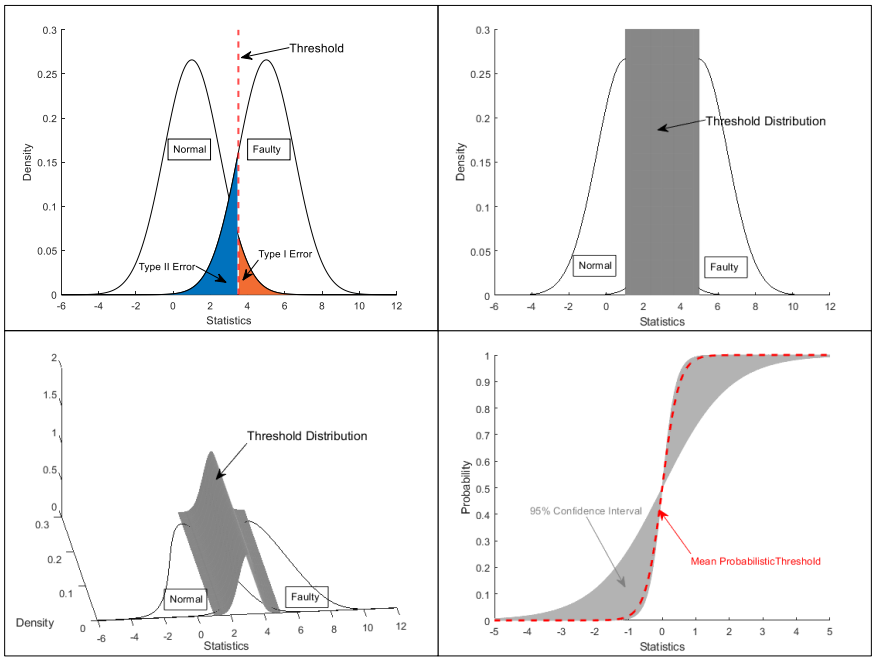


Figure 2.14: The graphical interpretation of the probabilistic threshold derived from the conventional likelihood ration test.

where K is the kernel.

Probability model of $H = H_1$:

$$p(H = H_1|\mathbf{Z}, f) = \sigma(f). \tag{2.23}$$

Likelihood model:

$$p(H|\mathbf{Z}, f) = \prod_i \big(H_i p(H = H_1|f, Z_i) + (1 - H_i)p(H = H_0|f, Z_i)\big). \tag{2.24}$$

Posterior distribution:

$$p(f|H, \mathbf{Z}) = \frac{p(H|\mathbf{Z}, f)p(f)}{p(H|\mathbf{Z})}. \tag{2.25}$$

Prediction:

$$p(f^*|\mathbf{Z}, H, Z^*) = \int p(f^*|f, \mathbf{Z}, Z^*)p(f|H, \mathbf{Z})df \tag{2.26}$$

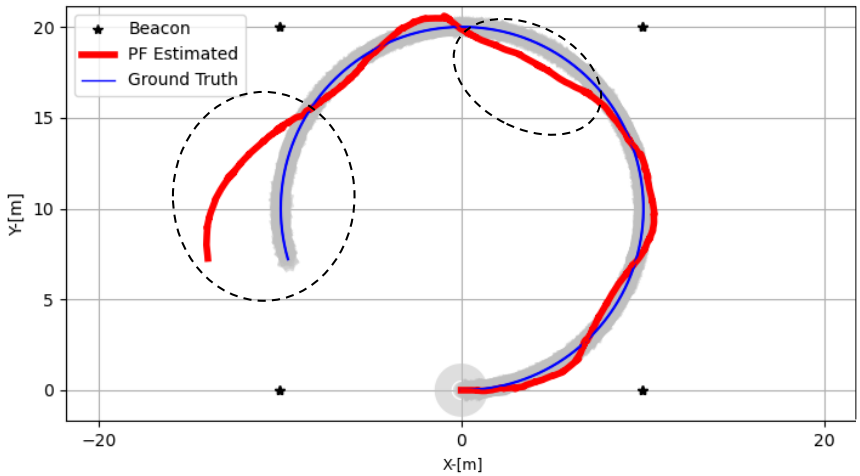
$$p(H = H_1|Z^*) = \int \sigma(f^*)p(f^*|\mathbf{Z}, H, Z^*)df^*. \tag{2.27}$$

Note that the posterior also needs to be approximated using Laplace Approximation or VI. For more information, see [103].

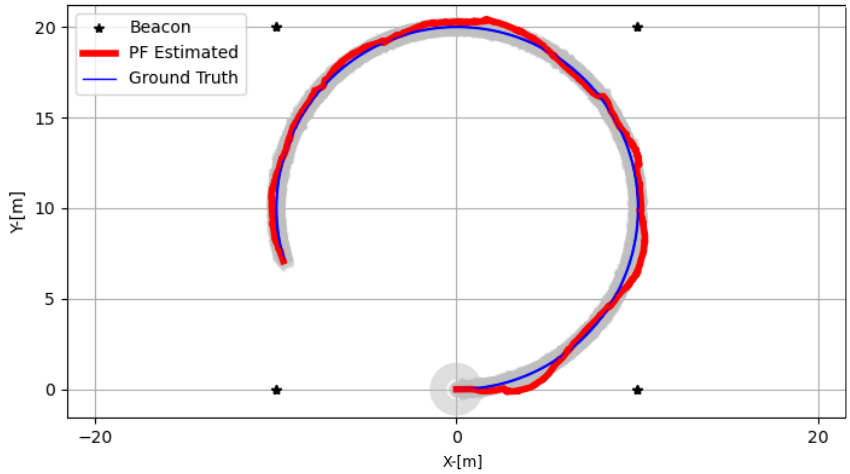
2.4.3 Validation Study

We conducted a particle filter (PF) based target localization simulation using Python to validate the feasibility of our method. The target dynamical model was obtained from the simulated INS while the range measurements were provided by the four beacons at known positions. The designed trajectory was a 3/4 circle. Random fault with magnitudes between $3 - 15$ m were manually added to the range measurements. And the FDE simulation result is shown in Fig 2.15.

As it can be seen in the figure, the probabilistic threshold method effectively discriminated the faulty measurements from the non-faulty ones so that the location accuracy is enhanced. Note that this is a full Bayesian method, comparing to the conventional deterministic methods, the detection uncertainty (confidence) in terms of the prediction variance is generated additionally, which provides more information and enables the evaluation of the detection results. Most importantly, this method utilizes full data and can be used for general FDE applications to generate optimal thresholds (optimal in the sense of minimum mean square error).



(a) w/o FDE



(b) w/ FDE

Figure 2.15: The implementation of the probabilistic threshold method in the simulation of a PF based localization scenario. The dashed circles highlight the impact of the faulty measurements.

2.5 Conclusion

We propose a modified Rényi divergence-based FDE approach for a tightly coupled GNSS/INS system. The main contribution is to analyze the components of the close-form of the Rényi divergence under Gaussian assumption and to propose a modified Rényi divergence as the new test statistics that has a lower computational complexity but delivers the same result. Further, we present a statistical procedure to select the order α of the Rényi divergence

measure for FDE to increase the detection accuracy. Additionally, we introduce a ratio test among the Rényi divergence measure between the propagated and updated estimates due to individual GNSS measurement channels to detect multiple faults in one single check. We demonstrate our results via a simulated flight platform in CADAC++. The simulation results show the desirable performance for FDE in reducing false alarm cases and eliminating miss detection cases. Lastly, we upgrade our FDE approach with a probabilistic threshold method which utilizes full data to generate optimal thresholds autonomously and enables the confidence analysis of the detection, which can be generalized to any FDE applications.

Chapter 3

UWB Sensor Placement for Foot-to-foot Ranging in Dual-foot Mounted ZUPT-aided INS

For the indoor pedestrian navigation, where the Global Navigation Satellite System (GNSS) is usually challenged, the pedestrian dead-reckoning (PDR) or pedestrian inertial navigation system (INS) works as an infrastructure-free self-contained navigation system that does not rely on external signals or pre-installed beacons and landmarks [104]. But, the navigation solution merely based on the INS suffers from high drifts in the position estimation due to a relatively high noise level and unknown time-varying biases in the inertial measurement unit (IMU) measurements. To reduce the growth rate of errors in foot-mounted INS, the zero velocity update (ZUPT) approach is frequently used [105]. ZUPT uses human-legged locomotion and detects the phases of the gate to re-calibrate inertial sensors during the rest phases of the foot. Nonetheless, the ZUPT's performance depends on the pre-determined ZUPT threshold, and it has a systematic error, which becomes significant for long-term navigation. To bound the INS localization error further, in the case of dual foot-mount INS

systems, foot-to-foot relative range measurement feedback has been proposed to aid the INS system [106, 107]. In our work, we propose to use foot-mounted ultra-wideband (UWB) sensors to measure the foot-to-foot range. And we investigate the impact of UWB sensor placement as well as other factors on the navigation accuracy.

3.1 Problem Formulation and Objective Statement

UWB is a radio frequency (RF) signal whose performance can be affected by the transceivers' height from the ground. The relative orientation of the UWB transceivers can also affect the ranging performance. Therefore, to obtain high accuracy UWB range measurements with a low bias for foot-to-foot ranging, in this letter, we investigate the dual-foot mounted UWB sensor placement. Our investigation method includes first exploring the effect of the UWB placement, height, and relative orientation on the foot-to-foot ranging performance when the feet are stationary. Next, we investigate how different placements affect the performance of the ZUPT-aided pedestrian inertial navigation. The Root Mean Square Error (RMSE) is used as the performance metric for the static measurement accuracy. The Circular Error Probable (CEP) is chosen to evaluate the sensor configurations in dynamic experiments. In our experimental study, we used the DWM1000 UWB transceiver, one of the most popular UWB transceivers on the market.

3.2 UWB sensor placement

This section reports on the effect of the UWB sensor placement on the foot-to-foot ranging, and subsequently, on the localization accuracy of a dual-foot mounted ZUPT-aided INS. The testbed we used to collect the data, called the Lab-On-Shoe platform, is shown in Fig. 3.1. Detailed description of the Lab-On-Shoe platform can be found in [108] and [109]. The raw

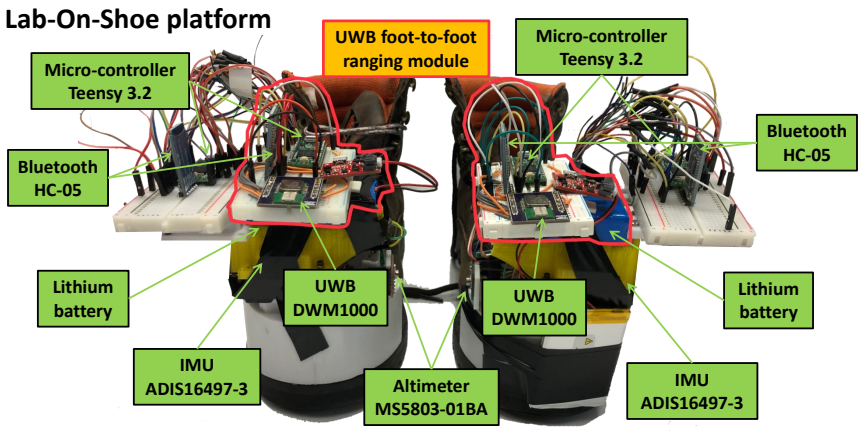


Figure 3.1: The Lab-On-Shoe platform.

measurements obtained from the UWB sensors are the range and the signal power metric (PM). The PM is a popular measure that is used for NLoS UWB signal identification [2], [110]. PM is the difference between the total received signal power and the direct-path signal power. The principle behind the power-based NLoS identification method is that in LoS condition, the power of the received direct-path signal takes a big proportion of the total received signal power, while in NLoS condition the direct-path is significantly attenuated or even completely blocked. When the difference between total received power and the direct-path power, which is the PM, is larger than a threshold value, the range measurement is identified as NLoS [2]. The performance of this approach, however, depends highly on the choice of the discrimination threshold value. When UWB sensors are installed well above the ground (for example on shoulders), the threshold used for a PM-based NLoS discriminator is 6 dBm, that is any signal with PM value above 6 dBm is identified as NLoS [2], [110].

3.2.1 Ground Effect

Due to the Fresnel zone effect [111], as shown in Fig. 3.2, the height above the ground at which the UWB sensors are placed affects the range measurement. A Fresnel zone is one of a series of confocal prolate ellipsoidal regions of space around a transmitter and a receiver. The

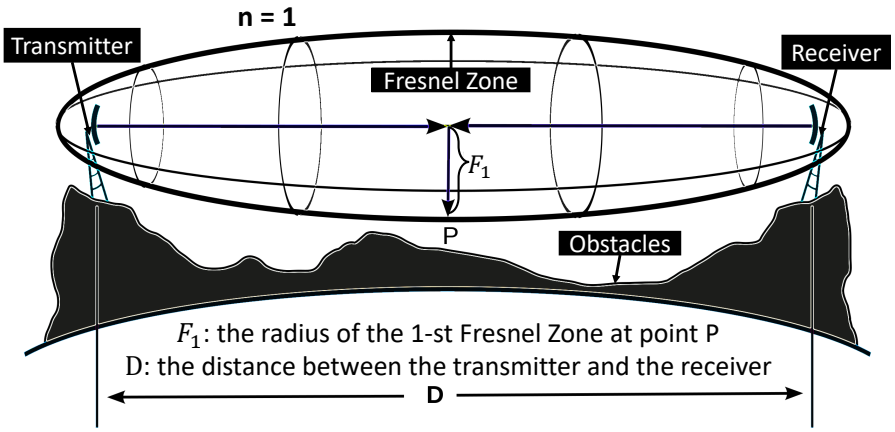


Figure 3.2: The first Fresnel zone.

obstacles inside the Fresnel zone can cause a significant interference in signal propagation between receiver and transmitter. Fresnel zone computations are used to anticipate obstacle clearances. On the other hand, the location of the transmitter and the receiver can be designed to avoid obstacles based on the Fresnel zone. Intuitively, clear LoS between the transmitter and the receiver guarantees the accuracy of UWB range measurements. But, because of the complex nature of the radio waves, obstructions within the first Fresnel zone can cause a significant weakness even if those obstructions are not blocking the apparent LoS signal path. For example, suppose the two UWB sensors are placed in LoS condition but not high enough above the ground and not close enough to each other. Then, the ground can be an obstacle inside of the first Fresnel zone, causing a significant impact on the signal strength. This phenomenon is known as the ground effect on UWB measurements.

Based on the distance between two parallel feet and the frequency of the UWB signal of the Decaware DWM1000 module, we calculated the radius of the first Fresnel zone F_1 according to (3.1)

$$F_1 = 0.5\sqrt{c\,\mathsf{D}/f}\,, \tag{3.1}$$

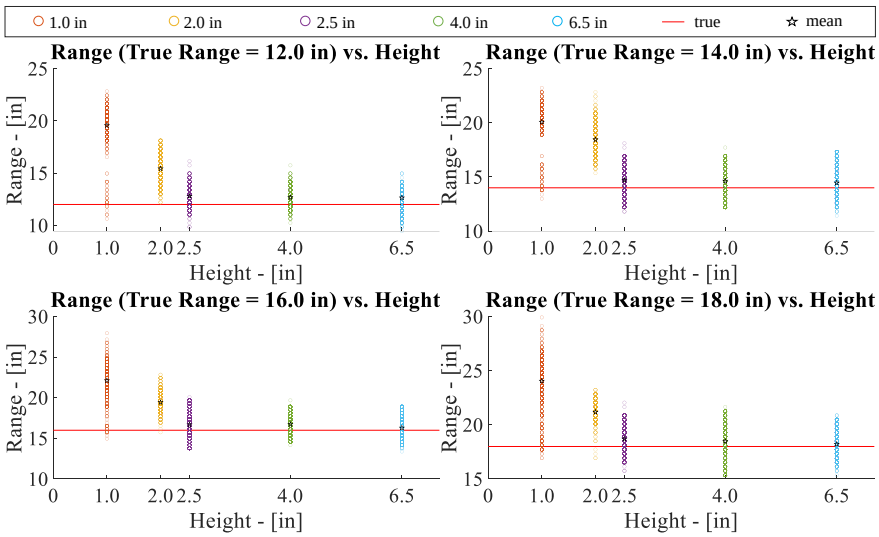
where c is the speed of light, D is the distance between the transmitter and the receiver

mounted on the feet, and f is the frequency of the transmitted signal. Given that the foot-to-foot distance is 12 in (standing parallel), and the frequency of the UWB signal of DWM 1000 sensors is 6.5 GHz, based on (3.1), the theoretical radius of the first Fresnel zone is 0.06 m. Therefore, the two UWB sensors should be placed at least 0.06 m above the ground to avoid the ground effect.

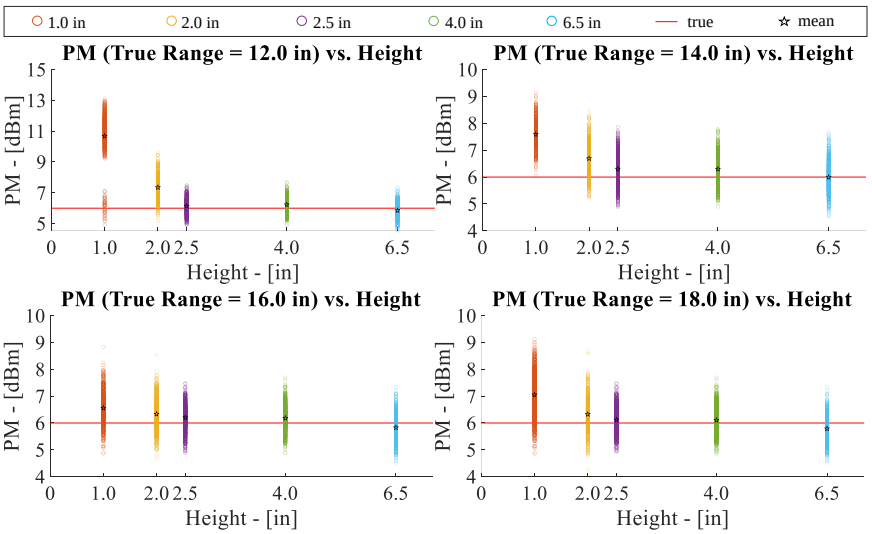
The minimum height obtained by the Fresnel zone criterion is a theoretical value. We tested the effect of the height on the ranging accuracy and the PM value of the DWM1000 UWB sensors by placing two UWB transceivers face-to-face (at 0° angle shown in Fig. 3.4(a)) and at different heights of 1.0, 2.0, 2.5, 4.0, and 6.5 inches and ranges of 12, 14, 16 and 18 inches from each other. The results are shown in Fig. 3.3. Fig. 3.3(a) clearly shows the adverse effect of approaching close to the ground on ranging accuracy, as predicted by the Fresnel zone study. Notice that the recommended height of 0.06 m $\approx$ 2.36 in by the Fresnel zone analysis when UWB sensors are 12 in apart correlates well with the experimental value observed in Fig. 3.3(a). With regards to PM value, we can see in Fig. 3.3(b) that the deterministic threshold of 6 dBm is not respected well when sensors are closer to ground. Only at the height of 6.5 in the average PM value starts to go under 6 dBm. Recall that PM values below 6 dBm indicated LoS measurements. This phenomenon can be resulted from the multipath propagation due to ground effects.

3.2.2 Orientation Effect

Besides the UWB sensors' height from the ground, the relative orientation of the transceivers also affects the accuracy of UWB ranging. To find an optimal relative orientation for the UWB sensors placed on the Lab-on-Shoe, we collected raw measurements at 8 relative angles shown in Fig. 3.4 (a). The UWB sensors were placed at 6.0 in (0.1524 m) above the ground, which is the same height that the UWB sensors are from ground on Lab-on-Shoe platform



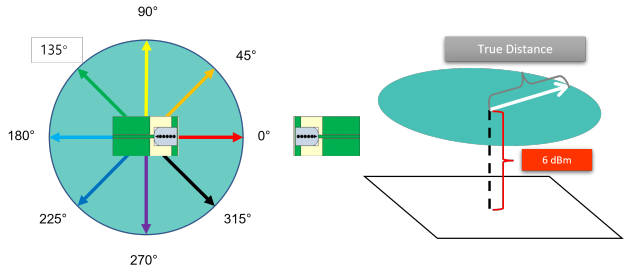
(a) Range vs. Height



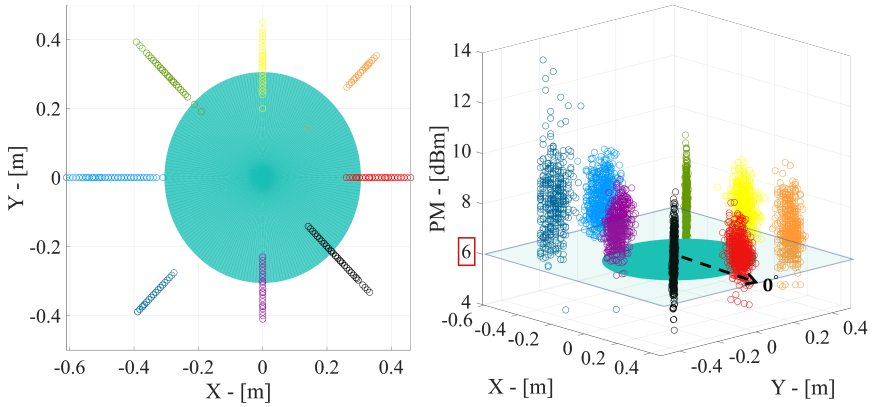
(b) PM vs. Height

Figure 3.3: The range and PM values at different heights and distances. In each height/distance pair, 1000 measurement samples are taken.

and 12.0 in (0.3048 m) from each other. The collected data is visualized in Fig. 3.4 (b). The radius of the solid green circle is the actual distance between the sensors (12.0 in). The sensors' location is shown in the $X - Y$ plane, with one of them placed fixed at the center and the other one placed at different relative angles as shown in Fig. 3.4 (a). The optimal relative orientation to place the UWB sensors is when the empirical mean of the collected measurements is on the perimeter of the green circle (to have zero mean Gaussian



(a) Schematic



(b) Measurement Cloud

Figure 3.4: UWB sensor measurements at various relative angles when they are 12 in apart and 6 in above the ground. (a) is the schematic for the collected data cloud in (b).

distribution). According to the results showing RMSE of the UWB range measurements at different orientations, Fig. 3.4 (b) and Table 3.1, we concluded that the optimal configuration to place the UWB sensors on the Lab-on-Shoe platform is when the sensors are at 270° relative orientation.

On the right-hand side plot in Fig. 3.4 (b), the Z axis denotes the PM, and the green circle is in the 6 dBm plane, which represents the deterministic threshold used in literature [2] to discriminate LoS and NLoS when the sensors are placed well above the ground. This experimental study, given that all the measurements are collected in clear LoS, similar to the results in Fig. 3.3(b), indicates that the PM value of 6 dBm is not the appropriate threshold value for distinguishing LoS and NLoS in lower heights, which is also reported in [112]. To investigate further, we collected a set of measurements in NLoS when we place

Table 3.1: The UWB range measurements RMSE in different orientations.

Angle (°)	0	45	90	135	180	225	270	315
RMSE (in)	3.0	5.5	2.3	4.9	6.6	7.6	1.7	2.4

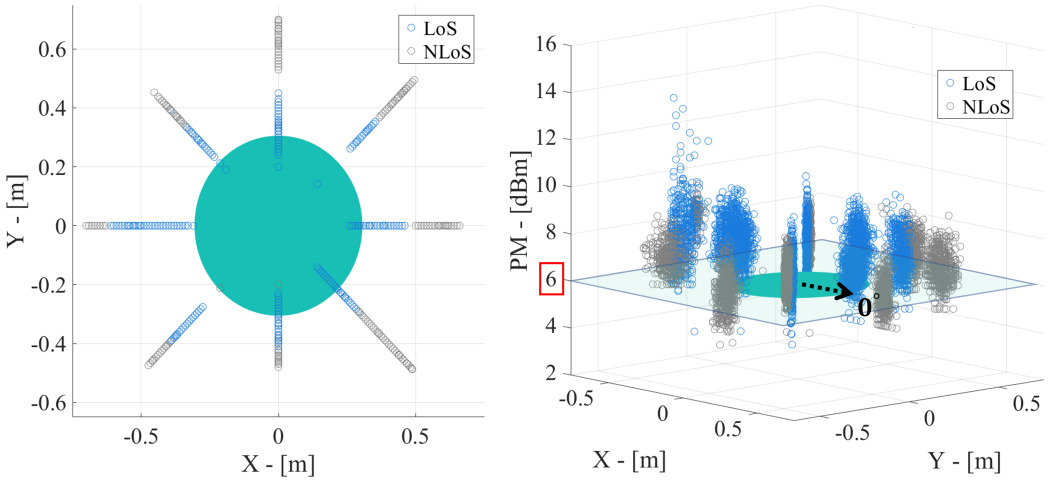


Figure 3.5: The UWB measurements under LoS and NLoS conditions.

a metal plate of 1 in thickness as an obstacle between two feet. The results are shown in Fig. 3.5. As it can be learned from the top view of data, there is a positive bias in the UWB range measurement under NLoS conditions compared to LoS conditions. Once the NLoS is identified, the bias can be removed manually. However, the LoS and NLoS UWB PM measurements are not separable with merely a fixed deterministic threshold according to the 3-D plot. Therefore, our results conclude that a power-based identification based on a fixed PM threshold is not an appropriate measure to identify NLoS measurements at lower heights.

