Johan Nicolas Suarez Lojan

Improving Accuracy of MCPD Distance Measurements

Master's thesis in Embedded Computing Systems
Supervisor: Kimmo Kansanen
Co-supervisor: Carsten Wulff
July 2023

NTNU
Norwegian University of
Science and Technology

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electronic Systems

Johan Nicolas Suarez Lojan

Improving Accuracy of MCPD Distance Measurements

Master's thesis in Embedded Computing Systems
Supervisor: Kimmo Kansanen
Co-supervisor: Carsten Wulff
July 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electronic Systems

NTNU
Norwegian University of
Science and Technology

ABSTRACT

The distance measurements obtained with Multi-Carrier Phase Difference algorithms exhibit a bias in the estimated values. This work aims to explore ways to reduce the effects of the bias in the measurements. Initially, an overview of different positioning techniques and previous findings with MCPD algorithms is presented. Next, a comprehensive bias model is developed to enhance the understanding of the underlying problem. Subsequently, the establishment of a network setup allows the collection of the necessary data to estimate the value of the bias. A special configuration of the devices is necessary to perform network measurements. A regression analysis to approximate the bias value is done using the Least Squares approach with the data collected by the measuring network. By incorporating the estimated bias into distance measurements, improvements in accuracy are tested by the values of the Mean Average Error and Mean Bias Error on the dataset.

PREFACE

This work concludes the Master of Science (MSc) degree of the European Master in Embedded Computing Systems at the Department of Electronic Systems at the Norwegian University of Science and Technology (NTNU) and the Department of Electrical Engineering and Information Technology at the University of Kaiserslautern-Landau (RPTU). This thesis was written during the Spring Semester of 2023 and it is a following work to the specialization project that pursues a deeper understanding of the process and techniques for distance estimation using Bluetooth radios. The master thesis used the findings in the specialization project as a foundation to try to further improve the accuracy of wireless distance estimation algorithms.

Throughout the course of this research, many individuals have played a significant role in shaping the direction and outcome of this work. I would like to express my gratitude to my supervisor, Kimmo Kansanen, and my co-supervisor, Carsten Wulff for their invaluable guidance and support. Their expertise, feedback, and insightful suggestions have been instrumental in shaping this thesis.

I am always grateful to my family for their unfailing support, encouragement, and understanding throughout this challenging journey. I would like to thank my friends and fellow students I met during my time at RPTU and NTNU. It has been two great years with a lot of fun and enriching experiences.

Johan N. Suarez L.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

ADC	Analog-Digital Converter
ADoA	Angle Difference of Arrival
AoA	Angle of Arrival
BLE	Bluetooth Low Energy
CSI	Channel State Information
CSV	Comma Separated Values
DK	Development Kit
DoA	Direction of Arrival
FTM	Fine Time Measurements
GPS	Global Positioning System
I/Q	In Phase and Quadrature
IFFT	Inverse Fast Fourier Transform
IoT	Internet of Things
ISM	Industrial, Scientific, and Medical
JSON	JavaScript Object Notation
KKT	Karush-Kuhn-Tucker
LED	Light Emitting Diode
LoS	Line of Sight
LSE	Least Squares Estimation
MAE	Mean Absolute Error
MBE	Mean Bias Error
MCPD	Multi-Carrier Phase Difference

MCPD-LR	Multi-Carrier Phase Difference with Linear Regression
MUSIC	Multiple Signal Classification
NLOS	Non-Line of Sight
NNLS	Non-Negative Least Squares
OFDM	Orthogonal Frequency Division Multiplexing
PDoA	Phase Difference of Arrival
PoA	Phase of Arrival
PWM	Pulse Width Modulation
RAM	Random Access Memory
RF	Radio Frequency
RFID	Radio Frequency Identification
RSSI	Received Signal Strength Indicator
RTT	Round Trip Time
RTToA	Round Trip Time of Arrival
SDK	Software Development Kit
SoC	System on Chip
TDoA	Time Difference of Arrival
ToA	Time of Arrival
ToF	Time of Flight
TWCR	Two-way Channel Reconstruction
UART	Universal Asynchronous Receiver / Transmitter
WSN	Wireless Sensor Network

CHAPTER ONE

INTRODUCTION

Positioning services and applications are widely used and have improved mobility, safety, efficiency, and productivity in various sectors. One of the most common examples is Global Positioning Systems (GPS) which nowadays is widely used in outdoor navigation. However, there exists the need for an alternative GPS in environments where its signal is attenuated or in applications where the accuracy requirements are higher, and the degradation of satellite signals is an issue. GPS-enabled smartphones are typically accurate to within a 4.9 m radius under open sky [1], but some applications require positioning accuracies far beyond those that GPS can provide. For example in an indoor navigation application, five meters can be the difference between two different rooms, another example could be in asset tracking of objects standing in close proximity to each other. An Indoor Positioning System applicable to most indoor environments and applications would have mean accuracies below one or two meters [2].

Indoor Positioning systems can use radio frequency signals such as Wi-Fi or Bluetooth Low Energy (BLE) to perform ranging. BLE is widely used in ubiquitous computing and in many applications of the Internet of Things (IoT) due to the advantages it presents. BLE emitters or beacons are portable, battery-powered, small, lightweight, easily deployable, and they have low energy consumption [3]. Therefore the importance of improving the accuracy of the current technologies appears. BLE can be used with different localization techniques using various parameters such as Received Signal Strength Indicator (RSSI), Angle of Arrival (AoA), Time of Flight (ToF), Phase Difference of Arrival (PDoA), or a combination of them for the distance estimation [4] [5]. The phase measurements provide the possibility of achieving high accuracy, as reported by [5] [6] [7]. Multicarrier Phase Difference (MCPD) is a technique that processes the PDoA measurements and relates them to the distance that separates two devices.