3.2.3 UWB placement effect on localization accuracy

To investigate the effect of UWB sensor placement on localization accuracy of a dual-foot mounted ZUPT-aided INS, we carried out a set of experiments in which a pedestrian walked on a straight line of length 43.5 m, as shown in Fig. 3.6. The experiments were conducted in the Engineering Gateway Building at the University of California, Irvine, using the Lab-on-

Shoe platform (Fig. 3.1) whose foot-mounted UWB sensors were at 6.0 in height from the ground in four different configuration shown in Fig. 3.7. In this setting, there is always a clear path between the two feet, which ensures the LoS condition throughout the experiments. The sampling rates of the IMU and the UWB sensor were set to 1000 Hz and 10 Hz, respectively. Comparing with sensors placement discussed in Section 3.2.2, configuration 1 to 4 corresponds to a relative orientation of 180°, 90°, 0° and 270° between the sensors. We carried out five sets of walks along straight reference trajectory for each configuration. The results of these experiments are shown in Fig. 3.8, which presents the localized trajectories, and Fig. 3.9, which shows the CEPs. The CEP is a measure of precision defined as the median error radius of a circle centered on the true value (the endpoint location in our experiments). As we can conclude from these plots, configuration 4 with CEP of 0.41% of the distance traveled compared to 0.42%, 1.07% and 0.85% of the distance traveled, respectively for configurations 1, 2 and 3, demonstrates the best localization performance. This result is consistent with what we concluded in Section 3.2.2, which indicated the relative orientation of 270° is the optimal configuration when the sensors are placed stationary and the pedestrian stands still with feet parallel. The reader should note that for all four configurations the CEP value reported in Fig. 3.9 normalized by the distance traveled is significantly less than the CEP values of 6.90% and 4.64% of the distance traveled reported in [107] (a walk over a straight line of 53 m) for, respectively, ZUPT aided by ultrasonic foot-to-foot ranging and ZUPT aided by vision-based foot-to-foot ranging.

3.3 Conclusions

This section investigates the effect of the UWB sensor placement for foot-to-foot ranging in dual-foot mounted ZUPT-aided pedestrian inertial navigation. We discuss the factors affecting the UWB measurements, including the height from the ground and the relative

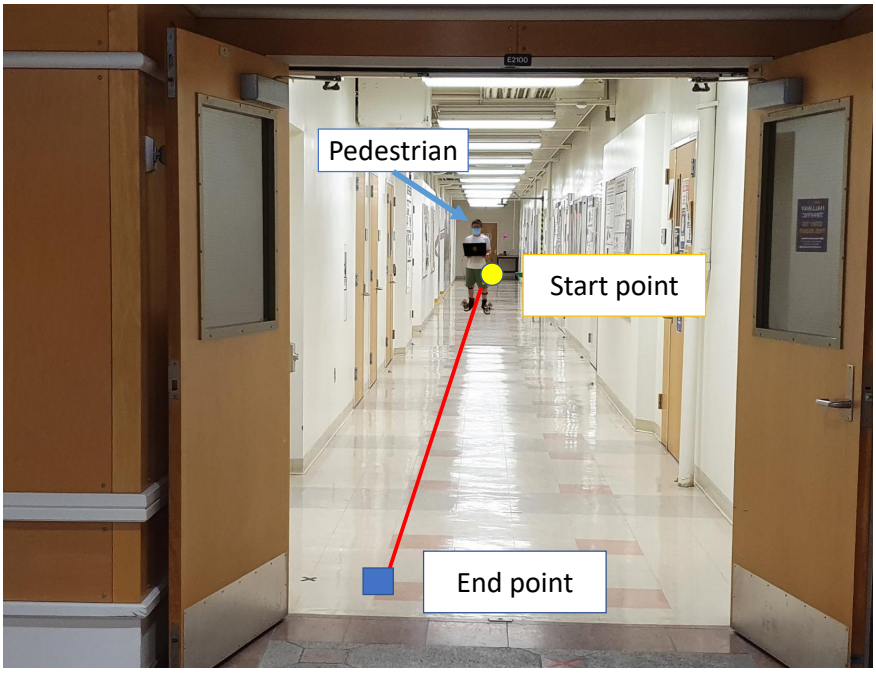


Figure 3.6: The ZUPT-aided pedestrian inertial navigation experiment.

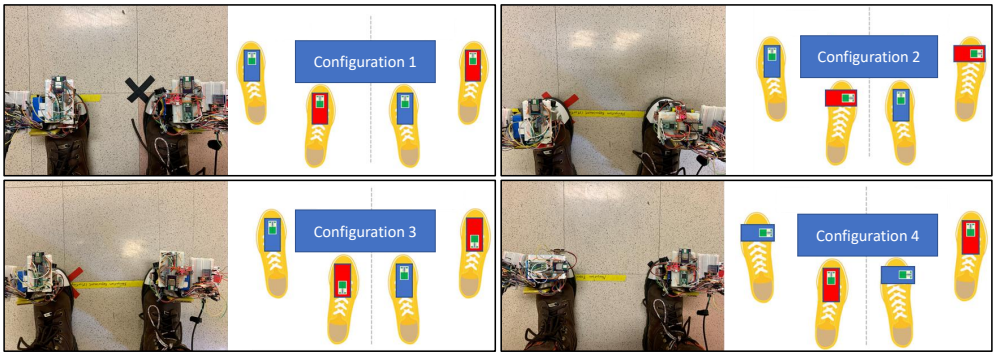


Figure 3.7: Four different UWB sensor placement configurations.

orientation of the UWB sensors. The result showed that the UWB orientation significantly influences the measurement quality. The height also impacts the measurements significantly in lower heights due to the ground effect. In a set of experimental studies, we demonstrate the best placement for the UWB sensors and derived a preferable configuration in the ZUPT-aided pedestrian inertial navigation. Our results also illustrated that the well-known power-based NLoS discriminator that uses a fixed power metric threshold to identify NLoS UWB range measurements is not an appropriate measure at low heights.

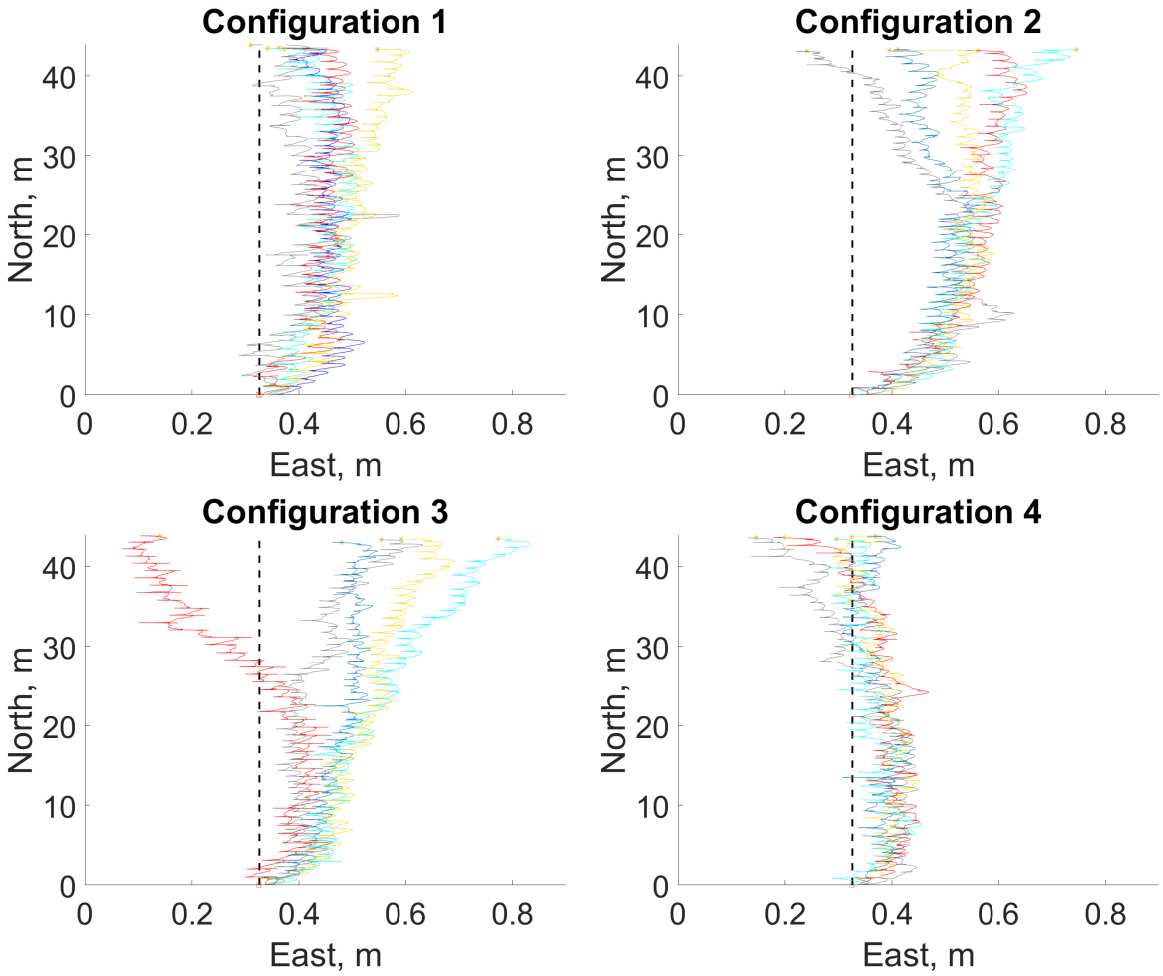


Figure 3.8: The estimated trajectories of the right foot in four configurations. The black dashed line represents the reference trajectory. The red, yellow, light blue, dark blue and gray lines denote the walks 1 to 5, respectively.

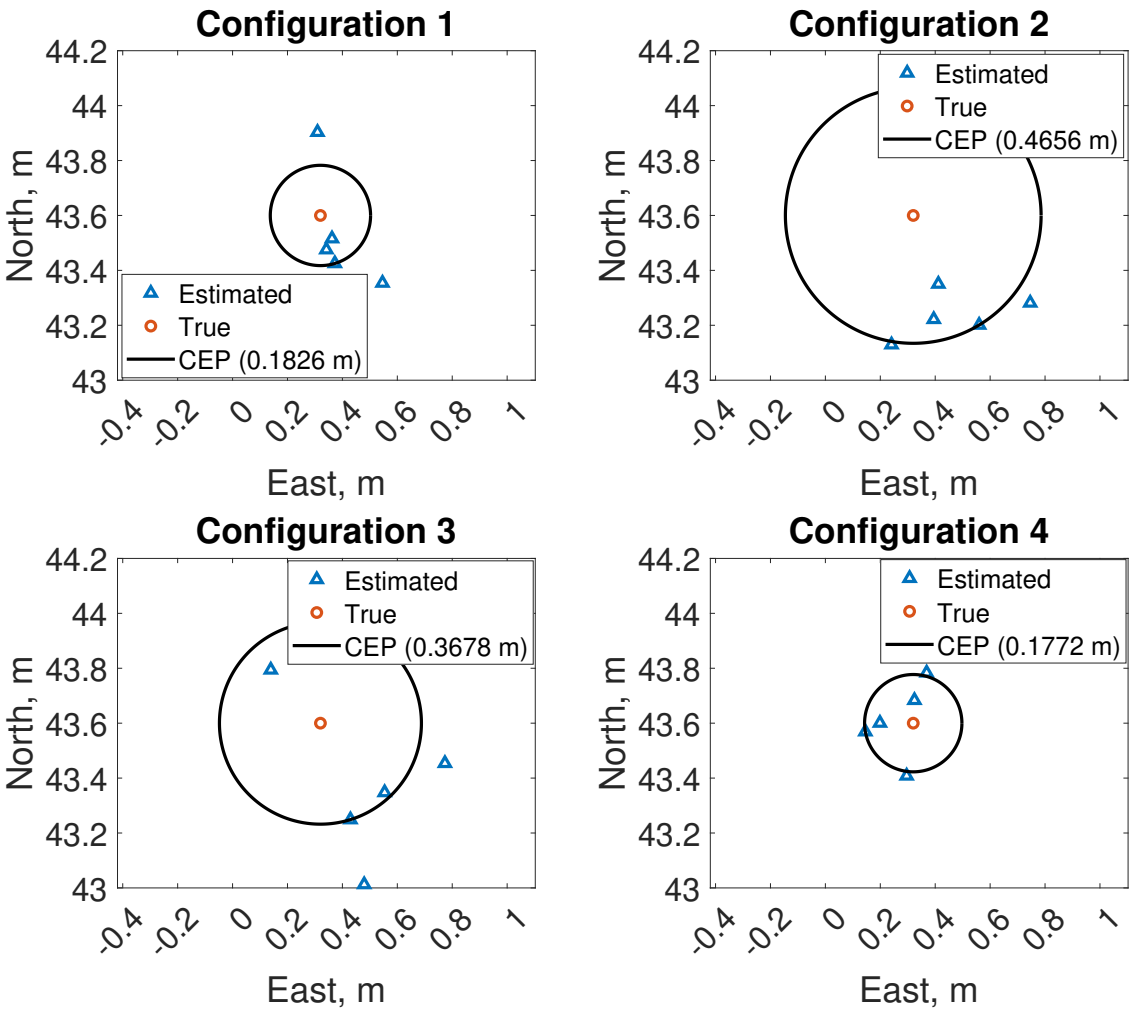


Figure 3.9: The CEP for four configurations.

Chapter 4

UWB LoS and NLoS Identification and Bias Compensation

In this chapter, we further investigate the approaches to increase the accuracy of the UWB ranging via identifying the LoS and NLoS scenarios and compensate the bias accordingly. In our earlier work in navigation filters that processed both LoS and NLoS UWB ranging (see our papers [113] and [114]) we assumed that the LoS and NLoS measurements could be identified and distinguished from each other with exact certainty. In these navigation filters, we used the popular power-based NLoS identification method of [2]. As we illustrate in previous chapter, the working principle of the power-based NLoS identification methods is that in LoS condition, the power of the received direct-path signal takes a big proportion of the total received signal power. In contrast, in NLoS conditions, the direct path is significantly attenuated or even completely blocked. When the difference between total received power and the direct-path power is larger than a threshold value, the range measurement is identified as NLoS [2]. The performance of this approach, however, depends highly on the choice of the discrimination threshold value and our experimental study in previous chapter shows that in practice, the deterministic identification of the UWB ranging mode is not

accurate.

In this dissertation work, we propose a novel bias correction framework called OptiTrack-aided learning-based UWB bias correction (OLUC) for both LoS and NLoS UWB ranging via low-cost UWB transceivers. Our method is depicted in Fig. 4.1. The novel features we use in our ML-based framework are the first path power level (FPPL), the received signal strength (RSS), the power metric (PM), and the raw range measurement, which are readily available on low-cost UWB transceivers such as the popular DWM1000 developed by DecaWave [115]; see Section 4.2.1 for more details. Recognizing that the nature of the bias generation in the LoS and the NLoS modes are different, we take a two-step approach to our bias correction. First, we use an artificial neural network (ANN), referred to as LaNANN (short for LoS and NLoS ANN), to discriminate the LoS and NLoS measurements. Then, we use a set of two ANNs, LoS bias correction ANN (LBCANN), and NLoS bias correction ANN (NBCANN), to learn the bias-free ranges out of the uncorrected UWB range measurements under LoS and NLoS conditions, respectively. We adopt ANN as our learning model because the universal approximation theorem claims that an ANN with enough depth can approximate any continuous function given certain weights. In other words, we can approximate the complex unknown UWB bias model using the ANN and minimize the size of the ANN to reduce the computational complexity.

4.1 Objective statement

Figure 4.2 shows the result of a set of experimental studies that compares the raw UWB range measurements with the true distance between UWB sensors. As seen in Fig. 4.2, the NLoS bias is more significant than the LoS bias. On the other hand, as the distance between the sensors grows the LoS bias also grows due to multi-path effects. Literature shows that the LoS bias can also become significant in short ranges when the sensors are close to the

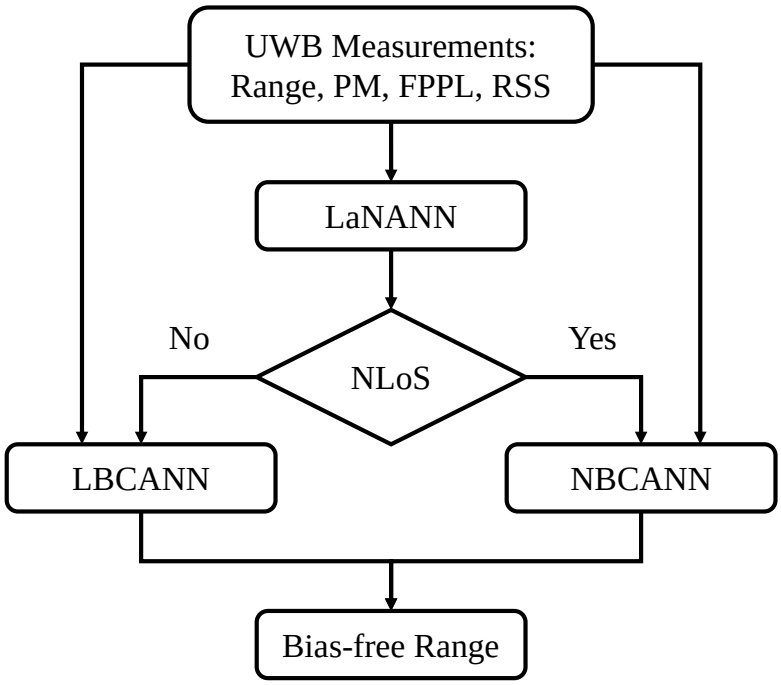


Figure 4.1: The diagram of our learning-based UWB LoS and NLoS classification and bias correction method. PM: Power Metric, FPPL: First Path Power Level, RSS: Received Signal Strength, LaNANN: Line-of-sight and Non-line-of-sight artificial neural network, NLoS: Non-line-of-sight, LBCANN: Line-of-sight bias compensation artificial neural network, LaNANN: Non-line-of-sight bias compensation artificial neural network.

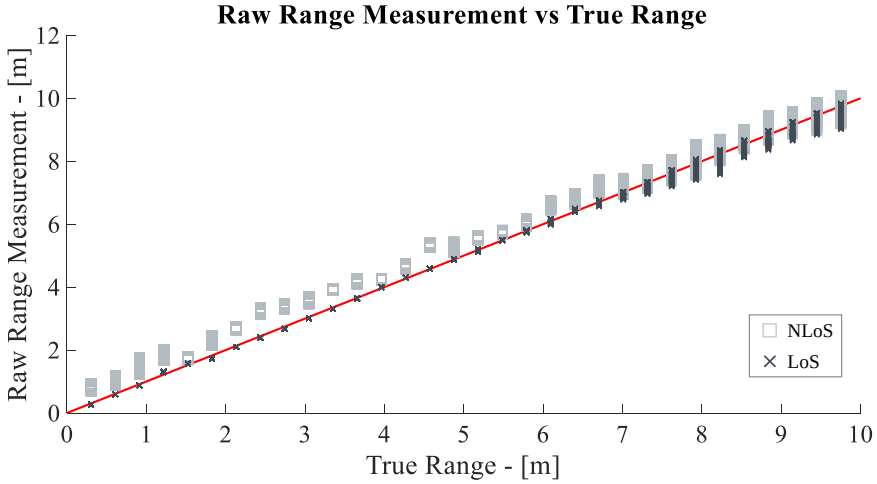


Figure 4.2: The raw UWB range measurements collected at different true distances within 10 m in LoS and NLoS conditions. The measurements are collect by DecaWave DWM1000 UWB sensors described in Section 4.2.3. The red dash-line, regarded as the "ground truth", represents the ideal raw range measurement of the UWB without any error.

grounds due to ground effect; for more details see e.g., [38] where we used UWB sensors for foot-to-foot ranging to improve pedestrian localization using a double-foot-mounded INS.

Bias in the range measurements degrades the ranging accuracy of the localization filters. To observe the effect of the bias, let us consider a beacon-based localization of an UWB-tagged mobile agent using a group of N anchor UWB nodes located at known positions $\{A_1, A_2, \cdots A_N\} \in \mathbf{R}^3$. Denote the position of the tagged agent by $P_{\text{Tag}} \in \mathbf{R}^3$. The agent can communicate with each of the N anchors and compute the corresponding ToF ranges. The measured range with respect to each anchor consists of the actual distance, the measurement noise, and a bias. Thus, the range measurement z_n between the UWB tag and the n-th anchor is computed as

$$z_n = \|A_n - P_{\text{Tag}}\|_2 + \omega + b_n, \tag{4.1}$$

where $\|.\|_2$ is the Euclidean distance, ω is the zero mean white Gaussian noise and

$$b_n = \begin{cases} b_n^{LoS}, & \text{LoS} \\ b_n^{NLoS}, & \text{NLoS} \end{cases}, \quad n = 1, 2, \cdots, N, \tag{4.2}$$

is the measurement bias whose value differs in the LoS and NLoS ranging conditions. The LoS bias is due to various effects such as multi-path, relative orientation, and radiation pattern whose value varies at different actual distances. For the NLoS range measurements, bias is additionally due to the range measurement increment resulted from the NLoS increased transmission time. The recursive least square (RLS) estimator[116] to estimate the tag location $\hat{P}_{\text{Tag}}^k$ when the range z_n^k between the UWB tag and the n-th anchor is obtained at timestamp k is

$$\hat{P}_{\text{Tag}}^k = \hat{P}_{\text{Tag}}^{k-1} + \mathbf{K}^k(z_n^k - \hat{z}_n^k), \tag{4.3}$$

where $\mathbf{K}^k$ is the estimator gain matrix, $\hat{z}_n = \|A_n - \hat{P}_{\text{Tag}}^{k-1}\|_2$ is the predicted range measurement. Given the biased range measurement model in (4.1), the estimation error $\mathbf{e}^k$ of the

RLS estimator at time k is

$$\mathbf{e}^k = P^k_{\text{Tag}} - \hat{P}^k_{\text{Tag}} = (\mathbf{I} - \mathbf{K}^k\mathbf{H}^k)\mathbf{e}^{k-1} - \mathbf{K}^k(b_n + \omega), \tag{4.4}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{H}^k = \frac{\partial \|A_n - P_{\text{Tag}}\|_2}{\partial P_{\text{Tag}}}|_{P_{\text{Tag}}=\hat{P}^{k-1}_{\text{Tag}}}$. On the other hand, he location estimation covariance matrix $\mathbf{\Sigma}^k$ at time k is

$$\mathbf{\Sigma}^k = (\mathbf{I} - \mathbf{K}^k\mathbf{H})\mathbf{\Sigma}^{k-1}. \tag{4.5}$$

As (4.4) and (4.5) show the bias b_n does not directly affect the covariance $\mathbf{\Sigma}^k$ but it aggravates the location estimation error $\mathbf{e}^k$. This means that the bias in the measurements not only increases the error it will also cause inconsistent estimation (overly optimistic estimates), which eventually leads to filter instability and divergence. The objective of this work is to improve the UWB localization filter accuracy and stability by removing the bias from the UWB measurements both in LoS and NLoS measurements using an ML approach.

4.2 Learning-based UWB LoS and NLoS Classification and Bias Correction

This section gives a detailed account of our learning-based bias correction method depicted in Fig. 4.1.