During the Specialization project work previous to the thesis, an evaluation was conducted on various algorithms utilizing phase measurement techniques of two Bluetooth radios. The outcomes revealed the presence of a measurement bias, prompting further investigation into effective methods of minimizing this error. This document presents an analysis of the results obtained and proposes the utilization of network measurements as a potential solution to mitigate the introduced bias. Some methods can improve accuracy in similar systems like Fingerprinting and lateration. This work presents a technique using Euclidean geometry and a Least Squares optimization approach to improve the distance estimates.

1.0.1 Problem Definition

The distance estimations using MCPD algorithms present a bias in the values. Finding the source of the bias in the estimation process and ways of reducing its value will improve the accuracy of the MCPD ranging. This thesis will seek the possibility of developing a comprehensive model of the bias to see if its value can be estimated. Subsequently, A Least Squares analysis could allow finding an optimal value and requires a way of obtaining redundant information on the measurements, for this, a network setup is proposed. If the bias value can be estimated the next step is to evaluate how much the accuracy is improved on the distance measurements using the selected method. By addressing these questions, this work aims to provide valuable insights and practical solutions for mitigating bias in MCPD-based distance measurements.

CHAPTER

TWO

RADIO DISTANCE ESTIMATION

The distance estimation or ranging in Indoor Positioning systems can be done by many technologies. These classifications include optical technologies that utilize visible light, sound-based technologies that make use of ultrasound, RF-based technologies like Bluetooth Low Energy and Wi-Fi, as well as methods that rely on naturally occurring signals, such as the Earth's natural magnetic field. By leveraging these diverse technological approaches, Indoor Positioning systems are able to provide accurate and reliable positioning information within indoor environments [3].

2.1 Principles of Radio Distance estimation

Radio Distance estimation or Ranging is the process of determining the distance from one location or position to another location or position using Radio signals. The distance estimation is based on different parameters. In Figure 2.1.1 a classification of the different parameters for indoor localization is presented. Indoor localization can be based on two primary parameters: distance-based and direction-based. Our focus lies predominantly on the first set of parameters, and thus a comprehensive exploration and analysis of distance-based parameters will be presented in the subsequent sections. Distance-based technologies can use signal parameters such as Channel State Information (CSI), Received Signal Strength (RSS), Phase Difference of Arrival (PDOA), or Phase of Arrival (POA). They can also be based on time parameters like Round Trip Time (RTT), Time of Arrival (TOA), and Time Difference of Arrival (TDOA). A disadvantage of the latter is that they require precise time synchronization. Direction-based methods can use angle of arrival (AOA), angle difference of arrival (ADOA), or Direction of Arrival (DOA) and they do not require time synchronization between nodes [8].

2.1.1 Time of Arrival (TOA) and Time Difference of Arrival (TDOA)

TOA (Time of Arrival) calculates the time it takes for a wireless signal to reach the receiver. It requires at least two or three reference nodes in line of sight (LoS) with the target to achieve high position accuracy. This method is susceptible to

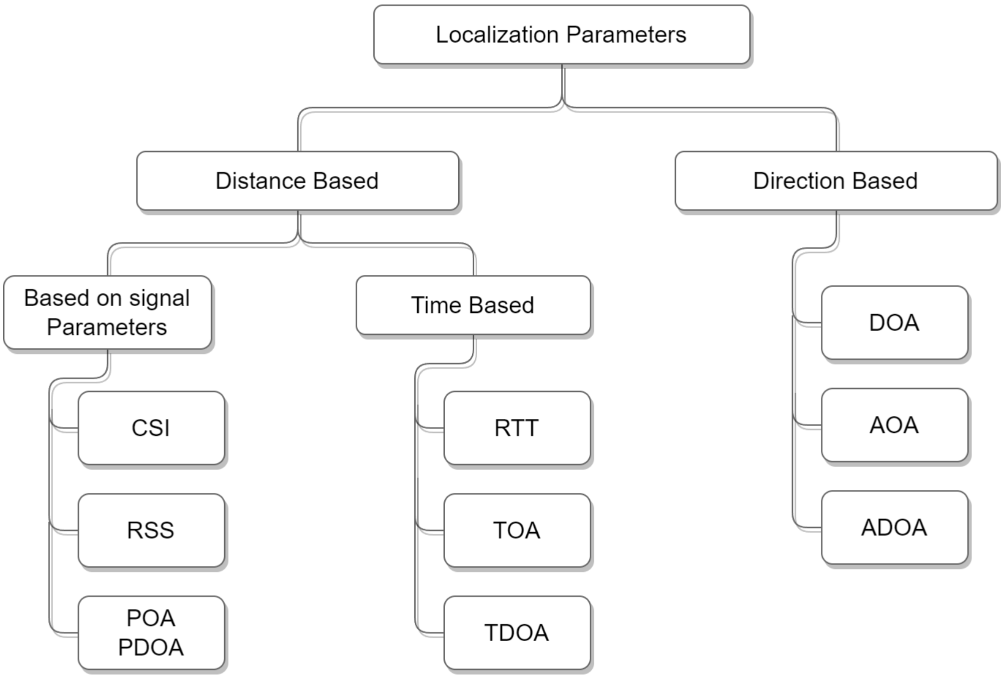

Figure 2.1.1: Localization system parameters, taken from [8]

multipath and additive noise and highly depends on synchronization. To overcome these issues, the TDOA (Time Difference of Arrival) method, along with the RTOA (Round Trip Time of Arrival) method, is employed. TDOA relies on determining the difference in arrival times between anchor nodes to calculate the distance. It requires a minimum of three anchor nodes with known coordinates for accurate object positioning. TDOA accuracy is also subject to the synchronization between the anchor nodes and the taken timestamp. [8], [9].

2.1.2 Round Trip Timing (RTT)

Similar to the previous technique RTT-based method is used in wireless networks with the use of fine time measurements (FTM) of the round-trip time (RTT) between a smartphone and an access point. This approach offers advantages such as clock synchronization independence, high reliability, and large range estimation. However, it is subject to limitations such as reflection, fading, shadowing, unstable clock speed, and multiple simultaneous inquiries of FTM [8].