4.2.1 Feature Analysis and Selection

We focus on designing a bias correction method that can be used with low-cost UWB transceivers. In this work, we use the popular UWB transceiver DWM1000 developed by

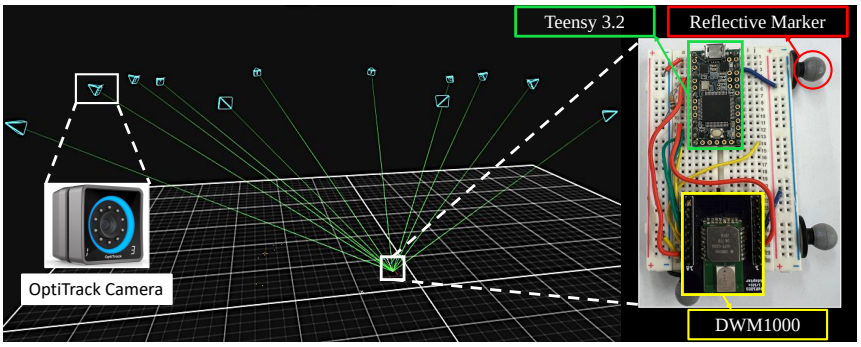


Figure 4.3: The screenshot of tracking the DecaWave DWM1000 UWB sensor attached with reflective markers in the Motive software.

DecaWave [115] whose data acquisition and ranging software runs on a Teensy 3.2 as shown in the right-hand side of Fig. 4.3. The Teensy 3.2 microcontroller implements a serial peripheral interface protocol to communicate with the UWB and collect sensor measurements [117]. To pick the features for measurement classification and correction, we constrained the choice to the signals that are readily available on this UWB ranging sensor.

To distinguish NLoS UWB measurements from LoS ones, [2] employs a PM-based ranging mode discriminator with a deterministic threshold. The motivation for this choice is that in the LoS condition, the received direct-path signal's power takes a big proportion of the total received signal power. In contrast, in the NLoS conditions, the direct path is significantly attenuated or even completely blocked. When the difference between the total received power and the direct-path power is larger than a threshold value, [2] argues that the range measurement can be identified as NLoS. To investigate the effectiveness of this feature, we carried out an experimental study at different distances between two UWB sensors under a controlled environment where we know the true measurement ranges and the measurement mode in terms of LoS and NLoS. The result shown in Fig. 4.4, however, indicates that the deterministic threshold represented by the red line cannot completely separate LoS and NLoS measurements, and there are still significant overlapping PM values.

To obtain a better separation between LoS and NLoS measurements, in addition to the

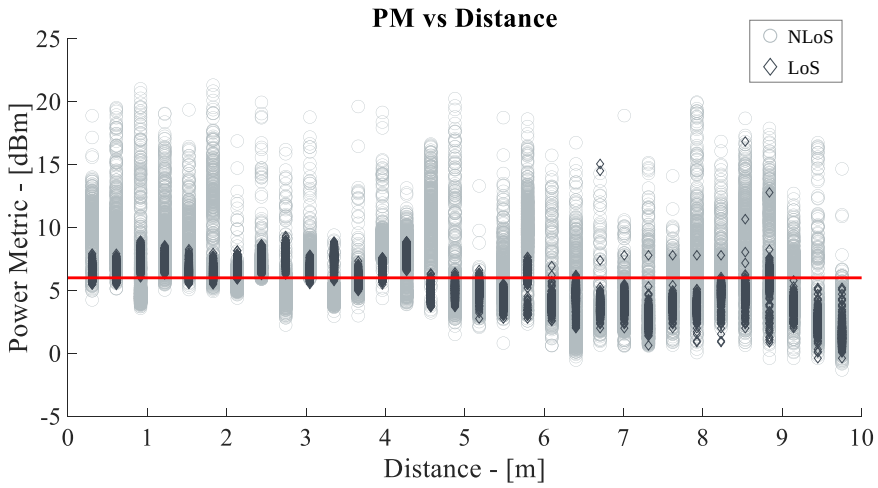


Figure 4.4: The power metric of UWB measurements collected at different distances within 10 m in LoS and NLoS respectively. The red line denotes the deterministic threshold to discriminate LoS and NLoS proposed by [2].

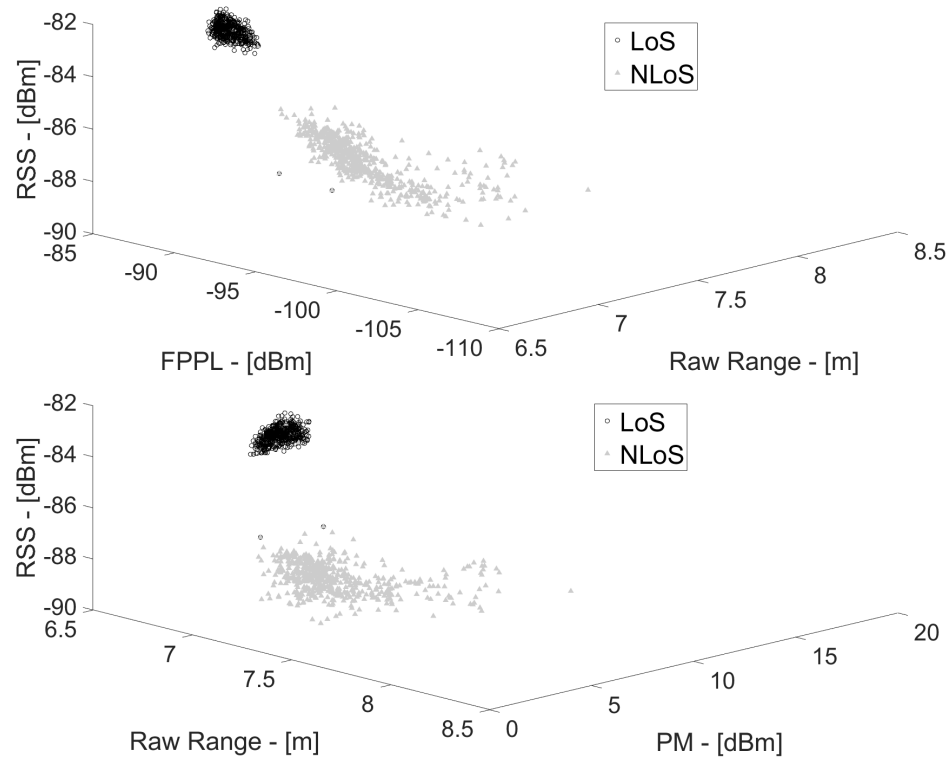


Figure 4.5: The 3-D plot of UWB measurements when true distance between the sensors is 7 m.

PM, we also included the FPPL and RSS of UWB signals and the raw range measurements (uncorrected) as features. For a fixed distance, e.g., 7 m, the 3-D visualization of the UWB measurements under controlled data collection is shown in Fig. 4.5. As seen in this plot, using

these features the LoS and NLoS measurements are more clearly separated from one another. As we discussed earlier, the relative pose of the transceivers can also affect the ranging bias and thus recent work like [59] considered relative bias as an input feature. However, since relative pose is hard to obtain without additional sensors, we did not consider it as a direct feature. Relative pose affects FPPL and RSS. In our learning approach, the effect of relative pose on the bias is captured implicitly by the use of diverse training data that is collected in various relative poses between ranging sensors. That is, we allow the ANNs to learn the effect of the relative pose on the bias from its implicit application on the other features i.e., PM, FPPL, and RSS. Our evaluation study for localizing a moving pedestrian in Section 4.3 shows that the use of diverse training data results in in a bias correction method that delivers high-accuracy localization results.

4.2.2 ANN architecture design

In general, the LoS and NLoS scenarios of the UWB measurement can be modeled as a random variable θ with a Bernoulli distribution mathematically whose probability mass function is

$$f(\theta; p) = \begin{cases} p, & \text{if } \theta = \text{LoS} \\ 1 - p, & \text{if } \theta = \text{NLoS} \end{cases} \tag{4.6}$$

The probability of LoS p, as well as the bias-free range $\tilde{z}_n = \|A_n - P_{\text{Tag}}\|_2 + \omega$, based on our selection of the features, are functions of range, RSS, FPPL and PM, respectively.

$$p = f_{\text{LaN}}(\text{range}, \text{RSS}, \text{FPPL}, \text{PM}), \tag{4.7a}$$

$$\tilde{z}_n = f_{\text{BC}}(\text{range}, \text{RSS}, \text{FPPL}, \text{PM}). \tag{4.7b}$$

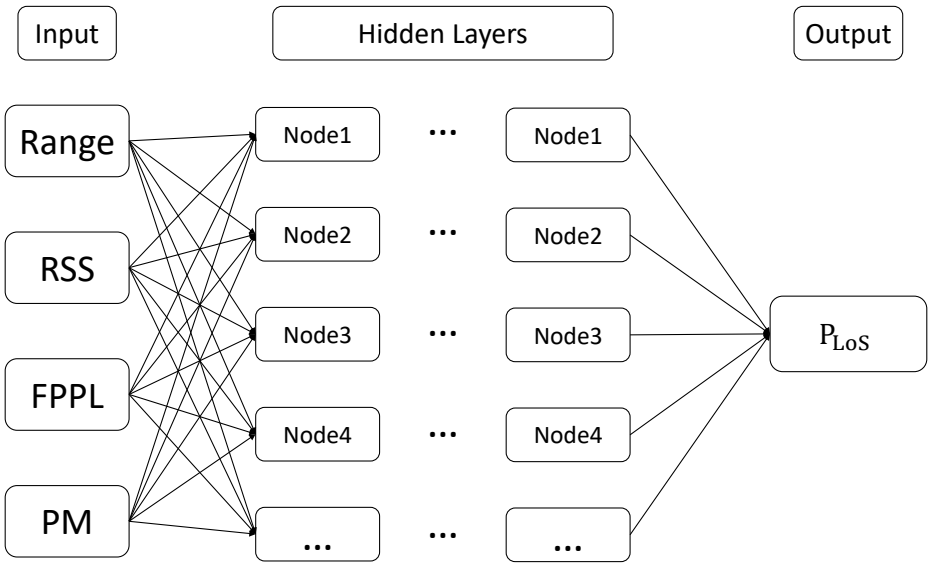


Figure 4.6: The ANN model for UWB LoS and NLoS Classification. The ANN structure for learning the range bias looks similar except that the output layer represents the bias-free range.

Then our objective boils down to use ANNs to approximate these functions without explicitly knowing the exact mathematical relations, that is, we will approximate them with

$$\hat{p} = \hat{f}_{\text{LaNANN}}(\text{range}, \text{RSS}, \text{FPPL}, \text{PM}), \tag{4.8a}$$

$$\hat{\tilde{z}}_n = \begin{cases} \hat{f}_{\text{LBCANN}}(\text{range}, \text{RSS}, \text{FPPL}, \text{PM}), & \text{if } \hat{p} \geq 0.5 \\ \hat{f}_{\text{NBCANN}}(\text{range}, \text{RSS}, \text{FPPL}, \text{PM}), & \text{if } \hat{p} < 0.5 \end{cases}. \tag{4.8b}$$

where $\hat{f}_{\text{LaNANN}}$, $\hat{f}_{\text{LBCANN}}$ and $\hat{f}_{\text{NBCANN}}$ denote, respectively, the generic functions of the feed-forward LaNANN, LBCANN and NBCANN of our method shown in Fig. 4.1. Figure 4.6 depicts the LaNANN component, which classifies the measurement type. It uses four nodes in the input layer corresponding to the UWB measurements of range, RSS, FPPL, and PM. The output layer has one node for the LoS probability. The NLoS probability is obtained via $1 - \hat{p}$. We use the same input for the two ANNs of LoS and NLoS bias correction in parallel while their output layers have one node of the bias-free UWB range.

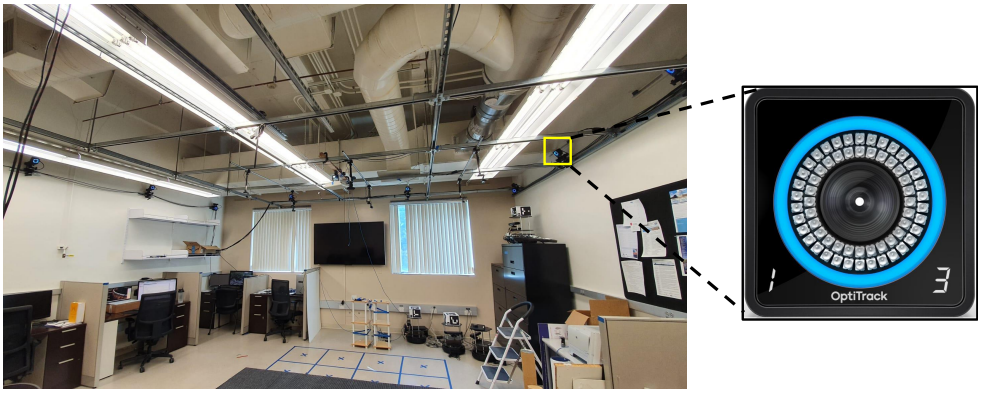


Figure 4.7: The OptiTrack motion capture system in KCS lab at University of California Irvine. The cameras have 1280×1024 resolution with a 120 Hz native frame rate.

4.2.3 OptiTrack-aided Training Data Generation

The success of ML-based methods depends on extensive and informative training data generation, otherwise, the out-of-sample performance cannot be guaranteed due to the distributional mismatch. An important factor of the bias in the measurements is the relative pose between the UWB ranging sensors. As mentioned in Section 4.2.1, we propose to incorporate the relative pose implicitly in the measurements. Therefore, we have to make our training data set contain as many relative poses as possible. Collecting training data manually at various relative poses is not feasible since there are infinite possibilities of relative poses. To obtain truly diverse and rich training data in terms of relative poses, we use the OptiTrack motion capture camera system to obtain an accurate bias-free range dynamically by taking measurements between a moving tagged agent and a set of stationary UWB beacons.

The OptiTrack system is a precise motion capture and 3-D tracking system with a localization error of less than 0.2 mm via optical ranging and trilateration for numerous applications, e.g., video game design, virtual reality, and robotics. The OptiTrack system in our Kia Cooperative Systems (KCS) lab at University of California Irvine (UCI) is shown in Fig. 4.7. The cameras can precisely perceive the particular reflective markers' locations via visual-based ranging techniques. As such, we attach at least 3 reflective markers on the UWB

sensors so that they form a rigid body that can be recognized and tracked by the OptiTrack system as shown in Fig. 4.3. The pivot point of the UWB rigid body defined by the reflective markers can be manually assigned and tracked. Consequently, we can obtain the accurate position of the UWB sensors by setting their antennas as the pivot points and then calculate precisely the distances between the pivot points as the label values, i.e., bias-free ranges, for the LBCANN and NBCANN.

To generate a diverse and abundant training data set, an agent holding a tag UWB sensor that connected to three anchor UWB sensors walked on random paths under the OptiTrack system in the LoS condition in our KCS lab for 5 minutes. The agent also waved the tag UWB in the air to create different relative poses. All collected data were labeled by LoS. For NLoS cases, we put various obstacles between the tag and the anchors and repeated the same data collection procedure as for the LoS data collection. The OptiTrack-aided training data generation experiment is illustrated in Fig. 4.8. By collecting data in such a dynamic setting, we enrich the training data set to compensate for the lack of explicit knowledge of the relative poses and thus mitigate the out-of-sample issue to a considerable extent.

4.2.4 Training Result

The UWB device we used to generate the training data is DecaWave DWM1000 and the sampling rate was set to 10 Hz. We generated 300000, 150000, and 150000 samples of the UWB range, PM, FPPL, RSS, and bias-free range for the LaNANN, LBCANN, and NBCANN, respectively, generated by data collection experiments described in the previous section. The samples are split into a training set, a validation set and a test set according to the ratio of 70%, 15%, and 15%. For the LaNANN, the label is LoS (1) and NLoS (0), while that of LBCANN and NBCANN is the bias-free range. We used the grid search method [118, 119] to tune the hyperparameters of all ANNs during the training process. The

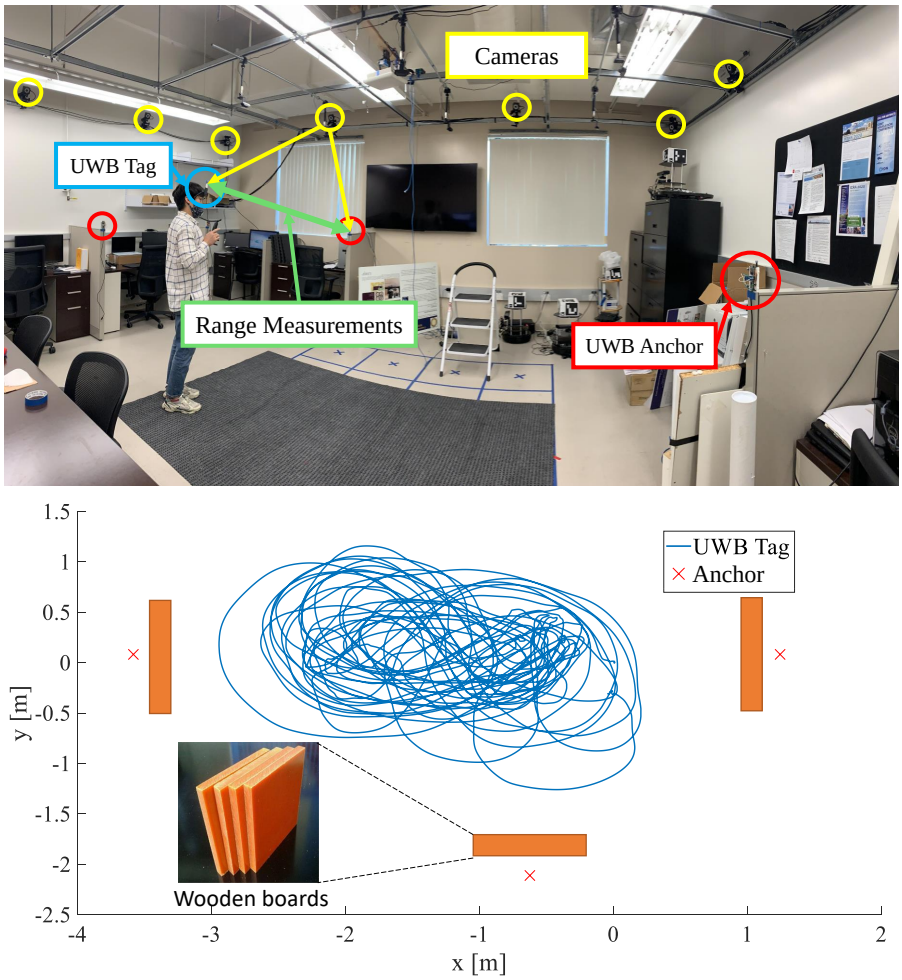


Figure 4.8: The OptiTrack-aided training data generation experiment where the agent holding a tag UWB sensor walks along a random trajectory. The reference position of all UWB sensors are obtained by the OptiTrack system. The blue line denotes the trajectory of the tag UWB sensor while the red crosses represent the anchors. For NLoS data collection, multiple obstacles such as wooden boards (shown as the orange rectangles), books, foams, human body, and other items seen in the lab are placed to block the direct path between the tag and the anchors.

Hyperparameters	LaNANN	LBCANN	NBCANN
Number of hidden layers	2	3	3
Number of units per layer	10	11	18
Loss	Cross-Entropy	MSE	MSE
Activation function	Sigmoid	ReLU	ReLU
Optimizer	Adam	Adam	Adam
Batch size	256	512	512
Learning rate	0.05	0.025	0.025

Table 4.1: The hyperparameter design of the ANNs.

fine-tuned design of all three ANNs is shown in Table 4.1.

	Training	Validation	Test
LaNANN (Accuracy: %)	94.39	94.12	92.73
LBCANN (RMSE: m)	0.1308	0.1316	0.1351
NBCANN (RMSE: m)	0.1557	0.1563	0.1662

Table 4.2: The summary of training result for all ANNs.

In summary, the training results are shown in Table. 4.2 where the best classification accuracy in the test sets achieved by LaNANN is 92.73%. Also, the root mean square error (RMSE) for the test set for LBCANN and NBCANN is 0.1351 m and 0.1662 m, respectively. The training period took about 2 hours in total. The prediction run time for LaNANN, LBCANN and NBCANN is 0.0022 ms, 0.0029 ms and 0.0041 ms using a Dell Laptop: Intel Core^{TM} i5-1135G7@4.20GHz, quad-core, 8 GB memory.

4.3 Experimental Evaluations

In this section, we evaluate the effectiveness of our proposed OLUC method through a set of pedestrian localization experiments using the RLS method described in Section II. To demonstrate the positive impact of using a training data set that is collected in a diverse set of relative poses, we compare the performance of the RLS-based pedestrian localization in two parallel processes: in one we use our proposed OLUC bias correction and in the other we use a bias correction method that we call "Stationary-ANN". Stationary-ANN has the same structure of the OLUC method as shown in Fig. 4.1, but it has been trained using a different data set that was collected in a limited number of stationary configurations where the true distances were measured by human using measuring tape. We also compare the effectiveness of our proposed OLUC method to the existing OptiTrack-aided method proposed in [63], which used an ANN method to fuse IMU/UWB-based localization results with Optitrack position measurements to improve localization accuracy. We denoted this method as "ANN-Fusion". All the ANNs of the OLUC and the Stationary-ANN method used

in the experimental studies described below were trained using the data collected in the KCS lab at UCI, whereas the ANN-Fusion used the position data collected at the experimental locations according to [63].

To show the generality of the proposed OLUC bias correction method, we conducted our experimental studies at two different unseen locations, i.e., no data from these locations was used in training the ANNs used in the experiments. The first location was the Public Safety Immersive Test Center (PSITC), which is a collaborative facility between the First Responder Network Authority and the National Institute of Standards and Technology's Public Safety Communications Research Division, located in Boulder, Colorado. The second place is in an office space in the CALIT2 building at UCI.

4.3.1 Experimental Evaluation at the first Location: PSITC

In this experiment, three UWB anchors were placed at known locations at the PSITC facility. A pedestrian carrying a tag UWB sensor, which maintained communications with the three UWB anchors, walked along three different trajectories: straight line, rectangle, and lemniscate (figure-eight shape), for about five to six minutes, as depicted in Fig. 4.9. The walks were repeated three times for each trajectory. Due to the blockage by human bodies, roadblocks and walls in the testing venue, the measurement scenarios occasionally switched between the LoS and the NLoS modes. We used the OptiTrack system installed in the PSITC facility to obtain high-accuracy reference trajectories for performance comparison. The reader should note that the reference trajectories were only used for the evaluation of our bias correction methods that were trained based on the data collected in the KCS lab. The experimental results are shown in Fig. 4.10 and Fig. 4.11.

As seen in Fig. 4.10, the RLS localization which uses our proposed OLUC method outperforms the Stationary-ANN and ANN-Fusion methods. The error plots, as well as the

	w/o Bias Correction	OLUC	ANN-Fusion
Loop-Closure Error (m)	0.8046	0.2643	1.5020

Table 4.3: The loop-closure error of the trajectory using the proposed OLUC method versus the ANN-Fusion.

performance metrics in Fig. 4.11, show that our method achieved the substantial bias reduction rates of 66.91%, 64.66%, and 63.13%, respectively in the three trajectories that we used. Note that the Stationary-ANN has an inferior performance due to the lack of diversity in its training data set (out-of-sample problem), which we discussed earlier. As for the ANN-Fusion method of [63], it has slightly worse performance compared to our proposed OLUC in this experimental setting but the reader should notice that this method only applies at the facilities that a motion capture system like OptiTrack exists because we used the accurate position data provided by the OptiTrack to train the ANN of the ANN-Fusion method in this particular location. Therefore, it does not posses the wide generality that our OLUC method offers, as it is shown in the next experimental study.