2.1.3 Received Signal Strength (RSSI)

RSSI is a metric commonly used to estimate the distance between nodes based on signal power loss. It can be employed in trilateration-based methods to achieve distance estimation using a few nodes. RSSI techniques can be categorized as range-based relying on path loss models and range-free utilizing fingerprinting databases. The fingerprinting technique is higher in accuracy and can be used for various indoor environments [8]. However, RSSI measurement can cause an error due to environmental effects, in cluttered environments for example, the attenuation of the signal could be poorly correlated with distance, particularly in non-line-of-sight (NLOS) channel conditions, resulting in inaccurate distance estimates [10].

2.1.4 Channel State Information (CSI)

New wireless communication systems, such as 4G LTE and Wi-Fi, employ orthogonal frequency division multiplexing (OFDM) to transmit data on multiple subcarriers. Channel State Information (CSI) in the physical layer plays a crucial role in these systems, providing detailed frequency response information for each subcarrier. Compared to traditional Received Signal Strength Indicator (RSSI), CSI captures more data and offers finer granularity, reflecting characteristics like amplitude and phase. CSI-based methods for localization, utilizing techniques like fingerprint matching, triangulation, and trilateration, demonstrate higher accuracy and stability than RSSI-based methods [8], [11].

2.1.5 Phase of Arrival (POA) and Phase Difference of Arrival (PDOA)

POA-based approaches require Line of Sight (LoS) for optimal accuracy. On the other hand, PDOA (Phase Difference of Arrival) ranging measurement utilizes the phase difference between anchor nodes or readers and the tag to calculate distance [8]. PDOA techniques were originally introduced for distance estimation in radar systems and have been recently rediscovered to improve the localization accuracy of RFID and WSN systems [10]. [5] shows the possibility of using PDOA parameters for distance estimation using BLE radios. Phase measurements have the potential for being a cheap and easy-to-deploy distance estimation method and it has the capability of reaching high accuracy [7].

2.2 Bluetooth Distance Estimation

Bluetooth Low Energy was introduced in 2009 as an extension to Bluetooth Classic. It operates within the 2.4 GHz band and utilizes 40 channels, which are divided into primary advertising channels and secondary advertising and data channels. BLE 5.1 introduces features like absolute positioning in three-dimensional space through angle of arrival and angle of departure, as well as phase difference of arrival. Providing precise direction in addition to distance-only information traditionally provided by received signal strength [3], [12], [13]. The advantage of PDOA for ranging is that the phase shift allows precise distance measurement as long as the phase shift is accurate, another advantage is that precise clock synchronization is not needed [10]. The objective of this study is to explore the potential of utilizing phase measurements for achieving precise distance readings among multiple devices.

2.2.1 Active Reflector Procedure

The Active Reflector Procedure allows the distance estimation between two radio transceivers by sampling the phase angle φ of an incoming Continuous Wave signal.

The reference signal is the Local Oscillator. By sampling φ on both transceivers, for multiple frequencies, synchronization is not needed [14]. In the Active Reflector, two roles can be defined, these are Initiator and Reflector. The initiator is the device that wants to perform the ranging.

The procedure goes as follows: First, the initiator starts the process by sending a tone in the first frequency. Next, the reflector comes into play, performing the phase measurement. Once the measurement is obtained, the reflector transmits both the measurements and a tone of the same frequency back to the initiator. The initiator then changes the frequency and this process repeats many times at different frequencies e.g. channels of Bluetooth. In the final stage, the initiator receives the transmission, measures the phase, and proceeds to estimate the distance using the measurements obtained at both sides and at all frequencies. A diagram explaining the procedure can be seen in Figure 2.2.1. This step-by-step procedure involves the exchange of messages between two BLE radios through usual data channels using the BLE Protocol Stack.

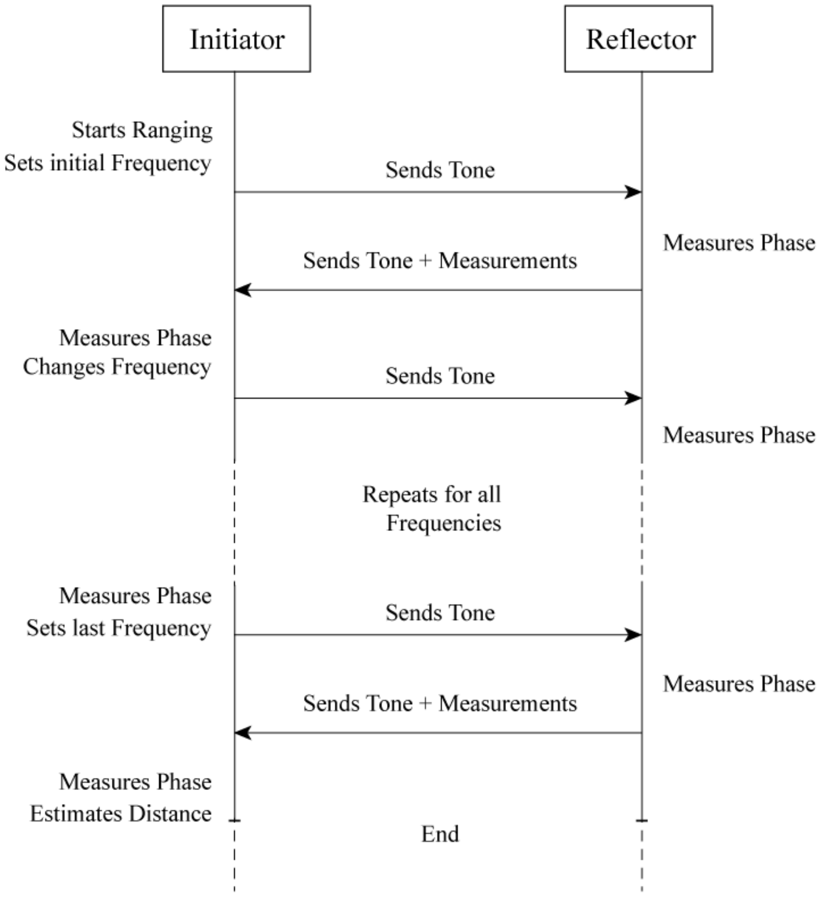

Figure 2.2.1: Active reflector measurement procedure

This procedure can be described in the IQ-domain. Let the signal transmitted by the initiator at a certain tone with frequency f be $\mathrm{e}^{\mathrm{j}\left(\omega_f t+\varphi_I\right)}$, where ω is the frequency of the tone in rad/sec and φ_I is the starting phase at the initiator. The signal goes through the channel $H(f)$. This signal on the Reflector is mixed to base-band with signal $\mathrm{e}^{\mathrm{j}\left(\omega_f t+\varphi_R\right)}$ where φ_R is the phase at the reflector. The phases φ_I and φ_R can have a value between π and $-\pi$. The measurement at the Reflector side corresponds to

$$\begin{aligned} Y_R(f) &= \mathrm{e}^{\mathrm{j}\left(\omega_f t+\varphi_I\right)} \cdot H_R(f) \cdot \mathrm{e}^{-\mathrm{j}\left(\omega_f t+\varphi_R\right)} \\ &= H_R(f) \cdot \mathrm{e}^{\mathrm{j}(\varphi_I-\varphi_R)}, \end{aligned} \tag{2.1}$$

where $Y_R(f)$ is the signal at the Reflector, after that, the Reflector sends a signal at the same tone f, and the Initiator measures $Y_I(f)$:

$$Y_I(f) = H_I(f) \cdot \mathrm{e}^{\mathrm{j}(-\varphi_I+\varphi_R)} \tag{2.2}$$

The Two-way Channel Reconstruction (TWCR) is obtained by multiplying both values

$$H_f^2 = Y_I(f) \cdot Y_R(f) = H_I(f) \cdot H_R(f) \tag{2.3}$$

where the measurements taken correspond to $Y_I(f)$ and $Y_R(f)$ and H_f^2 is defined as the squared transfer function of the channel. The multiplication gets rid of the phases φ_R and φ_I leaving only the phase rotation introduced by the channel

$$H(f) = h\mathrm{e}^{-\mathrm{j}\omega_f\tau}, \tag{2.4}$$

where h is the amplitude, ω_f the angular frequency of the tone and τ is the delay of the channel [6].

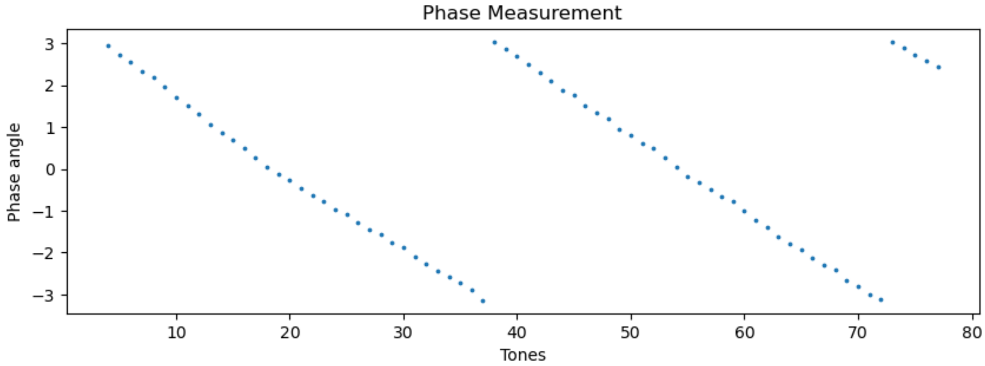

Figure 2.2.2: Measured wrapped phase for a distance of 3.6m

In the case of the measurement with a BLE radio, Bluetooth operates in the Industrial Scientific and Medical (ISM) band from 2.40GHz to 2.41GHz. For the ranging process, BLE uses 74 tones at different frequencies f with a spacing of 1MHz. The frequency changes are driven by an adaptive frequency hopping algorithm. In practice, approximately 3 complete-ranging processes can be done per second. By taking different measurements over different frequencies of the Bluetooth spectrum the graph Figure 2.2.2 is obtained. In this figure, it is noticeable that the values wrap around 2π, this will be resolved later by the ranging algorithms [14]. The graph corresponds to a ranging performed at a distance of 3.6m.

2.2.2 Distance Estimation Algorithms

In order to infer the distance from the phase difference measurements an algorithm has to be applied. Some algorithms are Multi-Carrier Phase Difference (MCPD), Inverse Fast Fourier Transform (IFFT), MUltiple SIgnal Classification (MUSIC), and others. MCPD is an algorithm described by [7], [5], and it estimates the distance in function of the slope of the unwrapped phase. The use of MCPD with a linear regression algorithm can produce more accurate results[15].

IFFT algorithm considers that (2.3) is the phase of a frequency response measurement of the communication channel. By transforming the frequency response back to the time domain, the impulse response is obtained. The maximum peak in the impulse response indicates the channel's propagation delay and by multiplying the delay with the propagation speed of the signal c, the distance between the devices is determined [14].

MUSIC algorithm performs an eigen-based subspace decomposition method utilized for frequency estimation of complex sinusoids in the presence of additive white noise, in this case, it is used to identify the LOS component where resolution bins in the 2.4 GHz ISM band are small and likely to contain multiple Non-LOS components alongside the LOS [6]. MUSIC presents a better performance in real scenarios with multipath.

The linear regression algorithm performs better than the classic MCPD, it presents a smaller standard deviation meaning more accuracy in the estimations, and it is also preferred over MUSIC due to computation time, this is why all improvements on the estimated distances will be made on distance values calculated using MCPD with Linear Regression.