4.3.2 Experimental Evaluation at the second Location: CALIT2

To verify the generalization of our proposed OLUC method as well as the ANN-Fusion method against unseen data (new location), we additionally conducted a set of localization experiments in an office space in the CALIT2 building at UCI, where the OptiTrack system is not available and the LoS/NLoS conditions are more complicate due to the cubicles in the environments as shown in Fig. 4.12. The experimental settings remained the same as the previous case, except the designed trajectory and the environment (no OptiTrack and more complicate LoS/NLoS conditions). Note that in this extra experiment, all methods were tested via a new trajectory on which they were not trained. Figure 4.13 and Table 4.3 illustrate the additional comparison results. We used the loop-closure error for the mobile UWB tag instead of the RMSE since its start and end point were identical and the reference

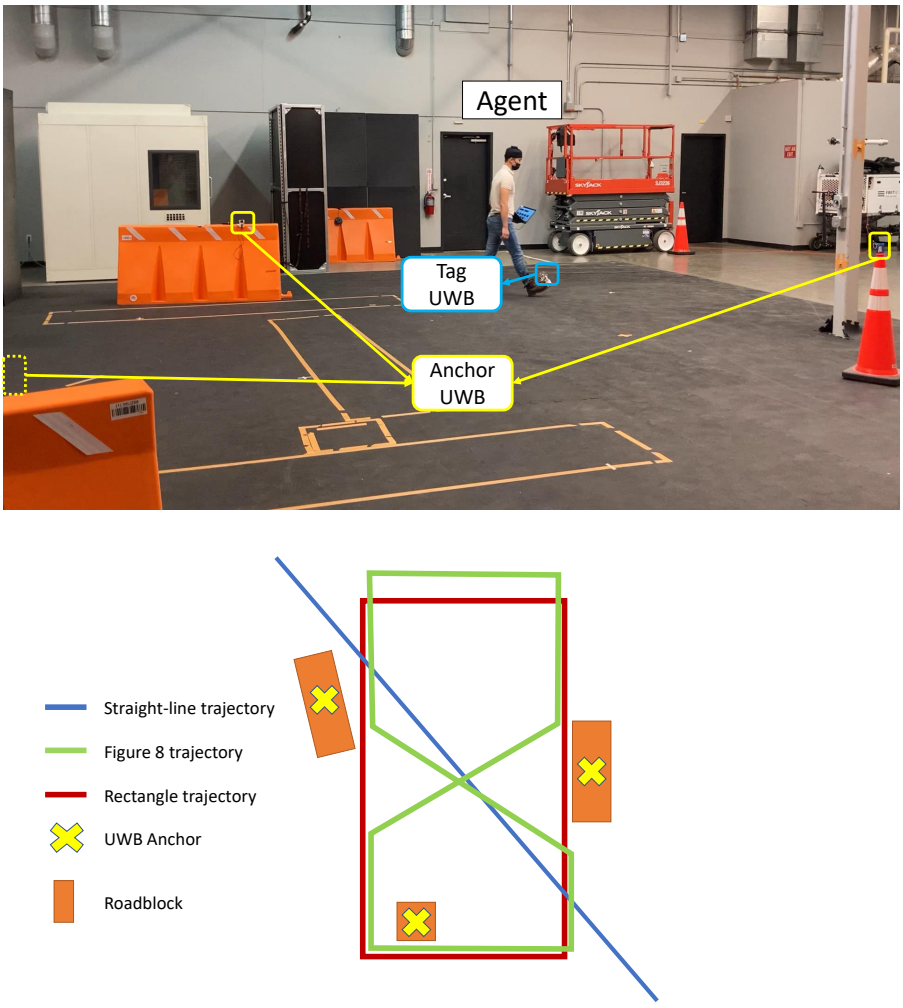
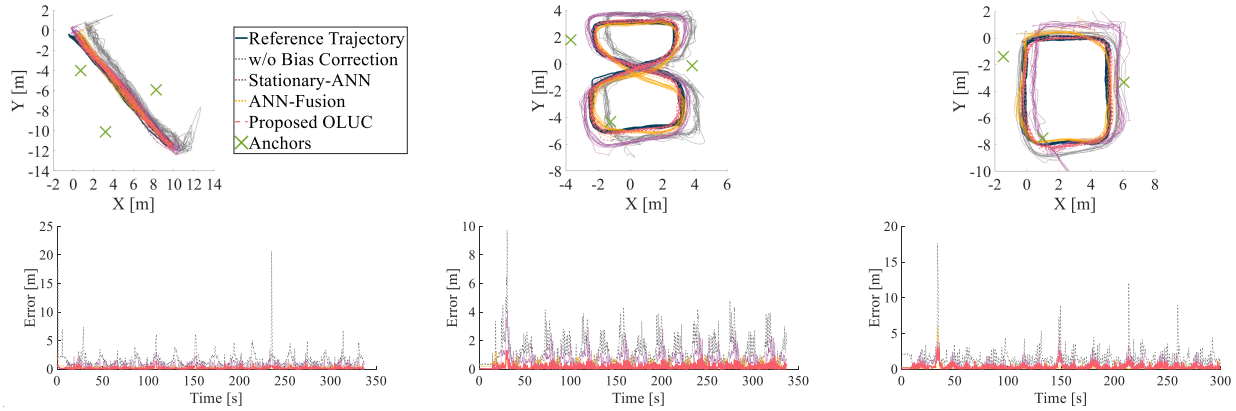


Figure 4.9: The demonstration experiments conducted in the PSITC facility associated with the bottom diagram that shows the designed trajectories of the mobile UWB tag, the positions of the anchors, and of the obstacle roadblocks.

trajectory was measured by human labor and thus was less accurate. As we can see, without the available OptiTrack and re-training the ANN (the ANN of the ANN-Fusion method was trained in the first location, PSITC), the ANN-Fusion method delivers poor localization results compared to our proposed OLUC method whose ANNs were only initially trained using the data collected in the KCS lab.

We can find the evidence in the second comparison studies from both Fig. 4.13 showing the trajectories and Table 4.3 comparing the loop-closure errors that the ANN-Fusion performs even worse than using the raw measurements without the bias correction which indicates the



(a) Straight Line (b) Figure 8 (c) Rectangle

Figure 4.10: The estimated trajectories of the agent using RLS localization method as well as the localization error plots without (w/o) bias correction (using raw measurements) and after applying different bias correction methods, i.e., Stationary-ANN method, ANN-Fusion and the proposed OLUC.

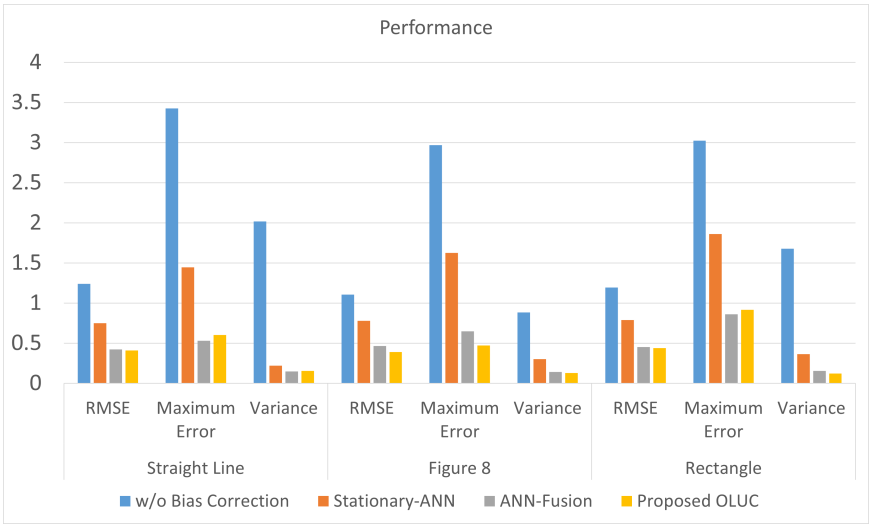


Figure 4.11: The average experimental statistics of the different trajectories using different bias correction methods.

ANN of the ANN-Fusion failed and cannot be generalized to other locations. On the contrary, our proposed method still performed well. It can be concluded that the big constraint of the ANN-Fusion is that it works only when the OptiTrack is available or the training and the evaluation trajectories are the same. Because it necessarily needs to train the ANN based on the estimated position obtained by both the UWB sensors and the OptiTrack rather than

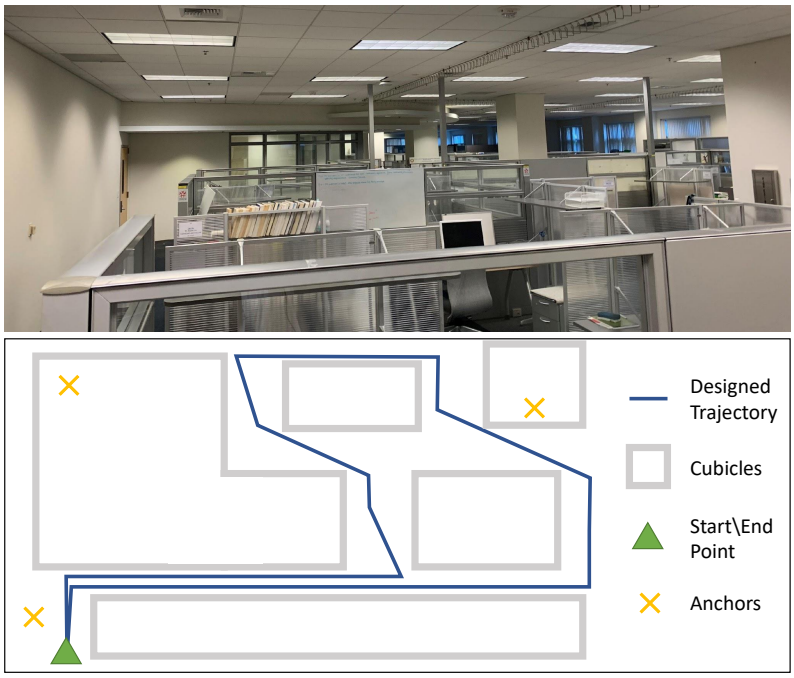


Figure 4.12: The demonstration experiments conducted in the cubicle environment in the CALIT2 building at UC Irvine associated with the bottom diagram that shows the designed trajectories of the mobile UWB tag, the start and end point, the positions of the anchors, and of the cubicle configurations.

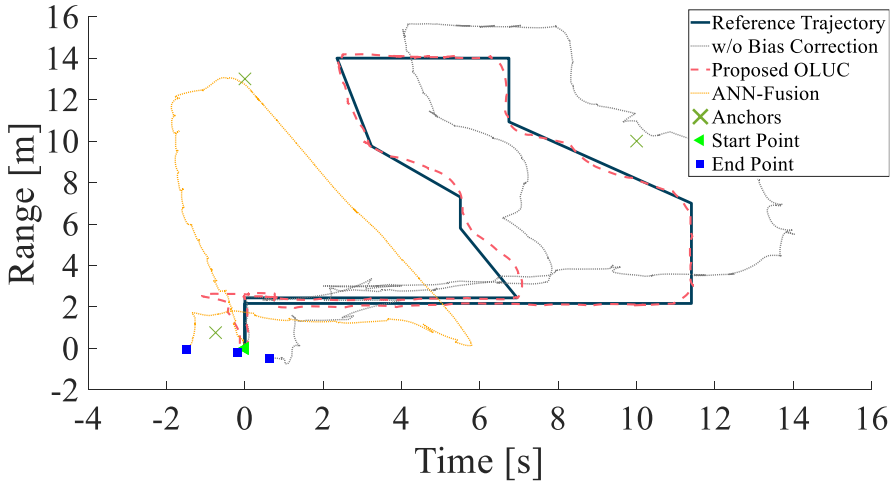


Figure 4.13: The estimated trajectory of the mobile UWB tag using the proposed OLUC method and the ANN-Fusion.

the UWB signal, which makes the generalization performance not satisfactory. Therefore, the ANN-Fusion method is only a position enhancement that uses and is highly dependent on the OptiTrack. If the OptiTrack is not available and the designed trajectory is not seen by the ANN, it fails easily. Conversely, our proposed OLUC learns the bias from the signal

which is comparatively robust against the change of locations and does not highly rely on the OptiTrack, since the OptiTrack is only used to generate training data. The evaluation using the OptiTrack is not a necessity.

4.4 Conclusions

UWB ranging bias correction has a significant impact on the localization accuracy of the UWB-based navigation algorithms. In this dissertation work, we addressed the problem of UWB ranging bias correction in complex localization scenarios by training neural network models to distinguish the LoS and NLoS scenarios and predict the corresponding bias-free range in real-time using features that are readily available on low-cost UWB sensors. Our experimental data collection resulted in identifying a novel set of features, raw range, PM, FPPL and RSS, to train ANNs for measurement type classification and bias-free range prediction. The effectiveness of any learning-based solution depends on the diversity and informativeness of its training data. Recognizing that the relative orientation of the ranging sensors affects the ranging accuracy, we used an Optitrack motion capture system to collect high-accuracy ranging between a tag UWB sensor on a moving agent and fixed UWB anchor nodes. In doing so, we collected a diverse set of data in various relative poses between the sensors. The effectiveness of our OptiTrack-aided supervised learning-based UWB bias correction method was demonstrated via a set of pedestrian localization experiments using the RLS algorithm for location estimation in two different locations. Our experiments showed that our method leads to a considerable improvement in the localization accuracy.

Chapter 5

Cooperative Localization Using Learning-based Constrained Optimization

To reduce the computational cost of the consistent implicit track-to-track fusion and CL methods, we propose using machine learning (ML) based solvers to solve the optimization problems used in these methods. We demonstrate our approaches for solving the optimization problems in the DMV and the PECMV methods for CL proposed in [48]. We evaluate the effectiveness of our proposed approaches using a set of experiments for pedestrian localization using an INS aided by CL. In these experiments, we observe that the run time of the proposed approaches compared with DMV and the PECMV is reduced 53.85% and 99.98% with no sacrifices on the localization accuracy and estimation consistency of CL filters. Notice that our proposed ML base optimizer is an approach that can be easily extended to the loosely-coupled estimation algorithms reviewed above as well as covariance union (CU) and arithmetic average (AA) density fusion [120].

ML-based solutions for optimization problems have been successfully used in the applications such as combinatorial optimization [81], wireless network optimization [82], and supply chain management [83]. ML-based approaches have also been used in CL problems but for different purposes. For example, [84] uses a deep neural network (DNN) to assist a CL for vehicular networks, where DNN is designed to solve the chronic nonlinear approximation problem. Or, [85] uses an ML-based surrogate model as a measurement scheduling merit function.

5.1 Problem Definition

Consider a group of N mobile agents with communication and computation capabilities. The equation of motion of each agent $i \in \mathcal{V} = \{1, \cdots, N\}$ at time step $t \in \mathbb{Z}^+$ is described by

$$\mathbf{x}^i(t) = \mathbf{f}(\mathbf{x}^i(t-1), \mathbf{u}^i_m), \quad \mathbf{x}^i \in \mathbb{R}^{n^i}, \tag{5.1}$$

where $\mathbf{x}^i(t) \in \mathbb{R}^{n^i}$ is the state of agent i (e.g., position, velocity, attitude) and $\mathbf{u}^i_m = \mathbf{u}^i(t) + \boldsymbol{\nu}^i_u$ is the self-motion measurement command obtained, e.g., from odometry or IMU. Here, $\boldsymbol{\nu}^i_u$ is the self-motion measurement noise. Each agent uses a local filter to obtain an estimate of its own state $\hat{\mathbf{x}}^{i-}(t) \in \mathbb{R}^{n^i}$ and its corresponding error covariance matrix $\mathbf{P}^{i-}(t) \in \mathbb{S}^{++}_{n^i}$ at each timestep $t \in \mathbb{Z}^+$ using its motion model and occasional access to absolute measurements through e.g. GPS or measurement from known landmarks. Here, $\mathbb{S}^{++}_n$ is the set of positive definite matrices of size n. We call $\text{bel}^{i-}(t) = (\hat{\mathbf{x}}^{i-}(t), \mathbf{P}^{i-}(t))$ the prior belief of agent i at time t. In this context, $\mathbb{S}^{++}_n$ represents the set of positive definite matrices of size n. We refer to $\text{bel}^{i-}(t) = (\hat{\mathbf{x}}^{i-}(t), \mathbf{P}^{i-}(t))$ as the prior belief of agent i at time t.

Due to the contaminating noise in the self-motion measurements, the localization accuracy of the agents degrades during the mission horizon. If access to absolute measurements to correct dead-reckoning based localization is limited, to bound the error and improve accuracy, CL

via joint processing of *occasional* relative measurements among two agents is used. Let the relative measurement (e.g., relative range, relative bearing, relative pose, or a combination of them) $\mathbf{z}_j^i(t)$ obtained by agent i from agent j at time t be

$$\mathbf{z}_j^i(t) = \mathbf{h}(\mathbf{x}^i(t), \mathbf{x}^j(t)) + \boldsymbol{\nu}^i(t), \quad \mathbf{z}_j^i \in \mathbb{R}^{n_z^i} \tag{5.2}$$

where $\boldsymbol{\nu}^i(t) \in \mathcal{N}(\mathbf{0}, \mathbf{R}^i(t))$ is the zero mean Gaussian measurement noise with covaraince matrix $\mathbf{R}^i(t) \in \mathbb{S}_{n_z^i}^{++}$.

When no inter-agent measurement is available to update the local belief, the updated belief is set to be the propagated belief, i.e., $\text{bel}^{i+}(t) = \text{bel}^{i-}(t) = (\hat{\mathbf{x}}^{i-}(t), \mathbf{P}^{i-}(t))$. Otherwise, the local belief is corrected using the fusion approaches as briefly outlined below. For more information, see [48]. To simplify the notation, hereafter we only include the time index t when the clarification is needed.

5.1.1 Relative measurement processing via DMV method in the absence of explicit knowledge of the inter-agent cross-covariances

Let the joint belief of the agents i and j be $\text{bel}_J^-(t) = (\hat{\mathbf{x}}_J^-(t), \mathbf{P}_J^-(t))$, where

$$\hat{\mathbf{x}}_J^-(t) = \begin{bmatrix} \hat{\mathbf{x}}^{i-}(t) \\ \hat{\mathbf{x}}^{j-}(t) \end{bmatrix}, \quad \mathbf{P}_J^-(t) = \begin{bmatrix} \mathbf{P}^{i-}(t) & \mathbf{P}_{ij}^-(t) \\ \mathbf{P}_{ij}^-(t)^\top & \mathbf{P}^{i-}(t) \end{bmatrix}. \tag{5.3}$$

When agent i takes a relative measurement from agent j, agent i can correct its local belief using the measurement feedback $\mathbf{z}_j^i(t) - \hat{\mathbf{z}}_j^i(t)$, where $\hat{\mathbf{z}}_j^i(t)$ is the estimated relative measurement based on agent i and j's prior beliefs. When the cross-covariance term $\mathbf{P}_{ij}^-(t)$ is known, the feedback gain can be computed from an Extended Kalman like update procedure. DMV

algorithm aims to process the relative measurement $\mathbf{z}_j^i(t)$ to correct the local beliefs in the absence of explicit knowledge of the cross-covariance $\mathbf{P}_{ij}^-(t)$, see [48]. In the DMV method, similar to the joint estimate interpretation of the data fusion problem in [121], a discorrelated upper bound is used to account for any unknown cross-covariance term $\mathbf{P}_{ij}^-(t)$ as

$$\begin{bmatrix} \mathbf{P}^{i-}(t) & \mathbf{P}_{ij}^-(t) \\ \mathbf{P}_{ij}^-(t)^\top & \mathbf{P}^{j-}(t) \end{bmatrix} \leq \begin{bmatrix} \frac{1}{\omega}\mathbf{P}^{i-}(t) & \mathbf{0} \\ \mathbf{0} & \frac{1}{1-\omega}\mathbf{P}^{j-}(t) \end{bmatrix}, \ \omega \in [0,1]. \tag{5.4}$$

Then, the DMV algorithm updates the propagated belief of agent i at time t, $\text{bel}^{i-}(t)$, according to

$$\hat{\mathbf{x}}^{i+}(t) = \hat{\mathbf{x}}^{i-}(t) + \bar{\mathbf{K}}^i(\omega_\star^i)\,(\mathbf{z}_j^i(t) - \hat{\mathbf{z}}_j^i(t)), \tag{5.5a}$$

$$\mathbf{P}^{i+}(t) = \bar{\mathbf{P}}^i(\omega_\star^i), \tag{5.5b}$$

where $\bar{\mathbf{K}}^i(\omega) = \frac{\mathbf{P}^{i-}}{\omega}\mathbf{H}_i^{i\top}\big(\mathbf{H}_i^i\frac{\mathbf{P}^{i-}}{\omega}\mathbf{H}_i^{i\top} + \mathbf{H}_j^i\frac{\mathbf{P}^{j-}}{1-\omega}\mathbf{H}_j^{i\top} + \mathbf{R}^i\big)^{-1}$, and $\bar{\mathbf{P}}^i(\omega) = \big(\omega(\mathbf{P}^{i-})^{-1} + (1-\omega)\mathbf{H}_i^{i\top}(\mathbf{H}_j^i\mathbf{P}^{j-}\mathbf{H}_j^{i\top} + (1-\omega)\mathbf{R}^i)^{-1}\mathbf{H}_i^i\big)^{-1}$, with $\mathbf{H}_i^i = \partial\mathbf{h}(\hat{\mathbf{x}}^{i-}, \hat{\mathbf{x}}^{j-})/\partial\mathbf{x}^i$ and $\mathbf{H}_j^i = \partial\mathbf{h}(\hat{\mathbf{x}}^{i-}, \hat{\mathbf{x}}^{j-})/\partial\mathbf{x}^j$. The optimal ω, denoted by $\omega_\star^i$, is obtained from the optimization problem (5.6)

$$\omega_\star^i = \underset{0\leq\omega\leq 1}{\text{argmin}} \ \log\det\bar{\mathbf{P}}^i(\omega). \tag{5.6}$$

According to [48, Theorem 3.1], despite the unknown $\mathbf{P}_{ij}^-(t)$, the DMV update is guaranteed to be no worse than the local belief of the agents.

5.1.2 Relative measurement processing via PECMV in the absence of explicit knowledge about the inter-agent cross-covariance

Since the upper bound of DMV on the joint covariance matrix accounts for all the possible values for the unknown cross-covariance $\mathbf{P}^{-}_{ij}(t)$, DMV produces conservative updates. To reduce the conservatism of DMV, the PECMV [48] proposes an alternative way in which the unknown cross-covariance $\mathbf{X}$ in the joint covariance matrix

$$\mathbf{P}^{-}_{J}(\mathbf{X}) = \begin{bmatrix} \mathbf{P}^{i-} & \mathbf{X} \\ \mathbf{X}^{\top} & \mathbf{P}^{j-} \end{bmatrix} \tag{5.7}$$

is estimated from the following optimization problem

$$\mathbf{X}^{\star} = \arg\max_{\mathbf{X}} \det \begin{bmatrix} \mathbf{I}_{n^i} \\ \mathbf{0} \end{bmatrix}^{\top} (\mathbf{P}^{-}_{J}(\mathbf{X})^{-1} + \mathbf{H}^{i\top}\mathbf{R}^{i-1}\mathbf{H}^{i})^{-1} \begin{bmatrix} \mathbf{I}_{n^i} \\ \mathbf{0} \end{bmatrix}, \tag{5.8a}$$

$$\text{subject to } \begin{bmatrix} \mathbf{P}^{i-} & \mathbf{X} \\ \mathbf{X}^{\top} & \mathbf{P}^{j-} \end{bmatrix} > 0. \tag{5.8b}$$

where $\mathbf{I}_{n^i}$ is the identity matrix. PECMV estimates $\mathbf{P}^{-}_{ij}$ by obtaining $\mathbf{X}^{\star}$ that provides the most conservative updated covariance. It is shown in [48] that this optimization problem can be cast in an equivalent convex matrix optimization with linear inequality (LMI) constraints and solved by CVXOPT [122], a Python software for convex optimization.

Once $\mathbf{X}^{\star}$ is obtained from (5.8), the PECMV updated belief $\text{bel}^{i+}_{\text{PECMV}}(t)$ =

$(\hat{\mathbf{x}}^{i+}_{\text{PECMV}}(t), \mathbf{P}^{i+}_{\text{PECMV}}(t))$ for agent i is

$$\hat{\mathbf{x}}^{i+}_{\text{PECMV}} = \hat{\mathbf{x}}^{i-} + \mathbf{K}^{i}_{\text{PECMV}}\,(\mathbf{z}^{i}_{j} - \hat{\mathbf{z}}^{i}_{j}), \tag{5.9a}$$

$$\mathbf{P}^{i+}_{\text{PECMV}} = \begin{bmatrix} \mathbf{I}_{n^i} \\ \mathbf{0} \end{bmatrix}^{\top} (\mathbf{P}^{-}_{J}(\mathbf{X}^{\star})^{-1} + \mathbf{H}^{i\top}\mathbf{R}^{i-1}\mathbf{H}^{i})^{-1} \begin{bmatrix} \mathbf{I}_{n^i} \\ \mathbf{0} \end{bmatrix} \tag{5.9b}$$

where $\mathbf{K}^{i}_{\text{PECMV}} = \begin{bmatrix} \mathbf{I}_{n^i} & \mathbf{0} \end{bmatrix} \mathbf{P}^{-}_{J}(\mathbf{X}^{\star})\mathbf{H}^{i\top}_{i}(\mathbf{H}^{i}_{i}\mathbf{P}^{-}_{J}(\mathbf{X}^{\star})\mathbf{H}^{i\top}_{i} + \mathbf{R}^{i})^{-1}$. The PECMV update satisfies $\mathbf{P}^{i+}_{\text{PECMV}}(t) \leq \mathbf{P}^{i-}(t)$ and $\mathbf{P}^{i+}_{\text{PECMV}}(t) \leq \mathbf{P}^{i+}_{\text{DMV}}(t)$.

5.1.3 Objective statement

The optimization problems (5.6) and (5.8) used respectively in DMV and PECMV algorithms can be viewed as the the following functions

$$\omega^{i}_{\star} = f_{\text{DMV}}(\mathbf{P}^{i-}, \mathbf{P}^{j-}, \mathbf{H}^{i}_{i}, \mathbf{H}^{i}_{j}), \tag{5.10a}$$

$$\mathbf{X}^{\star} = f_{\text{PECMV}}(\mathbf{P}^{i-}, \mathbf{P}^{j-}, \mathbf{H}^{i}_{i}, \mathbf{H}^{i}_{j}). \tag{5.10b}$$

Evaluating these functions to obtain the desired outputs is computationally expensive and time-consuming. Our objective is to use an ML approach to learn these functions and circumvent solving constrained optimization problems required to evaluate the function values.

5.2 LDMV and LPECMV

The universal approximation theorem claims that a neural network (NN) with enough depth can approximate any continuous function given certain weights [123]. If one perceives f_{DMV} and f_{PECDMV} as a continuous function, then they can be approximated by NNs similar to

the one in Fig.5.2. But learning these functions cannot be carried out in a trivial manner. $\omega_\star^i$ and $\mathbf{X}^\star$, the output of f_{DMV} and f_{PECDMV}, are each constrained values, i.e., $\omega_\star^i \in [0,1]$, and $\mathbf{X}^\star$ should be computed such that the joint covariance matrix (5.7) is positive definite. As we explain below, we ensure $\omega_\star^i \in [0,1]$ by appropriate choice of a proper activation functions for the output layer of NN representation of f_{DMV}. To ensure positive definiteness of joint covaraince matrix, however, we need to take further actions when designing an NN model of f_{PECDMV}. The main tool aiding us is the following result. Here, recall that a matrix $\mathbf{M} \in \mathbb{R}^{p\times q}$ is a strict contraction matrix if and only if $\|\mathbf{M}\|_2 < 1$, where $\|.\|_2$ is the 2-norm [124].

Lemma 5.2.1 (c.f. [124, page 207 and page 350])**.** *Let* $\mathbf{A} \in \mathbb{R}^{p\times p}$ *and* $\mathbf{B} \in \mathbb{R}^{q\times q}$*, and* $\mathbf{X}\in\mathbb{R}^{p\times q}$ *be given. Then, the joint matrix* $\left[\begin{smallmatrix}\mathbf{A} & \mathbf{X}\\ \mathbf{X}^\top & \mathbf{B}\end{smallmatrix}\right]$ *is positive definite if and only if* $\mathbf{A}$ *and* $\mathbf{B}$ *are positive definite and there is a strict contraction matrix* $\mathbf{M}$ *such that* $\mathbf{X} = \sqrt{\mathbf{A}}^\top \mathbf{M} \sqrt{\mathbf{B}}$*.*

Invoking Lemma 5.2.1, to ensure positive definiteness of the joint covariance matrix (5.7), we can write

$$\mathbf{X} = \sqrt{\mathbf{P}^i}^\top \mathbf{C} \sqrt{\mathbf{P}^j} \tag{5.11}$$

and require that $\|\mathbf{C}\|_2 < 1$. Thus, to obtain $\mathbf{X}^\star$ using an NN we will learn

$$\mathbf{C}^\star = \left(\sqrt{\mathbf{P}^i}^\top\right)^{-1} \mathbf{X}^\star \left(\sqrt{\mathbf{P}^j}\right)^{-1}, \mathbf{C}^\star \in \mathbb{R}^{n^i\times n^j} \tag{5.12}$$

i.e., the function we want to learn is

$$\mathbf{C}^\star = f_{\text{PECMV}}(\mathbf{P}^{i-}, \mathbf{P}^{j-}, \mathbf{H}_i^i, \mathbf{H}_j^i).$$

For the learning process, all the matrices are flatten into vectors, meaning that the matrices are learned in an element-wise manner. We carry out a supervised learning in which we collect

the labeled data by solving the optimization problem (5.6) to obtain $\omega_\star^i$ and optimization problem (5.8) followed by (5.12) for $\mathbf{C}^\star$. The input and the corresponding label values are

$$input = \mathsf{vectorization}(\mathbf{P}^{i-}, \mathbf{P}^{j-}, \mathbf{H}_i^i, \mathbf{H}_j^i), \tag{5.13a}$$

$$label_{\mathrm{DMV}} = \omega_\star^i, \tag{5.13b}$$

$$label_{\mathrm{PECMV}} = \mathsf{vectorization}(\mathbf{C}^\star). \tag{5.13c}$$

The feed-forward result of the $\mathbf{C}^\star$ prediction network is reconstructed to have the same shape as the $\mathbf{C}^\star$ matrix and used to compute the predicted $\mathbf{X}^\star$.

For the ω prediction network, we set the activation function of the output layer to the standard logistic function (Sigmoid) $S(\chi) = \frac{1}{1+\mathrm{e}^{-\chi}}$ which maps $\mathbb{R} \to [0, 1]$. This choice naturally constraints the learned $\omega_\star^i$, denoted as $\hat{\omega}_\star^i$, to $[0, 1]$. Enforcing the learned $\mathbf{C}^\star$, denoted as $\hat{\mathbf{C}}^\star$, to be strict contraction matrix, i.e., $\|\hat{\mathbf{C}}^\star\|_2 < 1$ is not straightforward and requires a careful selection of the activation functions for the output layer. To enforce $\|\hat{\mathbf{C}}^\star\|_2 < 1$, we implement three methods. In the first method, we add a barrier function [125] into the loss function for training the NN as shown below,

$$\mathcal{L}(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{M} \sum_{m=1}^{M} \Big(\|\mathbf{Y}_m - \hat{\mathbf{Y}}_m\|_2 + \lambda \log(1 - \|\hat{\mathbf{Y}}_m\|_2) \Big), \tag{5.14}$$

where M is the number of the total data points, $\mathbf{Y}_m \in \mathbb{R}^{n^i n^j \times 1}$ is the m-th vector of label values, $\hat{\mathbf{Y}}_m \in \mathbb{R}^{(n^i \times n^j) \times 1}$ is the m-th feed-forward prediction of the network and $\mathbf{B} = \lambda \log(1 - \|\hat{\mathbf{Y}}_m\|_2)$ is the barrier function with the barrier parameter λ selected properly [125]. It is observed from (5.14), when $\hat{\mathbf{Y}}_m$ approaches 1, the value of the barrier function approaches infinity, which prevents the solution from violating the inequality constraint $\|\hat{\mathbf{Y}}_m\|_2 < 1$ for all $m = 1, 2, \cdots, M$. For the output layer activation function, we use hyperbolic tangent

function (Tanh), $T(\chi) = \frac{e^{\chi}-e^{-\chi}}{e^{\chi}+e^{-\chi}}$ to constraint every output to $[-1, 1]$. Given the norm relations [124]

$$\|\hat{\mathbf{C}}^{\star}\|_2 \leq \|\hat{\mathbf{C}}^{\star}\|_{\mathrm{F}} = \|\hat{\mathbf{Y}}_m\|_2 < 1, \tag{5.15}$$

where $\|.\|_{\mathrm{F}}$ denotes the Frobenius norm, the barrier $\mathbf{B} = \lambda \log(1 - \|\hat{\mathbf{Y}}_m\|_2)$ constraints the output of the NN to respect the constraint $\|\hat{\mathbf{C}}^{\star}\|_2 < 1$ for the training data but there is no guarantee for non-training data. The potential distributional mismatch between the training and non-training data can lead to inconsistent performance or even unsafe execution [126]. In the second method, given (5.15) and $\|\hat{\mathbf{C}}^{\star}\|_{\mathrm{F}} = \sqrt{\sum_{i=1}^{n^i} \sum_{j=1}^{n^j} |\hat{c}_{ij}^{\star}|^2}$, we choose the activation function of the output layer to be Tanh normalized by $D = \sqrt{n^i \times n^j}$, i.e., $T(\chi) = \frac{1}{D} \frac{e^{\chi}-e^{-\chi}}{e^{\chi}+e^{-\chi}}$. Since the constraint $\|\hat{\mathbf{C}}^{\star}\|_2 < 1$ is embedded in the NN, we drop the barrier function from the loss function (5.14). Our second method enforces the constraint $\|\hat{\mathbf{C}}^{\star}\|_2 < 1$ in a hard way, but comes with some degree of conservatism, as it limits the ranges of the elements of matrix $\hat{\mathbf{C}}^{\star}$. For cases that $n^i = n^j = n$, we propose an alternative method that enforces the constraint $\|\hat{\mathbf{C}}^{\star}\|_2 < 1$ and it also comes with lower computational complexity. For this third method, we approximate $\mathbf{C}^{\star}$ with a diagonal matrix, i.e., $\mathbf{C}^{\star} \approx \mathbf{C}^{\star}_{\mathrm{Diag}} = \mathrm{Diag}(\boldsymbol{\rho}^{\star})$, where $\boldsymbol{\rho}^{\star} = [\rho_1^{\star}, \rho_2^{\star}, \cdots, \rho_n^{\star}]$ and to enforce $\|\hat{\mathbf{C}}^{\star}\|_2 < 1$ we set the output layer activation functions to Tanh function. We set our objective to find a $\mathbf{C}^{\star}_{\mathrm{Diag}}$ that results in minimum value for $\|\mathbf{C}^{\star} - \hat{\mathbf{C}}^{\star}\|_F$. Given the definition of the Frobenius norm, the off-diagonal elements of $\mathbf{C}^{\star}$ does not play a role in choosing $\mathbf{C}^{\star}_{\mathrm{Diag}}$. Therefore, our loss function is $\mathcal{L}(\mathbf{Y}, \bar{\mathbf{Y}}) = \frac{1}{M} \sum_{m=1}^{M} \|\mathbf{Y}_m - \bar{\mathbf{Y}}_m\|_2$ where $\bar{\mathbf{Y}}_m$ is the vectorized form of diagonal elements of training $\mathbf{C}^{\star}$s.

We train three NNs summarized in Table 5.1, each for one of the methods we discussed to enforce $\|\hat{\mathbf{C}}^{\star}\|_2 < 1$ and test each well-trained NN on a test data set. We evaluate the performance of each proposed method by comparing the value of the objective function (5.8a) with (5.11) computed using the prediction of each NN, i.e., $\hat{\mathbf{C}}^{\star}$ to the optimal objective function value attained by $\mathbf{C}^{\star}$, and we subtract the optimal value from the value of each

Methods	Output layer activation function	Loss function
1	$T(\chi)=\frac{e^{\chi}-e^{-\chi}}{e^{\chi}+e^{-\chi}}$	$\mathcal{L}(\mathbf{Y},\hat{\mathbf{Y}})=\frac{1}{M}\sum_{m=1}^{M}\Big(\|\mathbf{Y}_m-\hat{\mathbf{Y}}_m\|_2+\lambda\log(1-\|\hat{\mathbf{Y}}_m\|_2)\Big)$
2	$T(\chi)=\frac{1}{D}\frac{e^{\chi}-e^{-\chi}}{e^{\chi}+e^{-\chi}}$	$\mathcal{L}(\mathbf{Y},\hat{\mathbf{Y}})=\frac{1}{M}\sum_{m=1}^{M}\Big(\|\mathbf{Y}_m-\hat{\mathbf{Y}}_m\|_2\Big)$
3	$T(\chi)=\frac{e^{\chi}-e^{-\chi}}{e^{\chi}+e^{-\chi}}$	$\mathcal{L}(\mathbf{Y},\bar{\mathbf{Y}})=\frac{1}{M}\sum_{m=1}^{M}\|\mathbf{Y}_m-\bar{\mathbf{Y}}_m\|_2$

Table 5.1: Summary of the three NNs using three methods, where $D=\sqrt{n^i\times n^j}$.

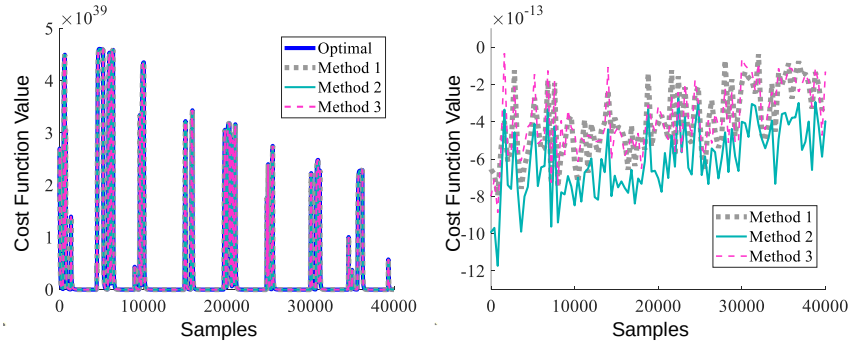


Figure 5.1: The objective function value associated with the optimal $\mathbf{C}^\star$ and $\hat{\mathbf{C}}^\star$ obtained from different learning methods (left) and the difference between each learning method and the optimal one, respectively (right).

learning method to show the difference. The result is shown in Fig. 5.1.

As depicted in Fig.5.1, all proposed learning methods perform less optimally comparing to the original optimization (5.8) as expected, but the differences are insignificant. Also, the information loss when approximating matrix $\mathbf{C}^\star$ with a diagonal matrix in the third method is negligible while normalizing the output value in the second method degrades the optimality further since the hard constraint imposed by the normalized activation function restricts conservatively the feasible set for the optimization of NN weights, which may exclude the optimal NN weights in the corresponding unconstrained optimization, resulting in a prediction that performs less optimally than others. Given the possibility of $\hat{\mathbf{C}}^\star\geq 1$ in the first method, we therefore adopt the third method, i.e., learning a diagonal approximation $\mathbf{C}^\star_{\text{Diag}}$ of matrix $\mathbf{C}^\star$ to build the NN to learn the matrix constrained optimization. The label value of the $\mathbf{C}^\star_{\text{Diag}}$ prediction network is $\boldsymbol{\rho}^\star$, the diagonal of $\mathbf{C}^\star_{\text{Diag}}$, instead of (5.13c). The prediction $\hat{\mathbf{C}}^\star_{\text{Diag}}$ is used to calculate the predicted $\mathbf{X}^\star$. Figure 5.2 shows the NN structure that we used

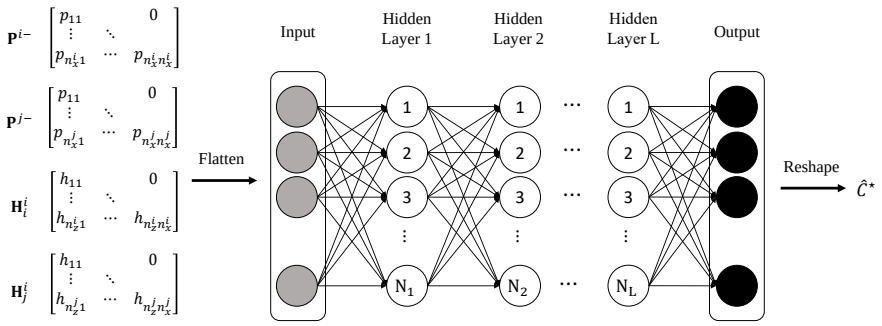


Figure 5.2: The NN structure for learning $\mathbf{C}^\star$ of PECMV. The structure for learning $\omega^\star \in [0, 1]$ of DMV looks similar except that at the output layer we have a single neuron.

to learn the outputs of f_{DMV} and f_{PECMV}. The NN regression we use for learning-based DMV (LDMV) and learning-based PECMV (LPECMV) is fast and suitable for real-time application since the training process is conducted off-line despite the large amount of the training data collected with a high sampling rate. The training and the hyperparameter fine-tuning process for the NNs for the experimental demonstration is presented in Section 5.3.

5.3 Experimental Result

We generated the training data for LDMV and LPECMV via a set of CL-aided pedestrian inertial navigation experiments implementing DMV and PECMV, which were conducted in the Calit2 building at the University of California, Irvine (UCI) and the Firstnet building of the National Institute of Standards and Technology (NIST) in Colorado. Two agents with shoe-mounted IMUs and UWB sensors, shown in Fig. 5.3, walked along different trajectories, e.g., rectangles, circles, lemniscates, and even more complex trajectories. Two example trajectories are shown in Fig. 5.4. Two additional UWB sensors are placed at known locations acting as beacons. Only agent 1 has access to the beacons, and agent 2 can merely communicate with agent 1. Based on the UWB range measurements, CL can be implemented to correct the zero velocity update (ZUPT) [105] aided pedestrian inertial navigation solution

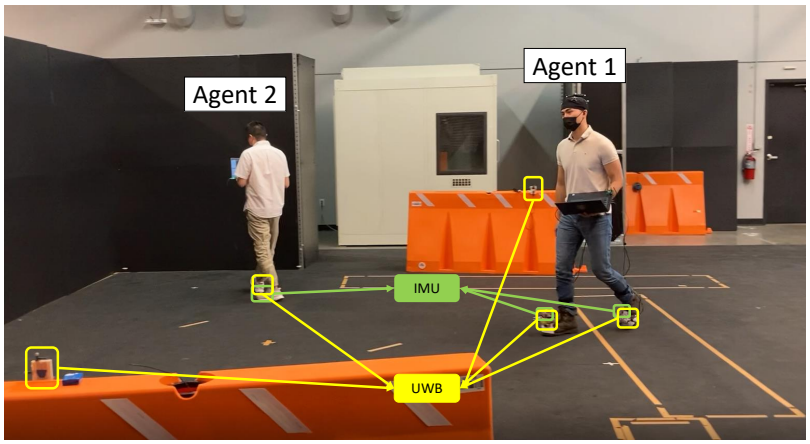


Figure 5.3: Training data generation experiments in Firstnet building.

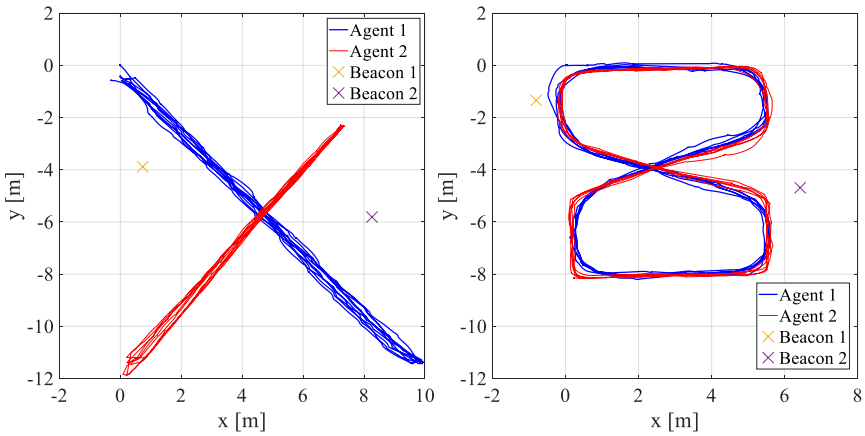
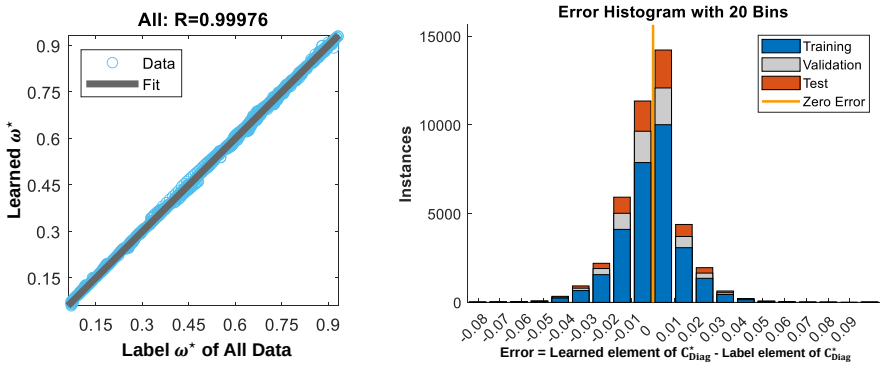


Figure 5.4: Examples of the designed trajectories.

of the agent 2. To generate independent and identically distributed samples for training, validating, and testing, we performed single-step propagation and update from initial belief with randomly generated errors for each pair of self-motion measurement and relative measurement. We solved the optimization problems in both DMV and PECMV using the data from each experiment and obtained an abundant and diverse solution set. The training data is generated by randomly sampling from the entire solution set.

The execution time of DMV is much shorter than PECMV in the sample experiments, as expected. From all experiments, we sampled 50000 samples which are split into a training set (40000 samples), validating set (5000 samples), and test set (5000 samples). The number



(a) $\omega^\star$ NN Fitting (b) $\mathbf{C}^\star_{\text{Diag}}$ NN Error

Figure 5.5: The training results of the $\omega^\star$ prediction NN (a) and the $\mathbf{C}^\star_{\text{Diag}}$ prediction NN (b), respectively. (a) is the fitting performance of $\omega^\star$ prediction NN for all data sets with R to be the square root of the coefficient of determination which indicates the fitness of the predicted output and the target value. The closer to 1, the better the fitness. Also, the perfect fit line is shown as "Fit" in gray. (b) is the training error histogram of the $\mathbf{C}^\star_{\text{Diag}}$ prediction NN which represents the difference between the predicted outputs and the label values of the NN for the training, validation and test set respectively.

of the states of each agent i is $n^i = 9$, and the size of the measurement vector is $n^i_z = 1$. Each sample training data after vectorization is flattened to a vector with $2\times\big(9\times(9+1)/2\big)+2\times 9 = 108$ features due to the symmetry of covariance matrices. The features are normalized to achieve easier optimization and faster learning. Based on the NN structures in Fig. 5.2, we put 108 nodes in the input layer and only 1 node in the output layer of $\omega^\star$ prediction network while the output layer node number of $\mathbf{C}^\star_{\text{Diag}}$ prediction network is 9, i.e., $n^i = n^j = n = 9$. The number of hidden layers and the number of nodes in each hidden layer are fine-tuned using a grid search method during the training process. The activation function of the $\omega^\star$ network is Sigmoid for each layer while the one of $\mathbf{C}^\star_{\text{Diag}}$ network is Tanh. We use Mean-squared-error (MSE) as the loss function due to its convexity and also facilitate the learning using the stochastic gradient descent (SGD) and Adam [127]. The fine-tuned design of the two NNs is shown in Table 5.2, and the training results are shown in Fig. 5.5.

Hyperparameters	$\omega^\star$ network	$\mathbf{C}^\star_{\text{Diag}}$ network
Number of hidden layers L	4	3
Number of units N_L per layer	9	20
Loss	MSE	MSE
Activation function	Sigmoid	Tanh
Optimizer	SGD	Adam
Batch size	256	512
Learning rate α	0.05	0.025
Epochs	200	200

Table 5.2: The hyperparameter design of the NNs.