2.2.2.1 MCPD

MCPD distance estimation is done based on the principle that the phase shift introduced by a pure Line-of-Sight radio channel on a radio signal is a linear function of both frequency (f) and range (r) [7]. And is given by

$$\varphi(k, r) = \frac{2\pi f r}{c} \bmod 2\pi \tag{2.5}$$

To get rid of the 2π ambiguity, the two-way ranging is used as explained in Section 2.2.1. For the two-way ranging method, the distance can be found using

$$\hat{r} = \frac{c}{4\pi k \Delta f} \cdot \Delta\varphi, \tag{2.6}$$

where c is the speed of light in a vacuum, k is the index of the tone sent, Δf is the spacing between the tones (for BLE 1MHz), and $\Delta\varphi$ is the difference of the phase measurement between adjacent tones. The phases on (2.3) are wrapped and must be unwrapped in order to find $\Delta\varphi$.

A Script in Python for the MCPD estimation is presented in Apendix A1

2.2.2.2 MCPD with Linear Regression

The linear regression is an improvement on the MCPD procedure by using the slope of a fitted line to estimate the distance. This linear regression is done to fit the unwrapped phase to a linear curve, this increases the accuracy relating all measurements to only a single value, an example of this method is presented in Figure 2.2.3. The slope of the curve can be used to calculate the distance using a simplification of the equation (2.6),

$$\hat{r} = -\frac{c \cdot m}{4\pi \Delta f}, \tag{2.7}$$

where m is the slope of the fitted line. A Python Script for the MCPD-LR algorithm is shown in Appendix A2.

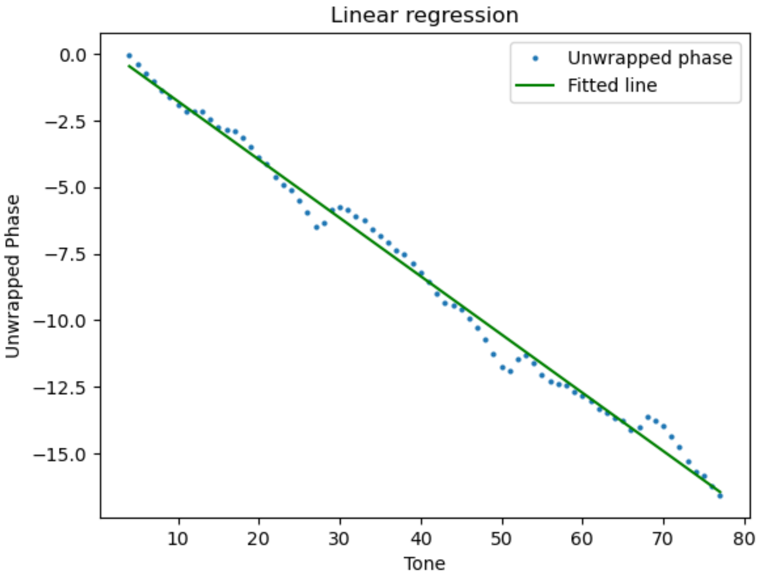

Figure 2.2.3: Fitted line over a set of unwrapped phase measurements

2.3 Reviewing Past MCPD Measurements

During the specialization project done by the author during the Fall semester of 2022, different ranging algorithms were tested to observe the effects of fading and multipath on distance estimation. Three algorithms, namely MCPD, MCPD-LR, and MUSIC were implemented and their estimated distances were compared.

2.3.1 Measurement Setup

The hardware used was the development kit nRF52833 from Nordic Semiconductor. The nRF52833 is a general-purpose multiprotocol SoC with a Bluetooth Direction Finding capable radio, is built around a 64 MHz Arm Cortex-M4, has 512 KB flash and 128 KB RAM memory [16]. The initiator was connected to a laptop where the phase measurements stored as a JSON file were stored using a Python script and UART communication. The development kit was configured using the Distance Measurement Library by Nordic Semiconductors. The measurements were carried inside an electromagnetic anechoic chamber, which is a shielded room that has radio-wave absorbing material applied to the walls, ceiling, and floor and isolates the devices from any interference, shadowing, or multipath effects.

The Distance Measurement library provides an easy-to-use interface for measuring distances. To compute the distance it can use the measured differential RF physical channel frequency response (MCPD mode) or the real-time flight of packets (RTT mode) [17]. The library also can be used to obtain the raw I/Q measurements, this is in-phase (I) and quadrature (Q) measurements.

2.3.2 Insights into Bias

From the results obtained, one particular behavior drew attention. On a set of measurements done in the anechoic chamber, a bias in the values of the distance estimates was present. The bias persisted regardless of the algorithm used to estimate the distance as can be seen in Figure 2.3.1. The red dashed line depicts the real distance between the two devices. There exists a mean absolute error of 0.55m with MCPD, 0.52m with the Linear Regression and, 0.46m with MUSIC. The fact that the measures were made in an anechoic chamber assures ideal propagation conditions. Hence, it is plausible to attribute the bias to potential inaccuracies in the phase measurements.