5.3.1 Numerical Evaluation

We evaluated the efficiency of our proposed LDMV and LPECMV algorithms by implementing them in another set of CL-aided pedestrian inertial navigation experiments conducted in the KCS lab in the Engineering Gateway building at UCI. In these experiments, two agents equipped with shoe-mounted IMUs and UWB sensors walked along two rectangular trajectories maintaining the communication as depicted in Fig. 5.6. Agent 1 kept receiving information from both a UWB anchor at a known location and agent 2. The sampling rates of the IMU for agent 1 and agent 2 were, respectively, 200 Hz and 40 Hz, making agent 1 the more accurate agent and thus creating a non-homogeneous scenario where agent 2 improved its accuracy via CL with agent 1. Both agents implemented a local ZUPT aided INS. We used an OptiTrack real-time motion capture system, as shown in Fig. 5.6, to provide high accuracy reference trajectories for evaluating the performance of our CL algorithms. We attached the reflective markers of the OptiTrack system to the IMUs so that their locations are obtained as references. With the aid of the reference trajectory, the Root Mean Square Error (RMSE) and the Normalized Estimation Error Squared (NEES) are calculated and used as the performance metric to evaluate different localization algorithms. The experiments were repeated 10 times. The experiment results for the position RMSE and the NEES plots are shown in Table 5.3 and Fig. 5.7. The two-sided 95% region for the NEES is $[0.96, 3.42]$ for this set of experiments, see [100]. Also, we compared the average execution time of the CL

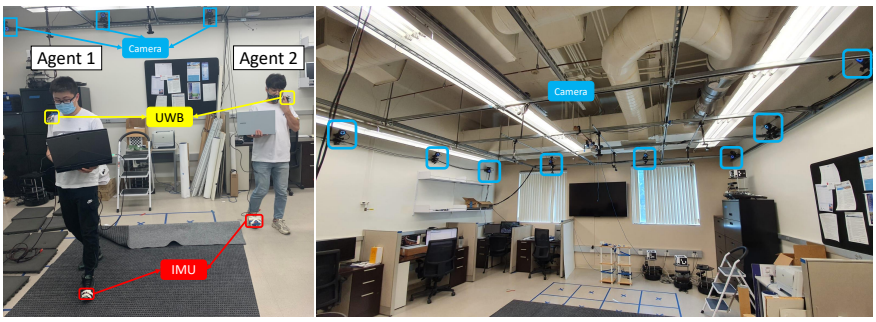
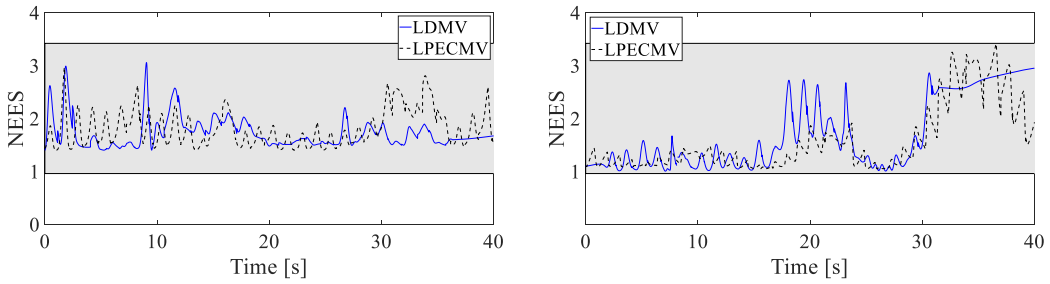


Figure 5.6: The test experiments for LDMV and LPECMV conducted in the KCS lab with the aid of the OptiTrack motion capture system.



(a) Agent 1 (b) Agent 2

Figure 5.7: The NEES plots for the two agents over 50 Monte Carlo runs. The shaded area in the NEES plots show the consistency zone.

	ZUPT-only	DMV	LDMV	PECMV	LPECMV
Agent 1	0.4732	0.1050	0.1158	0.1035	0.1125
Agent 2	0.4902	0.3221	0.3528	0.2950	0.3027

Table 5.3: The average RMSE (m) of the estimated trajectories.

	DMV	LDMV	PECMV	LPECMV
Full simulation time (s)	86.9828	40.1316	189282.6081	43.9001
Single execution time (ms)	3.2197	0.2122	7261.5138	0.2002
Computing platform	Dell Laptop: Intel Core™ i5-1135G7@4.20GHz, quad-core, 8 GB memory			

Table 5.4: The average run time of each algorithm with the computing platform.

algorithms. The results are reported in Table 5.4. The estimated trajectories of the agents are shown in Fig. 5.8.

The NEES plots in Fig. 5.7 show that LDMV and LPECMV CL algorithms maintain the consistency of the estimates the same way as DMV and PECMV as reported in [48]. As it

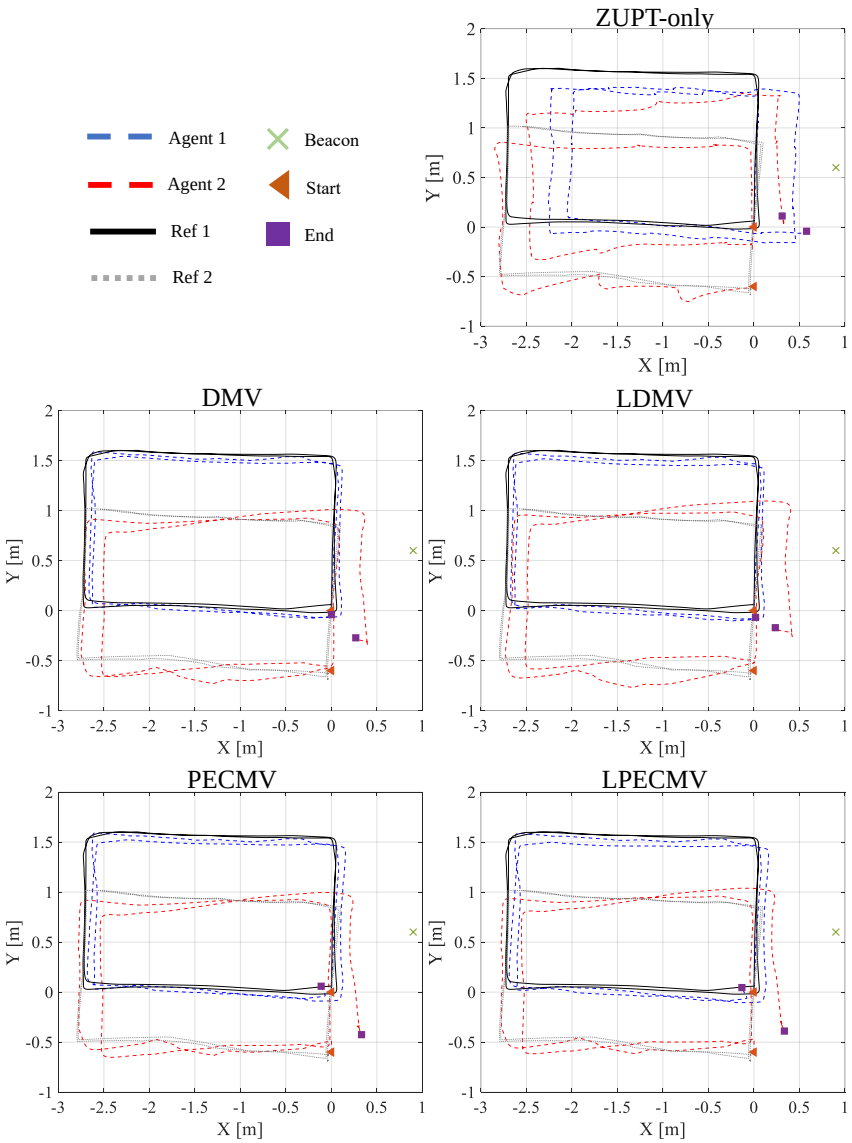


Figure 5.8: The estimated trajectories of the agents implementing ZUPT-only, DMV, LDMV, PECMV and LPECMV algorithms.

can be concluded from Fig. 5.8 and Table 5.3, all the CL algorithms significantly improve the ZUPT/INS-only solution. As expected, PECMV and LPECMV outperform DMV and LDMV respectively in terms of the RMSE due to their less conservatism as reported in [48], and the PECMV and DMV have better localization accuracy compared to the corresponding learning-based algorithms while the differences are insignificant. Because the solution of DMV and PECMV is provably the global optimum [48], LDMV and LPECMV can only perform no better than DMV and PECMV. Moreover, Table 5.4 shows that LDMV achieves

a considerable reduction in execution time in contrast to DMV. Besides, despite the colossal amount of computation time of PECMV, LPECMV takes only 43.9001 seconds. Overall, the percentage reduction of the execution time is 53.85% and 99.98% after applying LDMV and PECMV, respectively, which indicates a significant improvement in the time complexity. In summary, the computational cost is reduced substantially by the proposed learning-based optimization without compromising the localization accuracy and consistency, which enables conducting CL updates at higher rates, especially in embedded computing systems.

5.4 Conclusions

Multi-agent estimation solutions in which the correlation among the local estimates of the agents in cooperative estimation problems are accounted for implicitly, either via the use of conservative uncorrelated estimate upper bounds or by estimating the unknown correlation locally, are attractive because of eliminating persistent inter-agent communications. But, these methods often come with higher computation costs. To reduce the computation cost, this dissertation work proposed to use NN surrogate functions to produce the solution of the time-consuming optimization problems that appear in the implicit approaches to account for correlations. To demonstrate our idea, we focused on the problem of cooperative localization and two particular solutions for this problem from literature: DMV and PECMV methods [48]. We proposed the learning-based algorithms, LDMV and LPECMV, to substitute the time-consuming constrained optimization problems that appear in DMV and PECMV methods by learning their optimal solutions via a NN that is designed elaborately to incorporate the constraints. The training data is obtained by solving the optimization problems directly using the experimental data. Then, the well-trained NNs are used to predict the optimal solution. The efficacy and the generosity of the LDMV and LPECMV are demonstrated in a different set of CL experiments. The agents used the direct predic-

tions from the output of the NNs rather than solve the complex optimization problems in real-time, which makes the computational cost reduced remarkably without compromising the localization accuracy and enables the real-time implementation of the CL algorithms in embedded systems.

Chapter 6

Self-localizing On-demand Portable Wireless Beacons for Coverage Enhancement

Localization systems using RF signals typically utilize pre-installed devices (beacons) with known locations [128]. Robust solution requires access to at least three beacons at all times for continuous and high-accuracy localization. Despite the promise of the RF beacon-based indoor localization, the deployment of these wireless beacons still faces the coverage problem. Full coverage, i.e., making sure the targets will have access to at least three infra structure beacons, requires installing a large number of beacons. Covering an indoor area may involve hundreds of nodes since wireless beacons have effective ranges and coverage degrades due to the signal attenuation. Moreover, knowing the accurate positions of the deployed beacons is vital for the indoor localization algorithms. Therefore, fast and accurate beacon deployment and positioning are key to the implementation of these indoor techniques [129]. And the last part of this dissertation work focus on generalizing the navigation algorithm to reduce connectivity requirements.

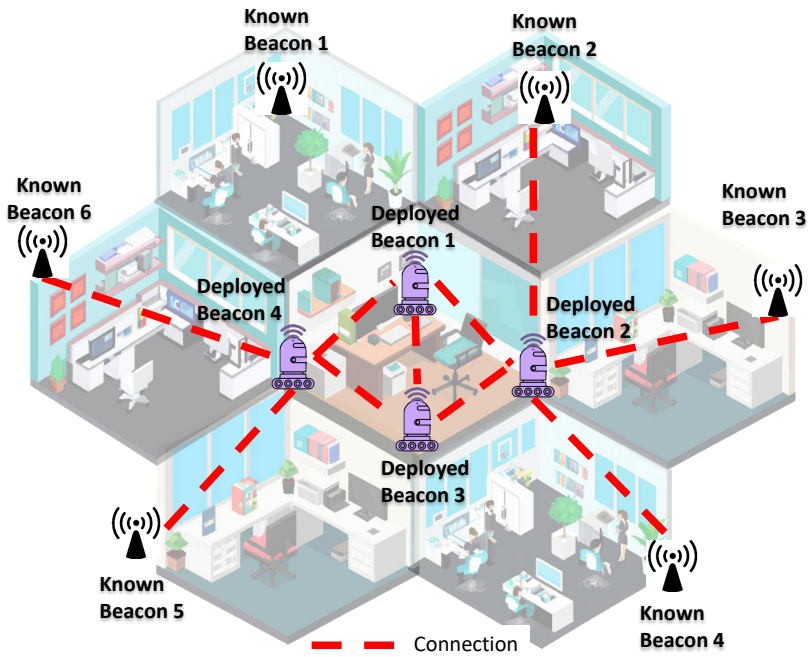


Figure 6.1: A schematic depiction of portable wireless beacon deployment for coverage enhancement of indoor localization systems.

Installing the beacons in underutilized spaces is not conducive. Moreover, in a dynamic environment, it is always possible to have blockage in line-of-sight access to beacons because of indoor obstructions. Deploying on-demand portable beacons to extend the coverage is a cost-effective solution for a robust and reliable RF beacon based localization system. The challenge though is how to localize these deployed beacons. In this work, we propose a novel decentralized cooperative self-localization method to localize portable wireless beacons that are deployed to tackle the limited connectivity/coverage of the infrastructure beacons, see Fig 6.1. Our method is able to localize the deployed beacons in the condition that some of them have no access to the infrastructure beacons, but they are able to self-localize by cooperating with nearby deployed beacons which have connectivity to the infrastructure beacons. Besides, the correlation among deployed beacons are treated carefully and implicitly to ensure the consistency of the positioning. Finally, since this method can be computed in a distributed way, it is also suitable for the large scale beacon deployment. We describe next the self-localization problem and the techniques we propose to solve it.

6.1 Problem Statement

Consider a network deployment of N ($N > 3$) beacons with communication and computation capabilities in an M-dimensional space, $M \in \{2, 3\}$, whose positions are denoted as $\mathbf{X} = [\mathbf{x}^1, \mathbf{x}^2, \cdots, \mathbf{x}^N] \in \mathbb{R}^{M \times N}$. Within the network, assume there are at least three infrastructure beacons installed at known positions $\mathbf{X}_k$ which have been measured elaborately, and the rest of the beacons are deployed, to enhance the beacon coverage and signal strength of certain areas, at unknown positions $\mathbf{X}_u$ which are required to be determined such that $\mathbf{X} = [\mathbf{X}_k, \mathbf{X}_u]$. Let the relative measurement (e.g., relative range, relative bearing, relative pose, or a combination of them) $\mathbf{z}_j^i \in \mathbb{R}^{1,2,3}$ taken by beacon i from beacon j be described as

$$\mathbf{z}_j^i = \mathbf{h}(\mathbf{x}^i, \mathbf{x}^j) + \boldsymbol{\nu}^i, \tag{6.1}$$

where $\mathbf{h}$ is the relative measurement model and $\boldsymbol{\nu}^i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}^i)$ is the zero mean Gaussian measurement noise with covariance matrix $\mathbf{R}^i$.

Given the available relative measurements $\{\mathbf{z}_j^i\}$ and the known beacon positions $\mathbf{X}_k$, our objective is to determine the unknown beacon positions $\mathbf{X}_u$ subject to limited connectivity between beacons, i.e., beacons are only connected to their neighbors whose positions may also belong to $\mathbf{X}_u$ and thus need to be estimated. In other words, due to the effective range of wireless beacons, some deployed beacons may not be able to obtain direct measurements from the beacons at known positions. Another issue is the correlation among unknown beacons. Once the estimates of any two beacons, $\hat{\mathbf{x}}_u^i$ and $\hat{\mathbf{x}}_u^j$, are correlated, to keep an explicit track of the correlations requires a high communication and computation capability [48]. Moreover, given that the beacon network can be dynamical, i.e., beacons can be added or removed from the network as desired, centralized localization method are less suitable in terms of autonomous self-localization.

In general, the position of each beacon i in the set of beacons with unknown positions (hereafter we call them unknown beacons) denoted as U is estimated by solving the below optimization problem

$$\hat{\mathbf{x}}_u^i = \underset{\mathbf{x}_u^i}{\operatorname{argmin}} \sum_{j \in N_G(i)} \left\| \mathbf{z}_j^i - \hat{\mathbf{z}}_j^i \right\|^2, \tag{6.2}$$

with the uncertainty $\mathbf{P}^i = \mathbb{E}[(\hat{\mathbf{x}}_u^i - \mathbb{E}[\hat{\mathbf{x}}_u^i])(\hat{\mathbf{x}}_u^i - \mathbb{E}[\hat{\mathbf{x}}_u^i])^\top]$, where $N_G(i)$ represents the set of neighbors of beacon i and $\hat{\mathbf{z}}_j^i = \mathbf{h}(\hat{\mathbf{x}}_u^i, \hat{\mathbf{x}}_u^j)$ is the estimated measurement.

6.2 Beacon Self-Localization

To solve (6.2), depending on whether the neighbor beacon j has a known position $\mathbf{x}_k^j$ or an unknown position $\mathbf{x}_u^j$, the solutions differ in two cases.

6.2.1 Recursive Least Squares Method

For the first case, if $j \notin U$, which indicates the beacon i is connected to the known neighbor j, (6.2) is solved for all $j \in N_{G(i)}$ at time t by the recursive least squares (RLS) algorithm [100] with uncertainty $\mathbf{P}^i[t]$ using sequential updating to process multiple concurrent measurements $\{\mathbf{z}_j^i[t]\}$ (See [100]) as follow

$$\hat{\mathbf{x}}_u^i[t] = \hat{\mathbf{x}}_u^i[t-1] + \mathbf{K}^i[t](\mathbf{z}_j^i[t] - \hat{\mathbf{z}}_j^i[t]), \tag{6.3a}$$

$$\mathbf{P}^i[t] = \big(\mathbf{I} - \mathbf{K}^i[t]\mathbf{H}^i[t]\big)\mathbf{P}^i[t-1], \tag{6.3b}$$

where $\mathbf{K}^i[t] = \mathbf{P}^i[t-1]\mathbf{H}^{i\top}[t]\big(\mathbf{H}^i[t]\mathbf{P}^i[t-1]\mathbf{H}^{i\top}[t] + \mathbf{R}\big)^{-1}$ is known as the Kalman gain and $\mathbf{H}^i = \partial\mathbf{h}[t](\hat{\mathbf{x}}_u^i[t-1], \hat{\mathbf{x}}_k^j[t-1])/\partial\mathbf{x}_u^i$.

If beacon i is connected to an unknown neighbor j, i.e., $j \in U$, solving the optimization problem in (6.2) requires careful handling of the correlation between unknown beacons. In short, once a relative measurement is proceeded to update the estimate of two unknown beacons, they are correlated. If the correlation is not properly taken care of or is completely ignored, the estimator will be likely to diverge [48]. Therefore, in the following sub-section, we introduce a decentralized cooperative localization algorithm, Discorrelated Minimum Variance (DMV), proposed in [48] to solve the self-localization problem (6.2) with elaborate design to deal with the correlation issue instead of the RLS. For more information, see [48].

6.2.2 Decentralized Cooperative Self-localizing Method

To simplify the notation and avoid confusion, hereafter we only include the subscripts u and k for position estimates when clarification is needed. For $i, j \in U$, let the joint position estimate mean and covariance of the beacons i and j at time $t-1$ be $\hat{\mathbf{x}}_J[t-1]$ and $\mathbf{P}_J[t-1]$, where

$$\hat{\mathbf{x}}_J[t-1] = \begin{bmatrix} \hat{\mathbf{x}}^i[t-1] \\ \hat{\mathbf{x}}^j[t-1] \end{bmatrix}, \tag{6.4a}$$

$$\mathbf{P}_J[t-1] = \begin{bmatrix} \mathbf{P}^i[t-1] & \mathbf{P}^{ij}[t-1] \\ \mathbf{P}^{ij}[t-1]^\top & \mathbf{P}^j[t-1] \end{bmatrix}. \tag{6.4b}$$

When beacon i takes a relative measurement $\mathbf{z}_j^i[t]$ from beacon j at time t, beacon i can correct its local estimate using the measurement feedback $\mathbf{z}_j^i[t] - \hat{\mathbf{z}}_j^i[t]$. When the cross-covariance term $\mathbf{P}^{ij}[t-1]$ is known, the feedback gain can be computed from a Kalman like update procedure. When $\mathbf{P}^{ij}[t-1]$ is not available or too computationally expensive to be tracked, the DMV algorithm is able to process the relative measurement $\mathbf{z}_j^i[t]$ to correct the local estimates in the absence of explicit knowledge of the cross-covariance $\mathbf{P}^{ij}[t-1]$. It is

important to note that correlation terms $\mathbf{P}^{ij}[t-1]$ due to prior relative measurement updates cannot be ignored trivially. Ignoring the correlations leads to overconfident estimates and even filter divergence [130].

In the DMV method, a discorrelated upper bound is used to account for any unknown cross-covariance term $\mathbf{P}^{ij}[t-1]$ as

$$\begin{bmatrix} \mathbf{P}^i[t-1] & \mathbf{P}^{ij}[t-1] \\ \mathbf{P}^{ij}[t-1]^\top & \mathbf{P}^j[t-1] \end{bmatrix} \leq \begin{bmatrix} \frac{1}{\omega}\mathbf{P}^i[t-1] & \mathbf{0} \\ \mathbf{0} & \frac{1}{1-\omega}\mathbf{P}^j[t-1] \end{bmatrix}, \quad \omega \in [0,1]. \tag{6.5}$$

Then, the DMV algorithm updates the position estimate of agent i at time $t-1$, $(\hat{\mathbf{x}}^i[t-1], \mathbf{P}^i[t-1])$, according to

$$\hat{\mathbf{x}}^i[t] = \hat{\mathbf{x}}^i[t-1] + \bar{\mathbf{K}}^i(\omega_\star^i)\,(\mathbf{z}_j^i[t] - \hat{\mathbf{z}}_j^i[t]), \tag{6.6a}$$

$$\mathbf{P}^i[t] = \bar{\mathbf{P}}^i(\omega_\star^i), \tag{6.6b}$$

where $\bar{\mathbf{K}}^i(\omega) = \frac{\mathbf{P}^i[k-1]}{\omega}\mathbf{H}_i^{i\top}\big(\mathbf{H}_i^i\frac{\mathbf{P}^i[k-1]}{\omega}\mathbf{H}_i^{i\top} + \mathbf{H}_j^i\frac{\mathbf{P}^j[k-1]}{1-\omega}\mathbf{H}_j^{i\top} + \mathbf{R}^i\big)^{-1}$, and $\bar{\mathbf{P}}^i(\omega) = \big(\omega(\mathbf{P}^i[t-1])^{-1} + (1-\omega)\mathbf{H}_i^{i\top}(\mathbf{H}_j^i\mathbf{P}^j[t-1]\mathbf{H}_j^{i\top} + (1-\omega)\mathbf{R}^i)^{-1}\mathbf{H}_i^i\big)^{-1}$, with $\mathbf{H}_i^i = \partial\mathbf{h}(\hat{\mathbf{x}}^i[t-1], \hat{\mathbf{x}}^j[t-1])/\partial\mathbf{x}^i$ and $\mathbf{H}_j^i = \partial\mathbf{h}(\hat{\mathbf{x}}^i[t-1], \hat{\mathbf{x}}^j[t-1])/\partial\mathbf{x}^j$. The optimal coefficient ω, denoted by $\omega_\star^i$, is obtained from the optimization problem

$$\omega_\star^i = \underset{0\leq\omega\leq1}{\operatorname{argmin}} \ \log\det\bar{\mathbf{P}}^i(\omega). \tag{6.7}$$

Similarly, the sequential updating technique is applied for DMV. According to [48, Theorem 3.1], despite the unknown $\mathbf{P}^{ij}[t-1]$, the DMV update is guaranteed to be no worse than the local estimates of beacon i and j. The computation complexity in (6.7) can be reduced by applying machine learning tools as we have demonstrated in our prior work in [131]. That

is, the optimal coefficient ω can be predicted directly from a neural network as

$$\omega_\star^i = f_{\text{DMV}}(\mathbf{P}^i, \mathbf{P}^j, \mathbf{H}^i, \mathbf{H}^j), \tag{6.8}$$

where the f_{DMV} is the function approximated by the neural network. See [131] for more information.

6.2.3 Full Algorithm

In summary, the entire beacon self-localization process consists of both RLS and DMV which is depicted in Algorithm 2.