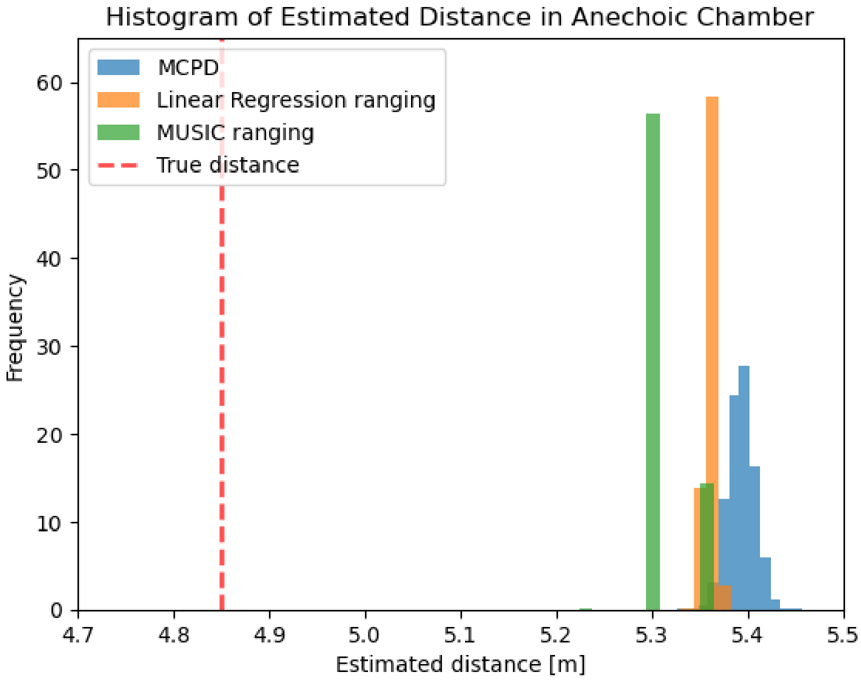

Figure 2.3.1: Distance estimations with various algorithms in an anechoic chamber

CHAPTER

THREE

BIAS MODELING AND ESTIMATION

This section focuses on the domain of bias modeling, aiming to gain a comprehensive understanding of the hardware model and identify the sources from which biases originate. An explanation of the active reflector principle and a model of the ranging process in the IQ domain will be used to develop a working hypothesis for the bias.

3.1 Mathematical Model

3.1.1 BLE System Model

We start with the simplification of the hardware in Figure 3.1.1. Each device is assumed to have a transmitter and receiver filter with an amplitude and phase, we assume that the filters are equal in both devices. From the Initiator a signal goes from $a_I(f)e^{j\left(\varphi_{a_I}\right)}$ then travel across the wireless channel with transfer $H(f)$ and is received by the Reflector through $b_I(f)e^{j\left(\varphi_{b_I}\right)}$. In the other direction, the signal travels through the alternative filters $a_R(f)e^{j\left(\varphi_{a_R}\right)}$ and $b_I(f)e^{j\left(\varphi_{b_I}\right)}$.

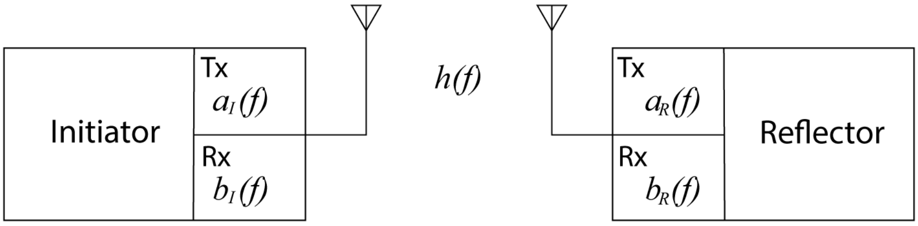

Figure 3.1.1: Communication system diagram

The process for one of the tones with frequency f can be expressed by a modification of equations (2.1) and (2.2), at the reflector side:

$$\begin{aligned} Y_R(f) &= e^{\mathrm{j}\left(\omega_f t+\varphi_I\right)} \cdot H_R(f) \cdot \mathrm{e}^{-\mathrm{j}\left(\omega_f t+\varphi_R\right)} \cdot a_I(f) \cdot e^{j\left(\varphi_{a_I}\right)} \cdot b_R(f) \cdot e^{-j\left(\varphi_{b_R}\right)} \\ &= a_I(f) \cdot b_R(f) \cdot H_R(f) \cdot e^{j\left(\varphi_{a_I}-\varphi_{b_R}+\varphi_I-\varphi_R\right)} \end{aligned} \tag{3.1}$$

And at the initiator, we have:

$$Y_I(f) = a_R(f) \cdot b_I(f) \cdot H_I(f) \cdot e^{j\left(\varphi_{a_R} - \varphi_{b_I} + \varphi_R - \varphi_I\right)} \tag{3.2}$$

The Two-way Channel Reconstruction (TWCR), as explained before, can be obtained by multiplying both signals:

$$\begin{aligned} H^2(f) &= Y_I(f) \cdot Y_R(f) \\ &= a_I(f) \cdot a_R(f) \cdot b_R(f) \cdot b_I(f) \cdot H_I(f) \cdot H_R(f) \cdot e^{j\left(\varphi_{a_I} - \varphi_{b_R} + \varphi_{a_R} - \varphi_{b_I}\right)} \end{aligned} \tag{3.3}$$

As can be seen, the phase of the local oscillator $\varphi_R(f)$ and $\varphi_I(f)$ are eliminated with the multiplication, leaving only the magnitudes and phases of the Rx and Tx filters and the channel response $H(f)$. Leaving the most probable cause for the bias to an effect on the Rx and Tx filters. To further look at this cause a more detailed description of the hardware is presented in the next section.

3.1.2 Hardware Block Diagram

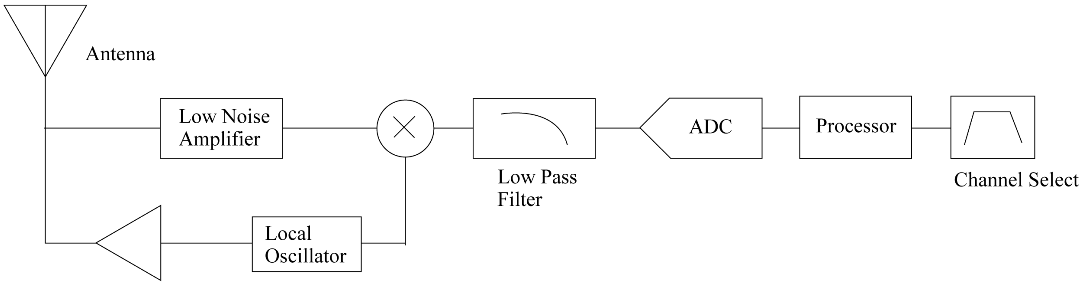

Figure 3.1.2: Hardware block diagram

A block diagram of the hardware is presented in figure 3.1.2. This is a representation of the nRF52833 development kits. Each device is equipped with an antenna that picks up or emits the radio signals, a low noise amplifier that increases the gain of the received signal, a mixer, then a Low Pass Filter, an Analog-Digital Converter (ADC), a Processor that captures the I/Q measurements and a Channel select block. In the bottom part, it has the local oscillator and an operational amplifier for transmitting. The upper section of the diagram is the equivalent of the Rx filter defined in the previous section, and the lower part is the Tx. The portion of the hardware where a delay in the phase measurements could be introduced is the path between the Antenna and the mixer. When the device is receiving a signal, i.e. acting as the reflector, the signal is picked up by the antenna, then it goes through a Low Noise Amplifier where it's mixed with the local oscillator frequency and the phase I/Q measurements are taken in the Processor. A delay of around 2-3ns could be introduced at this step and this represents around 60 to 90 cm at 2.4GHz.

3.1.3 Channel Response

To observe the effect of the magnitude response of the channel on the distance estimations a measure of the channel frequency response was made. The measurement involves two devices in the anechoic chamber at a distance of 4.90m, the raw measurements can be found in Appendix C1.

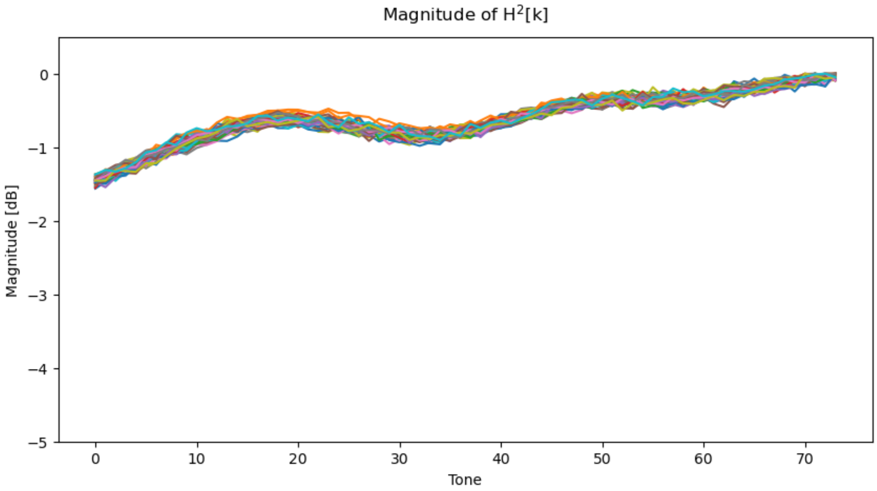

Figure 3.1.3: Magnitude response of the channel in an anechoic chamber

The magnitude of the channel transfer function squared 3.3 is plotted in Figure 3.1.3 in the function of the tone index. This includes the channel response magnitude and the filter's magnitudes $a_I(f)$, $a_R(f)$, $b_R(f)$ and $b_I(f)$, the filter magnitudes are assumed to not be very significant. The magnitude information could also be useful for the distance estimation in scenarios with multipath where the delay spread of the channel allows the correct selection of the LOS path [6], however, in the present model it will not be taken into account and it will be assumed that the magnitude response of the channel is flat. To prove this assumption the impulse response of the channel was obtained and it has been compared to an ideal impulse response of the channel at 4.9m.

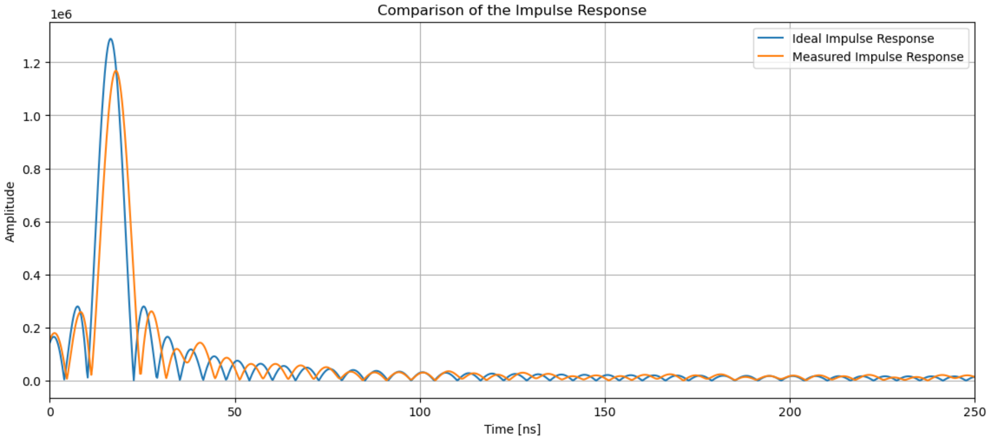

Figure 3.1.4: Comparison of the channel impulse response with flat magnitude response and the measured magnitude response

In Figure 3.1.4 It can be seen that the peak represents the delay of the LOS component on the measured signal at 17.82ns. The delay of the ideal response is 16.35ns this is a difference of 1.46ns that corresponds to a distance of 44cm which corresponds to our assumptions of the bias. The width of the pulses was also measured, with the ideal response being 12.5ns, and for the measured 13.28ns, the difference of 0.78ns is neglected and therefore, the bias can be modeled just as a delay with a flat channel response.