Algorithm 2 Beacon Self-Localization

Input: $\{\hat{\mathbf{x}}^i[t=0], \mathbf{P}^i[t=0], \mathbf{z}_j^i\}$ for all $i \in U$ and $j \in N_{G(i)}$
repeat
    $t = t + 1$
    for each $j \in N_{G(i)}$ **do**
        if $j \notin U$ **then**
            Calculate $\hat{\mathbf{x}}^i[t]$ and $\mathbf{P}^i[t]$ using (6.3)
        else
            Calculate $\hat{\mathbf{x}}^i[t]$ and $\mathbf{P}^i[t]$ using (6.6) and (6.7) or (6.8)
        end if
    end for
until Convergence
Output: $\hat{\mathbf{x}}^i[t]$, $\mathbf{P}^i[t]$

6.3 Simulation Study

In this section, we assess the effectiveness of our beacon self-localization algorithm through a simulated scenario. The scenario involved the deployment and self-localization of 25 UWB beacons in a 2-D space with an area of $60m \times 60m$, i.e., $M = 2$ and $N = 25$. UWB beacons can take the TOA ranging measurements, i.e., $\mathbf{h}(\mathbf{x}^i, \mathbf{x}^j) = \sqrt{(\mathbf{x}^i - \mathbf{x}^j)^2}$, to obtain the distance

information between beacons. Moreover, the measurement noise covariance were chosen to be $R = 0.01$ with respect to the empirical ranging error of UWB sensor whose standard deviation is 0.1m. The simulation time interval was $\Delta t = 0.1$, the total simulation time was $T = 50$ seconds and the sampling rate of the UWB beacon was 10 Hz. The configuration of the simulation is illustrated in Fig. 6.2. As it is shown in Fig. 6.2, there are 15 beacons with known positions denoted by red circles being installed in the hallway environment while 10 additional beacons denoted by the crosses are deployed at the locations as desired. The dashed lines represent the connectivity among the known and unknown beacons which is designed on purpose. We assumed the prior knowledge of the positions of the unknown beacons that we desired to deploy them inside the area surrounded by the known beacons. The objective is to determine the positions of all unknown UWB beacons x_u^i ($i = 1, 2, \cdots, 10$ in this case) given the range measurements and constrained by the connectivity. Note that in this setting, beacon 2, 7, 8 and 9 (marked with green crosses in Fig. 6.2) connect to two known beacons, beacon 3, 5 and 6 (marked with orange crosses) only connect to one known beacon and beacon 4 (marked with a blue cross) connects to no known beacons by design to simulate the limited connectivity condition while all unknown beacons connect to at least three beacons.

Three groups of simulations were conducted to estimate the unknown beacon positions separately and the simulations were repeated 1000 times in each group to mitigate the random error that occasionally caused by the bad initialization of unknown beacons. In the first group of simulations, we only proceed the measurements between the known and unknown beacons using RLS estimator to make it serve as a control group. The second one was what we called "NaiveCL" group referring to the naive way of processing the correlation between correlated beacons where the correlations are completely ignored. It is equivalent to using the RLS solely to estimate all unknown beacon positions with all available measurements. We implemented the proposed self-localization method in the last group of simulations. Besides, without loss of generosity, we intentionally moved the beacon 6 to a new assigned location at

timestamp 300 when the algorithm converged for the first time to demonstrate the robustness of our method in terms of the autonomous re-localization capability. The simulation results are illustrated in Fig. 6.3, Fig. 6.4, Fig. 6.5 and Table 6.1 where the average root mean square error (RMSE) and maximum error were used as the performance metric of the self-localization. Absolute error and the $3-\sigma$ bound were drawn to compare the consistency and robustness of the methods. And we drew gray semi-transparent boxes in the figures to highlight the beacons of interest and their corresponding ground truth locations.

As it is visualized in Fig. 6.3a, beacon 3, 4, 5 and 6 cannot self-localize themselves well with measurements from only one known beacon as expected because at least measurements from three beacons (non-collinear with the beacon to be estimated) are need to estimate precisely the positions in the 2-D space. Therefore, the relative measurements between unknown beacons should also be taken into account to estimate the beacon positions. However, if we naively update the local estimates and ignore the correlation between the unknown beacons, the localization performance will be deteriorated. Fig. 6.3b shows the corresponding evidence and in Table 6.1, the average RMSE and maximum error of the NaiveCL are significantly larger than those of our proposed method due to the inconsistency of the estimator when correlations are neglected although the NaiveCL outperforms the RLS method in the first group. In contrast, our proposed method significantly improves the localization performance as shown in Fig. 6.3c with the lowest average RMSE and maximum error in Table 6.1. With a particular care of the correlation between the unknown beacons, our proposed method enables the accurate beacon self-localization without the necessity of each beacon being connected to three pre-installed infrastructure beacons compared with the other two methods.

Another comparison regarding to the robustness is that after being moved for certain reasons to a new positions, the beacon 6 cannot accurately acquire its new position and even ends up with affecting the neighbor beacons and resulting in a higher localization error using the

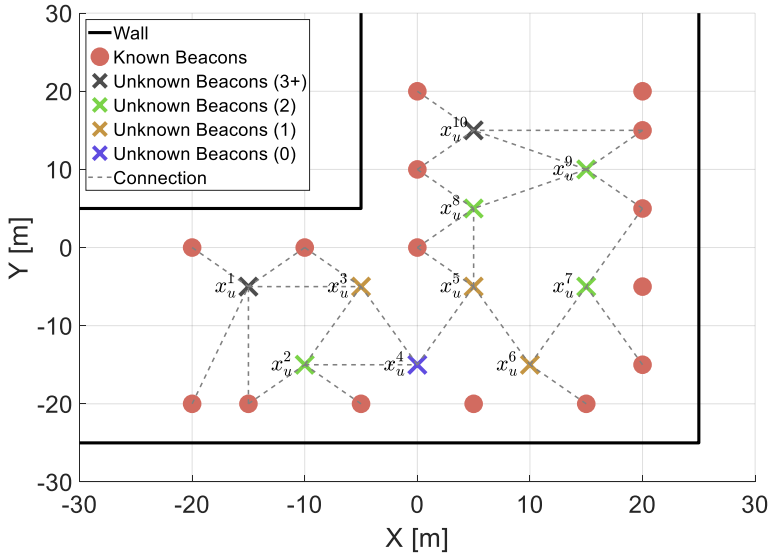
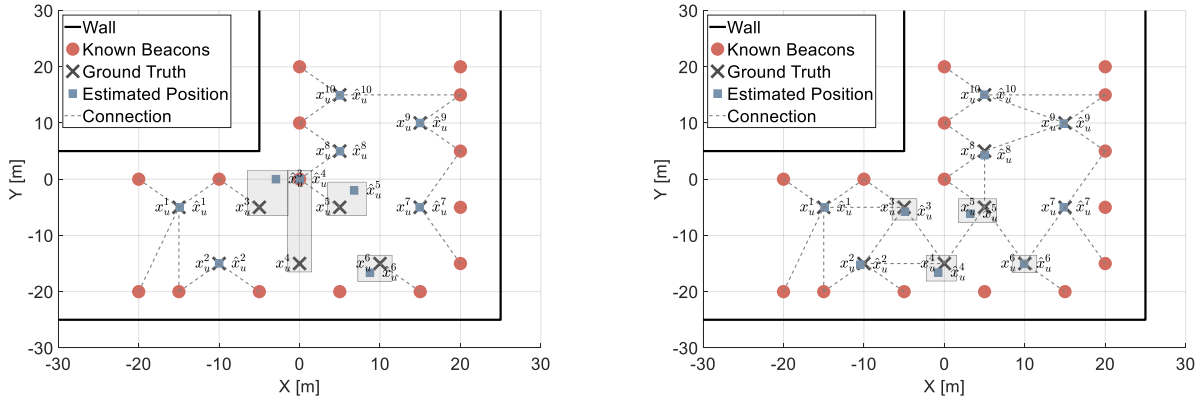


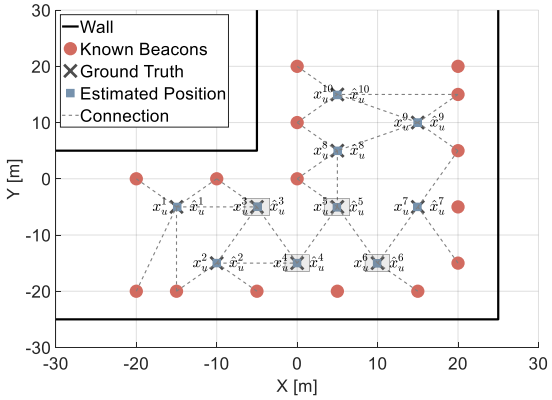
Figure 6.2: The simulation configuration of beacon self-localization. The black, green, orange and blue crosses represent the unknown beacons with more than three connections, two connections, only one connection and no connection to known beacons, respectively.

NaiveCL as shown in Fig. 6.4. However, by implementing our proposed method, beacon 6 still re-localize itself in the decentralized manner without effecting the neighbor beacons demonstrated in Fig. 6.5. The localization error falls into the $3-\sigma$ bound throughout the entire simulation, which exemplifies the robustness of our solution and the capability to deploy the beacons on demand to extend the coverage of the beacon networks. As long as each beacon has at least three connections to neighbors, pre-knowledge or manual measurement of only a handful of beacon position is sufficient to estimate other unknown ones accurately, which perfectly facilitates the coverage enhancement for the beacon-based localization systems. Future work will focus on the initialization of the unknown beacons to avoid the estimated beacon positions being stuck in a statistical local minimum and further enhance the robustness of the proposed algorithm.



(a) RLS

(b) NaiveCL



(c) Proposed

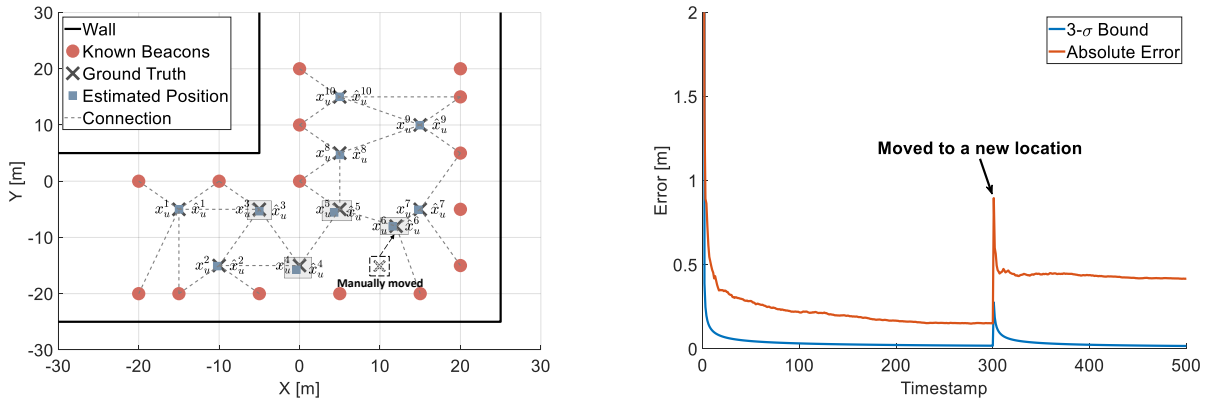
Figure 6.3: The self-localization results in three groups of simulations. The gray semi-transparent boxes in the figures bound the highlighted beacons of interest and their ground truth locations.

Table 6.1: Localization accuracy regarding average RMSE and maximum error in three groups over 1000 simulations.

Method	**RMSE [m]**	**Maximum Error [m]**
RLS	5.1599	12.0692
NaiveCL	1.6355	4.0473
Proposed	1.0727	2.3298

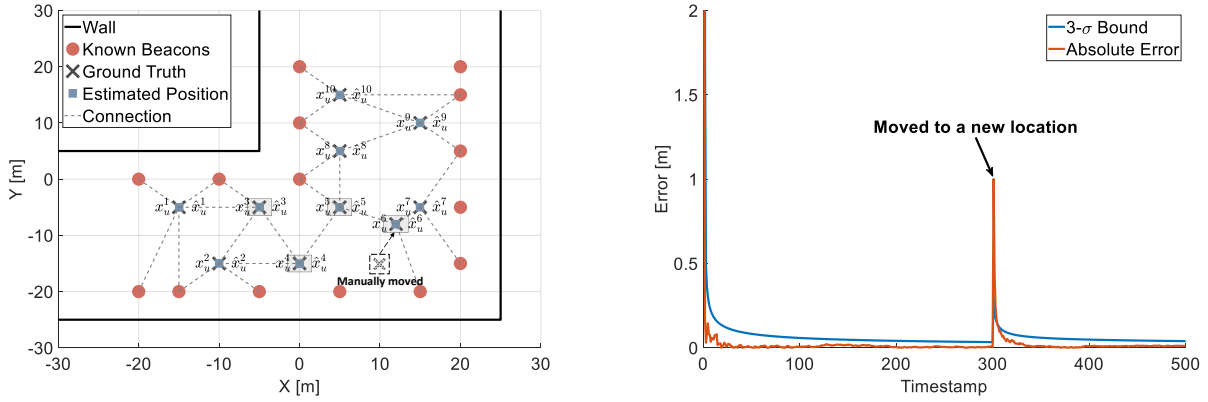
6.4 Conclusion

This dissertation work considered the beacon self-localization under limited connectivity for deploying the portable wireless beacons. Specifically, we proposed a decentralized beacon self-localization algorithm which relaxes the essential requirement of sufficient connections



(a) Re-localization Result (b) Absolute Error

Figure 6.4: The re-localization performance of the NaiveCL method. Before the beacon was moved to a new location, the NaiveCL has worse performance of self-localization in terms of consistency. After the re-localization, it generates larger error and cannot converge to the new ground truth.



(a) Re-localization Result (b) Absolute Error

Figure 6.5: The re-localization performance of our proposed method. After the beacon was moved to a new location, our method made the estimate converged to the new ground truth and stayed consistency as before.

to the beacons with known positions and avoids the labor-intensive and error-prone manual measurement of the beacon positions. The simulation results demonstrated the efficacy of our proposed method in terms of the accurate autonomous position estimation for multiple beacons and its robust re-localization capability which enables a more flexible deployment of the on-demand portable beacons. Additionally, the distributed computation pattern reduces the computational burden on each single device. When the position estimates of beacons are accurate sufficiently, they can serve as the new known beacon to consequently extend

the network coverage. Last but not least, in large-scale settings, the significant reduction in human labor fulfilled by our method is substantial.

Chapter 7

Conclusions

This dissertation work contributes towards enhancing the robustness of navigation algorithms via integrating integrity monitoring, effective use of machine learning tools for navigation systems, reducing the computational complexity to enable real-time implementation, and innovative navigation framework to enhance external signal access. The objective is the design of robust and light-weight navigation algorithms to monitor the sensor performance and enhance the navigation accuracy. The main application of this work is the integrity monitoring and measurement enhancement for INS aided by external measurements obtained from GPS and UWB sensors in a tightly coupled manner with cooperative processing. This work is end-to-end spanning algorithm design, theoretical modeling and analysis and experimental demonstrations/validation. The main contributions of this work are as follows.

- **Developing a novel integrity monitoring framework for Bayesian filter-based multi-sensor navigation systems to increase the fault detection rate and enhance the robustness of the navigation**

  In order to achieve robust navigation requirements, the faulty measurements have to be detected and excluded before using. Given that the modern navigation systems are

commonly based on filter-based framework, we propose a modified Rényi divergence-based FDE approach for a tightly coupled GNSS/INS system. The main contribution is to analyze the components of the close-form of the Rényi divergence under Gaussian assumption and to propose a modified Rényi divergence as the new test statistics that has a lower computational complexity but delivers the same result. Further, we present a statistical procedure to select the order α of the Rényi divergence measure for FDE to increase the detection accuracy. Additionally, we introduce a ratio test among the Rényi divergence measure between the propagated and updated estimates due to individual GNSS measurement channels to detect multiple faults in one single check. Lastly, a probabilistic threshold method is proposed to enable the autonomous threshold selection. We demonstrate our results via a simulated flight platform in CADAC++ and MATLAB. The simulation results show the desirable performance for FDE in reducing false alarm cases, eliminating miss detection cases, and selecting threshold optimally and autonomously.

- **Developing a learning-based UWB LoS/NLoS measurement model discriminator to increase the localization accuracy.**

  UWB ranging bias correction has a significant impact on the localization accuracy of the UWB-based navigation algorithms. In this dissertation work, we addressed the problem of UWB ranging bias correction in complex localization scenarios by training neural network models to distinguish the LoS and NLoS scenarios and predict the corresponding bias-free range in real-time using features that are readily available on low-cost UWB sensors. Our experimental data collection resulted in identifying a novel set of features, raw range, PM, FPPL and RSS, to train ANNs for measurement type classification and bias-free range prediction. The effectiveness of any learning-based solution depends on the diversity and informativeness of its training data. Recognizing that the relative orientation of the ranging sensors affects the ranging accuracy, we used an Optitrack motion capture system to collect high-accuracy ranging between a tag

UWB sensor on a moving agent and fixed UWB anchor nodes. In doing so, we collected a diverse set of data in various relative poses between the sensors. The effectiveness of our OptiTrack-aided supervised learning-based UWB bias correction method was demonstrated via a set of pedestrian localization experiments using the RLS algorithm for location estimation in two different locations. Our experiments showed that our method leads to a considerable improvement in the localization accuracy.

- **Designing learning-based surrogate functions to substitute the time-consuming constrained optimization problems that appear in the implicit CL algorithms by learning the optimal solutions and respecting the constraints and reduce the computational complex of the algorithms without sacrificing the navigation performance.**

  Multi-agent estimation solutions in which the correlation among the local estimates of the agents in cooperative estimation problems are accounted for implicitly, either via the use of conservative uncorrelated estimate upper bounds or by estimating the unknown correlation locally, are attractive because of eliminating persistent inter-agent communications. But, these methods often come with higher computation costs. To reduce the computation cost, this dissertation work proposed to use NN surrogate functions to produce the solution of the time-consuming optimization problems that appear in the implicit approaches to account for correlations. To demonstrate our idea, we focused on the problem of cooperative localization and two particular solutions for this problem from literature: DMV and PECMV methods. We proposed the learning-based algorithms, LDMV and LPECMV, to substitute the time-consuming constrained optimization problems that appear in DMV and PECMV methods by learning their optimal solutions via a NN that is designed elaborately to incorporate the constraints. The training data is obtained by solving the optimization problems directly using the experimental data. Then, the well-trained NNs are used to predict the optimal solution. The efficacy and the generosity of the LDMV and LPECMV are demonstrated

in a different set of CL experiments. The agents used the direct predictions from the output of the NNs rather than solve the complex optimization problems in real-time, which makes the computational cost reduced remarkably without compromising the localization accuracy and enables the real-time implementation of the CL algorithms in embedded systems.

- **Proposing the self-localizing on-demand portable wireless beacons for coverage enhancement**

  This dissertation work consider the beacon self-localization under limited connectivity for deploying the portable wireless beacons. Specifically, we proposed a decentralized beacon self-localization algorithm which relaxes the essential requirement of sufficient connections to the beacons with known positions and avoids the labor-intensive and error-prone manual measurement of the beacon positions. The simulation results demonstrated the efficacy of our proposed method in terms of the accurate autonomous position estimation for multiple beacons and its robust re-localization capability which enables a more flexible deployment of the on-demand portable beacons. Additionally, the distributed computation pattern reduces the computational burden on each single device. When the position estimates of beacons are accurate sufficiently, they can serve as the new known beacon to consequently extend the network coverage. Last but not least, in large-scale settings, the significant reduction in human labor fulfilled by our method is substantial and reversely the robustness is enhanced.

The work in this dissertation has led to the design of a robust navigation framework that works as an augmentation service on top of sensor fusion solutions to improve localization accuracy and enhance the navigation safety. Based on this dissertation work, we reinforce the robustness and security of modern navigation solutions, specifically the indoor UWB/INS navigation for firefighters and first-responders that run in real-time on a portable computing system. Extensive simulations and experiments demonstrated the effectiveness of our design.

7.1 Future Work

In light of the findings of our work, we discuss several future research directions below.

- In fault detection and exclusion, incipient faults are hard to be detected by residual-based integrity monitoring algorithms since the statistics we use for detection has high dependency on the historical state estimates. The incipient faults can grow slowly preventing the test statistics from exceeding the threshold but destroy the navigation solution completely. Therefore, developing a more robust FDE algorithm that can typically deal with incipient faults efficiently is of great importance.

- For the application of the FDE, the multiple object tracking (MOT) in computer vision (CV) is promising. Given that the forefront research effort of the MOT is to increase the detection rate of the target and maintain the robust tracking under occlusion, FDE is extremely important. A general mathematical FDE framework for image sequence data is necessary to be developed and thus contribute to the CV development.

- In measurement bias or fault modeling, the assumption that the bias or fault is a Gaussian random variable is not always true. Also, using probability-theoretical techniques to develop a bias modeling and compensation method, which eliminates the particular assumptions, can lead to more generalized applications to other navigation systems. Lastly, the learning-based method has the limitation of overfitting and cannot be easily generalized to more complex scenarios. As future work, one can investigate the combination of learning methods and statistical inference methods to further improve the bias modeling in RF signals.

- Another venue for the future work is investigating the robust cooperative localization algorithm without Gaussian assumption, i.e., generalizing the DMV and PECMV method to general Bayesian filtering framework. With a potential transformation, the

computational complexity can be significantly reduced for maintaining the correlation. Also, theoretical development of the FDE can be done for non-Gaussian filters.

Bibliography

[1] Jang-Sik Yoo, Jong-Sun Ahn, Young-Jae Lee, and Sang-Kyung Sung. Performance comparison of gps fault detection and isolation via pseudorange prediction model based test statistics. *Journal of Electrical Engineering and Technology*, 7(5):797–806, 2012.

[2] K. Gururaj, A. K. Rajendra, Y. Song, C. L. Law, and G. Cai. Real-time identification of NLOS range measurements for enhanced UWB localization. In *International Conference on Indoor Positioning and Indoor Navigation*, Sapporo, Japan, September 2017.

[3] Esmat Bekir. *Introduction to modern navigation systems*. World scientific, 2007.

[4] Oliver J. Woodman. An introduction to inertial navigation, 2007.

[5] David Titterton, John L Weston, and John Weston. *Strapdown inertial navigation technology*, volume 17. IET, 2004.

[6] Jianan Zhu and Solmaz S Kia. Decentralized cooperative localization with LoS and NLoS UWB inter-agent ranging. *IEEE Sensors Journal*, 22(6):5447–5456, 2021.

[7] Jianan Zhu and Solmaz S. Kia. Bias compensation for UWB ranging for pedestrian geolocation applications. *IEEE Sensors Letters*, 3(9):1–4, 2019.

[8] Changwei Chen and Solmaz S Kia. A Renyi divergence based approach to fault detection and exclusion for tightly coupled GNSS/INS system. In *Proceedings of the 2021 International Technical Meeting of the Institute of Navigation*, pages 674–687, 2021.

[9] R Grover Brown, BW Parkinson, and JJ Spilker. Receiver autonomous integrity monitoring. *Global Positioning System: Theory and applications.*, 2:143–165, 1996.

[10] Federal Radionavigation Plan. Dod-4650. 4. 1986.

[11] Mark A Sturza. Navigation system integrity monitoring using redundant measurements. *Navigation*, 35(4):483–501, 1988.

[12] Young C Lee. Analysis of range and position comparison methods as a means to provide GPS integrity in the user receiver. In *Proceedings of the 42nd Annual Meeting of the Institute of Navigation (1986)*, pages 1–4, Seattle, WA, 1986.

[13] Juan Blanch, Todd Walter, Per Enge, Young Lee, Boris Pervan, Markus Rippl, and Alex Spletter. Advanced RAIM user algorithm description: integrity support message processing, fault detection, exclusion, and protection level calculation. In *Proceedings of the 25th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2012)*, pages 2828–2849, Nashville, TN, 2012.

[14] Navid Hashemi, Eduardo Verdugo German, Jonatan Pena Ramirez, and Justin Ruths. Filtering approaches for dealing with noise in anomaly detection. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 5356–5361, Nice, France, 2019.

[15] Junsoo Kim, Jin Gyu Lee, Chanhwa Lee, Hyungbo Shim, and Jin H Seo. Local identification of sensor attack and distributed resilient state estimation for linear systems. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 2056–2061, Miami, FL, 2018. IEEE.

[16] Angelo Barboni, Alexander J Gallo, Francesca Boem, and Thomas Parisini. A distributed approach for the detection of covert attacks in interconnected systems with stochastic uncertainties. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 5623–5628, Nice, France, 2019. IEEE.

[17] Twan Keijzer and Riccardo MG Ferrari. A sliding mode observer approach for attack detection and estimation in autonomous vehicle platoons using event triggered communication. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 5742–5747, Nice, France, 2019. IEEE.

[18] Jun Xiong, Joon Wayn Cheong, Zhi Xiong, Andrew G Dempster, Shiwei Tian, and Rong Wang. Integrity for multi-sensor cooperative positioning. *IEEE Transactions on Intelligent Transportation Systems*, 22(2):792–807, 2019.

[19] Mathieu Joerger, Fang-Cheng Chan, Steven Langel, and Boris Pervan. RAIM detector and estimator design to minimize the integrity risk. In *Proceedings of the 25th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2012)*, pages 2785–2807, Nashville, TN, 2012.

[20] Imad Y Hoballah and Pramod K Varshney. An information theoretic approach to the distributed detection problem. *IEEE Transactions on Information Theory*, 35(5):988–994, 1989.

[21] Nourdine Aït Tmazirte, Maan E El Najjar, Joelle Al Hage, Cherif Smaili, and Denis Pomorski. Fast multi fault detection & exclusion approach for GNSS integrity monitoring. In *17th International Conference on Information Fusion (FUSION)*, pages 1–6, Salamanca, Spain, 2014. IEEE.

[22] Joelle Al Hage and Maan El Badaoui El Najjar. Improved outdoor localization based on weighted Kullback-Leibler divergence for measurements diagnosis. *IEEE Intelligent Transportation Systems Magazine*, 12(4):41–56, 2018.

[23] Khoder Makkawi, Nourdine Ait-Tmazirte, Maan E. El Najjar, and Nazih Moubayed. Combination of maximum correntropy criterion amp; ∝-Rényi divergence for a robust and fail-safe multi-sensor data fusion. In *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 61–67, Karlsruhe, Germany, 2020.

[24] Jinwon Kim, Jang-Gyu Lee, Gyu-In Jee, and Tae-Kyung Sung. Compensation of gyroscope errors and GPS/DR integration. In *Proceedings of Position, Location and Navigation Symposium - PLANS '96*, pages 464–470, Atlanta, GA, 1996.

[25] Kichun Jo, Keonyup Chu, and Myoungho Sunwoo. GPS-bias correction for precise localization of autonomous vehicles. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 636–641, Gold Coast City, Australia, 2013.

[26] Jianan Zhu and Solmaz S. Kia. Bias compensation for UWB ranging for pedestrian geolocation applications. *IEEE Sensors Letters*, 3(9):1–4, 2019.

[27] Fazeelat Mazhar, Muhammad Gufran Khan, and Benny Sällberg. Precise indoor positioning using UWB: A review of methods, algorithms and implementations. *Wireless Personal Communications*, 97(3):4467–4491, 2017.

[28] Daquan Feng, Chunqi Wang, Chunlong He, Yuan Zhuang, and Xiang-Gen Xia. Kalman-filter-based integration of IMU and UWB for high-accuracy indoor positioning and navigation. *IEEE Internet of Things Journal*, 7(4):3133–3146, 2020.

[29] Om Prakash Kumar, Pramod Kumar, Tanweer Ali, Pradeep Kumar, and Shweta Vincent. Ultrawideband antennas: Growth and evolution. *Micromachines*, 13(1):60, 2021.

[30] Dingyang Wang, Sungwon Yoo, and Sung Ho Cho. Experimental comparison of IR-UWB radar and FMCW radar for vital signs. *Sensors*, 20(22):6695, 2020.

[31] Kawon Han and Songcheol Hong. Phase-extraction method with multiple frequencies of FMCW radar for human body motion tracking. *IEEE Microwave and Wireless Components Letters*, 30(9):927–930, 2020.

[32] Nakorn Kumchaiseemak, Itthi Chatnuntawech, Surat Teerapittayanon, Palakon Kotchapansompote, Thitikorn Kaewlee, Maytus Piriyajitakonkij, Theerawit Wilaiprasitporn, and Supasorn Suwajanakorn. Toward ant-sized moving object localization using deep learning in FMCW radar: A pilot study. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–10, 2022.

[33] Alwin Poulose and Dong Seog Han. UWB indoor localization using deep learning LSTM networks. *Applied Sciences*, 10(18):6290, 2020.

[34] Thien Hoang Nguyen, Thien-Minh Nguyen, and Lihua Xie. Range-focused fusion of camera-IMU-UWB for accurate and drift-reduced localization. *IEEE Robotics and Automation Letters*, 6(2):1678–1685, 2021.

[35] Min-Won Seo and Solmaz S. Kia. Online target localization using adaptive belief propagation in the HMM framework. *IEEE Robotics and Automation Letters*, 7(4):10288–10295, 2022.

[36] Kegen Yu, Kai Wen, Yingbing Li, Shuai Zhang, and Kefei Zhang. A novel NLoS mitigation algorithm for UWB localization in harsh indoor environments. *IEEE Transactions on Vehicular Technology*, 68(1):686–699, 2018.

[37] Ardiansyah Musa, Gde Dharma Nugraha, Hyojeong Han, Deokjai Choi, Seongho Seo, and Juseok Kim. A decision tree-based NLoS detection method for the UWB indoor location tracking accuracy improvement. *International Journal of Communication Systems*, 32(13):e3997, 2019.

[38] Changwei Chen, Chi-Shih Jao, Andrei M. Shkel, and Solmaz S. Kia. UWB sensor placement for foot-to-foot ranging in dual-foot-mounted ZUPT-aided INS. *IEEE Sensors Letters*, 6(2):1–4, 2022.

[39] Jianlin Chen, Devin Raye, Wahab Khawaja, Priyanka Sinha, and Ismail Guvenc. Impact of 3D UWB antenna radiation pattern on air-to-ground drone connectivity. In *Vehicular Technology Conference*, pages 1–5, Porto, Portugal, 2018.

[40] K. Wen, K. Yu, and Y. Li. NLoS identification and compensation for UWB ranging based on obstruction classification. In *European Signal Processing Conference*, pages 2704–2708, Kos island, Greece, 2017.

[41] Niranjini Rajagopal, Patrick Lazik, Nuno Pereira, Sindhura Chayapathy, Bruno Sinopoli, and Anthony Rowe. Enhancing indoor smartphone location acquisition using floor plans. In *ACM/IEEE International Conference on Information Processing in Sensor Networks*, pages 278–289, Porto, Portugal, 2018.

[42] Zhuoqi Zeng, Steven Liu, and Lei Wang. UWB/IMU integration approach with NLOS identification and mitigation. In *2018 52nd Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6, Princeton, NJ, 2018. IEEE.

[43] Ardiansyah Musa, Gde Dharma Nugraha, Hyojeong Han, Deokjai Choi, Seongho Seo, and Juseok Kim. A decision tree-based NLOS detection method for the UWB indoor location tracking accuracy improvement. *International Journal of Communication Systems*, 32(13):e3997, 2019.

[44] Andreas De Preter, Glenn Goysens, Jan Anthonis, Jan Swevers, and Goele Pipeleers. Range bias modeling and autocalibration of an UWB positioning system. In *2019 International Conference on Indoor Positioning and Indoor Navigation*, pages 1–8, Pisa, Italy, 2019.

[45] Julian Blueml, Alessandro Fornasier, and Stephan Weiss. Bias compensated UWB anchor initialization using information-theoretic supported triangulation points. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5490–5496, Xi'an, China, 2021.

[46] Justin Cano, Gael Pages, Eric Chaumette, and Jerome Le Ny. Clock and power-induced bias correction for UWB time-of-flight measurements. *IEEE Robotics and Automation Letters*, 2022.

[47] Seong Yun Cho. Two-step calibration for UWB-based indoor positioning system and positioning filter considering channel common bias. *Measurement Science and Technology*, 30(2):025003, 2018.

[48] Jianan Zhu and Solmaz S. Kia. Cooperative localization under limited connectivity. *IEEE Transactions on Robotics*, 35(6):1523–1530, 2019.

[49] Bas van der Heijden, Anton Ledergerber, Rajan Gill, and Raffaello D'Andrea. Iterative bias estimation for an ultra-wideband localization system. *IFAC-PapersOnLine*, 53(2):1391–1396, 2020.

[50] Wenda Zhao, Abhishek Goudar, and Angela P. Schoellig. Finding the right place: Sensor placement for UWB time difference of arrival localization in cluttered indoor environments. *IEEE Robotics and Automation Letters*, 7(3):6075–6082, 2022.

[51] Jeppe Bro Kristensen, Michel Massanet Ginard, Ole Kiel Jensen, and Ming Shen. Non-line-of-sight identification for UWB indoor positioning systems using support vector machines. In *2019 IEEE MTT-S International Wireless Symposium (IWS)*, pages 1–3, Guangzhou, China, 2019. IEEE.

[52] Hongchao Yang, Yunjia Wang, Chee Kiat Seow, Meng Sun, Minghao Si, and Lu Huang. UWB sensor-based indoor LOS/NLOS localization with support vector machine learning. *IEEE Sensors Journal*, 23(3):2988–3004, 2023.

[53] Rui Qi, Xiuping Li, Yi Zhang, and Yubing Li. Multi-classification algorithm for human motion recognition based on IR-UWB radar. *IEEE Sensors Journal*, 20(21):12848–12858, 2020.

[54] Lorenz Schmid, David Salido-Monzú, and Andreas Wieser. Accuracy assessment and learned error mitigation of UWB ToF ranging. In *2019 International Conference on Indoor Positioning and Indoor Navigation*, pages 1–8, Pisa Italy, 2019.

[55] Bo Yang, Jun Li, Zhanpeng Shao, and Hong Zhang. Self-supervised deep location and ranging error correction for UWB localization. *IEEE Sensors Journal*, 23(9):9549–9559, 2023.

[56] Xiaogang Deng, Zhiyuan Ping, and Rui Sun. UWB NLOS recognition based on improved convolutional neural network assisted by wavelet analysis and gramian angular field. *IEEE Sensors Journal*, 23(14):16384–16392, 2023.

[57] Víctor Miramá, Alfonso Bahillo, Víctor Quintero, and Luis Enrique Díez. NLOS detection generated by body shadowing in a 6.5 GHz UWB localization system using machine learning. *IEEE Sensors Journal*, 23(17):20400–20411, 2023.

[58] Valentín Barral, Carlos J Escudero, José A García-Naya, and Roberto Maneiro-Catoira. NLoS identification and mitigation using low-cost UWB devices. *Sensors*, 19(16):3464, 2019.

[59] Wenda Zhao, Jacopo Panerati, and Angela P Schoellig. Learning-based bias correction for time difference of arrival ultra-wideband localization of resource-constrained mobile robots. *IEEE Robotics and Automation Letters*, 6(2):3639–3646, 2021.

[60] Simone Angarano, Vittorio Mazzia, Francesco Salvetti, Giovanni Fantin, and Marcello Chiaberge. Robust ultra-wideband range error mitigation with deep learning at the edge. *Engineering Applications of Artificial Intelligence*, 102:104278, 2021.

[61] Mohammed Ayman Shalaby, Charles Champagne Cossette, James Richard Forbes, and Jerome Le Ny. Calibration and uncertainty characterization for ultra-wideband two-way-ranging measurements. *arXiv preprint arXiv:2210.05888*, 2022.

[62] Anton Ledergerber and Raffaello D'andrea. Calibrating away inaccuracies in ultra wideband range measurements: A maximum likelihood approach. *IEEE Access*, 6:78719–78730, 2018.

[63] Ahmed MM Almassri, Natsuki Shirasawa, Amarbold Purev, Kaito Uehara, Wataru Oshiumi, Satoru Mishima, and Hiroaki Wagatsuma. Artificial neural network approach to guarantee the positioning accuracy of moving robots by using the integration of IMU/UWB with motion capture system data fusion. *Sensors*, 22(15):5737, 2022.

[64] Bahador Khaleghi, Alaa Khamis, Fakhreddine O Karray, and Saiedeh N Razavi. Multisensor data fusion: A review of the state-of-the-art. *Information fusion*, 14(1):28–44, 2013.

[65] S. J. Juliera and J. K. Uhlmannb. Using covariance intersection for SLAM. *Robotics and Autonomous Systems*, 55:3–20, 2007.

[66] Reza Olfati-Saber. Kalman-consensus filter : Optimality, stability, and performance. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 7036–7042, Shanghai, China, 2009.

[67] Chee-Yee Chong, Shozo Mori, Felix Govaers, and Wolfgang Koch. Comparison of tracklet fusion and distributed Kalman filter for track fusion. In *17th International Conference on Information Fusion (FUSION)*, pages 1–8, Salamanca, Spain, 2014.

[68] Solmaz S Kia, Jonathan Hechtbauer, David Gogokhiya, and Sonia Martínez. Server-assisted distributed cooperative localization over unreliable communication links. *IEEE Transactions on Robotics*, 34(5):1392–1399, 2018.

[69] S.J. Julier and J.K. Uhlmann. A non-divergent estimation algorithm in the presence of unknown correlations. In *Proceedings of the 1997 American Control Conference (Cat. No.97CH36041)*, volume 4, pages 2369–2373 vol.4, Albuquerque, NM, 1997.

[70] Zili Deng, Peng Zhang, Wenjuan Qi, Jinfang Liu, and Yuan Gao. Sequential covariance intersection fusion Kalman filter. *Information Sciences*, 189:293–309, 2012.

[71] Jinwen Hu, Lihua Xie, and Cishen Zhang. Diffusion Kalman filtering based on covariance intersection. *IEEE Transactions on Signal Processing*, 60(2):891–902, 2012.

[72] Ondrej Hlinka, Ondrej Slučiak, Franz Hlawatsch, and Markus Rupp. Distributed data fusion using iterative covariance intersection. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1861–1865, Florence, Italy, 2014. IEEE.

[73] P. O. Arambel, C. Rago, and R. K. Mehra. Covariance intersection algorithm for distributed spacecraft state estimation. In *American Control Conference*, pages 4398–4403, Arlington, Virginia, USA, 2001.

[74] L. C. Carrillo-Arce, E. D. Nerurkar, J. L. Gordillo, and S. I. Roumeliotis. Decentralized multi-robot cooperative localization using covariance intersection. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1412–1417, Tokyo, Japan, 2013.

[75] H. Li and F. Nashashibi. Cooperative multi-vehicle localization using split covariance intersection filter. *IEEE Intelligent Transportation Systems Magazine*, 5(2):33–44, 2013.

[76] D. Marinescu, N. O'Hara, and V. Cahill. Data incest in cooperative localisation with the common past-invariant ensemble Kalman filter. In *IEEE Int. Conf. on Information Fusion*, pages 68–76, Istanbul, Turkey, 2013.

[77] J. Sijs, M. Lazar, and P.P.J.v.d. Bosch. State fusion with unknown correlation: Ellipsoidal intersection. In *Proceedings of the 2010 American Control Conference*, pages 3992–3997, Maryland, MD, USA, 2010.

[78] Benjamin Noack, Joris Sijs, Marc Reinhardt, and Uwe D Hanebeck. Decentralized data fusion with inverse covariance intersection. *Automatica*, 79:35–41, 2017.

[79] M. Zarei-Jalalabadi, S. M. B. Malaek, and S. S. Kia. A track-to-track fusion method for tracks with unknown correlations. *IEEE Control Systems Letters*, 2(2):189–194, 2018.

[80] S. Leonardos and K. Daniilidis. A game-theoretic approach to robust fusion and Kalman filtering under unknown correlations. In *American Control Conference*, pages 2568–2573, Seattle, Washington, USA, 2017.

[81] Hanjun Dai, Elias B. Khalil, Yuyu Zhang, Bistra Dilkina, and Le Song. Learning combinatorial optimization algorithms over graphs, 2018.

[82] Lu Liu, Yu Cheng, Lin Cai, Sheng Zhou, and Zhisheng Niu. Deep learning based optimization in wireless network. In *2017 IEEE International Conference on Communications (ICC)*, pages 1–6, Paris, France, 2017.

[83] B. Abbasi, T. Babaei, Z. Hosseinifard, and K. Smith-Miles. Predicting solutions of large-scale optimization problems via machine learning: A case study in blood supply chain management. *Computers & Operations Research*, 119:104941, 2020.

[84] Jewon Eom, Hyowon Kim, Sang Hyun Lee, and Sunwoo Kim. DNN-assisted cooperative localization in vehicular networks. *Energies*, 12(14), 2019.

[85] Jianan Zhu and Solmaz S Kia. Learning-based measurement scheduling for loosely-coupled cooperative localization. *IEEE Robotics and Automation Letters*, 7(3):6313–6319, 2022.

[86] Yuan Zhuang, Jun Yang, You Li, Longning Qi, and Naser El-Sheimy. Smartphone-based indoor localization with bluetooth low energy beacons. *Sensors*, 16(5):596, 2016.

[87] David Gualda, Jesús Ureña, José Alcalá, and Carlos Santos. Calibration of beacons for indoor environments based on a digital map and heuristic information. *Sensors*, 19(3):670, 2019.

[88] Pablo Corbalán, Gian Pietro Picco, Martin Coors, and Vivek Jain. Self-localization of ultra-wideband anchors: From theory to practice. *IEEE Access*, 11:29711–29725, 2023.

[89] Camellia S Mouhammad, Ahmed Allam, Mohamed Abdel-Raouf, Ehab Shenouda, and Maha Elsabrouty. BLE indoor localization based on improved RSSI and trilateration. In *2019 7th International Japan-Africa Conference on Electronics, Communications, and Computations,(JAC-ECC)*, pages 17–21. IEEE, 2019.

[90] Bo Yang, Luyao Guo, Ruijie Guo, Miaomiao Zhao, and Tiantian Zhao. A novel trilateration algorithm for RSSI-based indoor localization. *IEEE Sensors Journal*, 20(14):8164–8172, 2020.

[91] Tian Yang, Adnane Cabani, and Houcine Chafouk. A survey of recent indoor localization scenarios and methodologies. *Sensors*, 21(23):8086, 2021.

[92] Dominik Csík, Peter Sarcevic, Richard Pesti, and Ákos Odry. Comparison of different radio communication-based technologies for indoor localization using trilateration. In *2023 IEEE 17th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, pages 487–492. IEEE, 2023.

[93] Yishan Su, Lei Guo, Zhigang Jin, and Xiaomei Fu. A mobile-beacon-based iterative localization mechanism in large-scale underwater acoustic sensor networks. *IEEE Internet of Things Journal*, 8(5):3653–3664, 2020.

[94] Alexandru-Tudor Popovici, Constantin-Catalin Dosoftei, and Cristina Budaciu. Kinematics calibration and validation approach using indoor positioning system for an omnidirectional mobile robot. *Sensors*, 22(22):8590, 2022.

[95] Vaishali R Kulkarni. Comparative analysis of static and mobile anchors in sensor localization. In *2023 International Conference on Device Intelligence, Computing and Communication Technologies, (DICCT)*, pages 115–120, 2023.

[96] Qinghua Luo, Kexin Yang, Xiaozhen Yan, Jianfeng Li, Chenxu Wang, and Zhiquan Zhou. An improved trilateration positioning algorithm with anchor node combination and k-means clustering. *Sensors*, 22(16):6085, 2022.

[97] Ahmed Mahmoud, Pedro Coser, Hamza Sadruddin, and Mohamed Atia. Ultra-wideband automatic anchor's localization for indoor path tracking. In *2022 IEEE Sensors*, pages 1–4. IEEE, 2022.

[98] Peter H Zipfel. Cadac: multi-use architecture for constructive aerospace simulations. *The Journal of Defense Modeling and Simulation*, 9(2):129–145, 2012.

[99] Peter Zipfel. *Modeling INS/GPS/Star-Tracker in 6DoF; Simulating N&G&C of a Three Stage Solid Rocket booster in CADAC++*. GloryGram Books, 01 2015.

[100] Yaakov Bar-Shalom, X Rong Li, and Thiagalingam Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[101] Siddhartha Chib and Edward Greenberg. Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4):327–335, 1995.

[102] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

[103] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

[104] Xiaoping Yun, Eric R. Bachmann, Hyatt Moore, and James Calusdian. Self-contained position tracking of human movement using small inertial/magnetic sensor modules. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, Roma, Italy, April 2007.

[105] E. Foxlin. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Computer graphics and applications*, 25(6):38–46, 2005.

[106] T. Brand and R. Phillips. Foot-to-foot range measurement as an aid to personal navigation. In *the 59th Annual Meeting of The Institute of Navigation and CIGTF 22nd Guidance Test Symposium*, Albuquerque, New Mexico, USA, June 2003.

[107] Chi-Shih Jao, Yusheng Wang, and Andrei M. Shkel. Pedestrian inertial navigation system augmented by vision-based foot-to-foot relative position measurements. In *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Portland, Oregon, USA, April 2020.

[108] Sina Askari, Chi-Shih Jao, Yusheng Wang, and Andrei M Shkel. A laboratory testbed for self-contained navigation. In *2019 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, Naples, Florida, USA, April 2019.

[109] Yusheng Wang, Chi-Shih Jao, and Andrei M Shkel. Scenario-dependent ZUPT-aided pedestrian inertial navigation with sensor fusion. *Journal of Gyroscopy and Navigation*, 12(1):1–16, 2021.

[110] Qinglin Tian, Kevin I-Kai Wang, and Zoran Salcic. An INS and UWB fusion approach with adaptive ranging error mitigation for pedestrian tracking. *IEEE Sensors Journal*, 20(8):4372–4381, 2020.

[111] Y Jay Guo and Stephen K Barton. *Fresnel zone antennas*. Springer Science & Business Media, 2013.

[112] Abbas Albaidhani, Antoni Morell, and Jose Lopez Vicario. Ranging in uwb using commercial radio modules: Experimental validation and nlos mitigation. In *International Conference on Indoor positioning and indoor navigation (IPIN)*, pages 1–7, Alcalá de Henares, Madrid, Spain, 2016.

[113] J Zhu and SS Kia. UWB-based infrastructure-free cooperative navigation with NLoS ranging bias compensation for indoor pedestrian geolocation. In *International Technical Meeting of The Institute of Navigation*, pages 334–344, Reston, Virginia, USA, 2019.

[114] Jianan Zhu and Solmaz S Kia. Bias compensation for UWB ranging for pedestrian geolocation applications. *IEEE Sensors Letters*, 3(9):1–4, 2019.

[115] A Alarifi, A Al-Salman, M Alsaleh, A Alnafessah, S Al-Hadhrami, MA Al-Ammar, et al. Decawave DWM1000 module. *Available online: https://www.decawave.com/product/dwm1000-module/*.

[116] Y. Bar-Shalom, X. Li, and T. Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[117] Paul Stoffregen. Teensy 3.2 datasheet. `https://www.pjrc.com/teensy/techspecs.html`, 2016. Accessed: March 17, 2023.

[118] Jean-Marie Dufour and Julien Neves. Finite-sample inference and nonstandard asymptotics with Monte Carlo tests and R. In *Handbook of statistics*, volume 41, pages 3–31. Elsevier, 2019.

[119] Rabiya Khalid and Nadeem Javaid. A survey on hyperparameters optimization algorithms of forecasting models in smart grid. *Sustainable Cities and Society*, 61:102275, 2020.

[120] T. Li, Y. Xin, Y. Song, E. Song, and H. Fan. Some statistic and information-theoretic results on arithmetic average density fusion, 2021. `https://arxiv.org/pdf/2110.01440.pdf`.

[121] Simon Julier and Jeffrey K Uhlmann. General decentralized data fusion with covariance intersection. In *Handbook of multisensor data fusion*, pages 339–364. CRC Press, 2017.

[122] Cvxopt. `http://cvxopt.org`. [Online; accessed 14-August-2014].

[123] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[124] R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, 1991.

[125] David G Luenberger, Yinyu Ye, et al. *Linear and nonlinear programming*, volume 2. Springer, 1984.

[126] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

[127] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[128] Andreas Haeberlen, Eliot Flannery, Andrew M Ladd, Algis Rudys, Dan S Wallach, and Lydia E Kavraki. Practical robust localization over large-scale 802.11 wireless networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 70–84, New York, New York, USA, 2004.

[129] Pooyan Shams Farahsari, Amirhossein Farahzadi, Javad Rezazadeh, and Alireza Bagheri. A survey on indoor positioning systems for IoT-based applications. *IEEE Internet of Things Journal*, 9(10):7680–7699, 2022.

[130] S. S. Kia, S. Rounds, and S. Martínez. Cooperative localization for mobile agents: a recursive decentralized algorithm based on Kalman filter decoupling. *IEEE Control Systems Magazine*, 36(2):86–101, 2016.

[131] Changwei Chen and Solmaz S Kia. Cooperative localization using learning-based constrained optimization. *IEEE Robotics and Automation Letters*, 7(3):7052–7058, 2022.