3.1.4 Bias Model

With the previous assumptions and also that only exist a single propagation path in the anechoic chamber, equation (3.3) can be represented as:

$$H^2(f) = H_I(f) \cdot H_R(f) \cdot e^{j\tau_b} \tag{3.4}$$

Where τ_b is the delay introduced, this delay is assumed to have the same value for all frequencies f and it is particular for all devices of the nRF52833 model.

$$\tau_b = \varphi_{a_I} - \varphi_{b_R} + \varphi_{a_R} - \varphi_{b_I} \tag{3.5}$$

To summarize the working model of the bias is a delay that when it is introduced in the phase measurement makes the estimated distance larger than the true distance. By working with the measured distances this value can be estimated as the next sections will show.

3.2 Geometric Considerations of the Measuring Network

To estimate the bias value, a network configuration is proposed, as it provides additional information required to solve for the unknown bias. The concept involves conducting multiple measurements that are interconnected through a system of equations. The network enables the possibility of taking redundant distance measurements. The nRF52833 Development Kits have the capability to establish such a network. With the use of geometric considerations, the network makes use of the redundant measurements to establish an equation system that allows the estimation of unknown distances or in the case of this work for the estimation of the bias.

3.2.1 Distance Addition Postulate

The proposed method to estimate the value of the bias is based on using a congruent Euclidean geometry. In this geometry, the distances between the nodes will be constrained by a set of equations. As a proof of concept, a one-dimension approach will be adopted and can be extended for more complex network geometries. We start our approach with the Distance Addition Postulate, it states that if three points A, B, and C are collinear and B is between A and C, then the distance between A and C ($\overline{AC}$) is equal to the sum of the distances between A and B ($\overline{AB}$) and between B and C ($\overline{BC}$) [18]. This Postulate is the basis on which the set of equations is defined.

$$\overline{AC} = \overline{AB} + \overline{BC} \tag{3.6}$$

Let us say there are three devices located in a line like figure 3.2.1 shows, then using equation (3.6) the distances between the devices are related through

$$r_{13} = r_{12} + r_{23}, \tag{3.7}$$

where r_{13},r_{12}, and r_{23} are the true distances of the setup, see Figure 3.2.1. In our measurement scenario, the true values in (3.7) are unknown. Let the distances obtained with the BLE kits be denoted as m_{12}, m_{23}, m_{13}, and the bias b. Then the following equations are found:

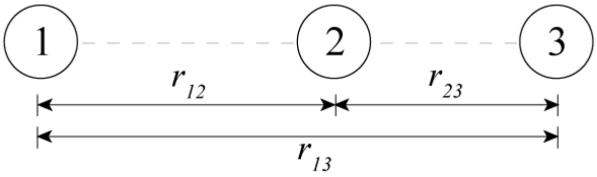

Figure 3.2.1: Distances between three devices

$$m_{12} = r_{12} + b \tag{3.8}$$
$$m_{13} = r_{13} + b \tag{3.9}$$
$$m_{23} = r_{23} + b \tag{3.10}$$

Replacing (3.8), (3.9) and (3.10) in (3.7) and solving for b the following expresion is found:

$$b = m_{12} + m_{23} - m_{13} \tag{3.11}$$

The equation system can also be expressed in matrix form as:

$$\begin{bmatrix} m_{12} \\ m_{23} \\ m_{13} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} r_{12} \\ r_{23} \\ b \end{bmatrix} \tag{3.12}$$

3.2.2 Implications for Two and Three Dimensions

As outlined in the preceding section, the estimation of the bias in a one-dimensional scenario is relatively straightforward. While it is improbable to encounter such ideal conditions in practical real-world scenarios, this approach serves as a proof of concept. However, in scenarios involving two or three-dimensional geometries, the increased number of unknowns necessitates a correspondingly greater number of minimum nodes within the network. Consequently, the theorems and postulates employed for solving these complex systems differ, and the necessity of angle values between nodes becomes crucial for resolving all the unknowns.

3.3 Least Squares Estimation

Least Squares belongs to a class of estimators where no probabilistic assumptions are made about the data, only a signal model is assumed. It is widely used due to its ease of implementation and it has a broad range of applications [19]. In this thesis, the purpose of least squares computation is to find a set of numbers by minimizing the squared difference between the measured data and the assumed noiseless true distances [20]. LSE is regarded as a favorable technique because it selects the best value among multiple measured values. It effectively combines these measurements to provide the best estimates and for some problems, it can deliver an optimal result.

3.3.1 Linear Least Squares

To apply the Linear Least Squares approach for a scalar parameter we assume

$$\mathbf{m} = \mathbf{H}\boldsymbol{\theta} \tag{3.13}$$

where $\mathbf{m}$ is a vector containing the observations of multiple measurements, and $\mathbf{H}$ is a known $N \times p$ matrix of full rank p. The matrix $\mathbf{H}$ is the observation matrix, and θ is a vector containing the unknown values [19]. The Least Squares Estimate is found solving the following problem:

$$\begin{aligned} &\text{Minimize:} \\ &J(\boldsymbol{\theta}) = ||\mathbf{m} - \mathbf{H}\boldsymbol{\theta}||^2 \end{aligned} \tag{3.14}$$

The function $J(\boldsymbol{\theta})$ can also be expressed as

$$J(\boldsymbol{\theta}) = (\mathbf{m} - \mathbf{H}\boldsymbol{\theta})^T(\mathbf{m} - \mathbf{H}\boldsymbol{\theta}). \tag{3.15}$$

Setting the gradient of $J(\boldsymbol{\theta})$ equal to zero yields the LSE estimator

$$\begin{aligned} \frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= -2\mathbf{H}^T\mathbf{x} + 2\mathbf{H}^T\mathbf{H}\boldsymbol{\theta} \\ \hat{\boldsymbol{\theta}} &= \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{x}. \end{aligned} \tag{3.16}$$

3.3.2 Proposed Objective Function

It can be noticed that (3.12) looks in the form of (3.13). So we can say that:

$$\begin{aligned} &\mathbf{m} = \mathbf{H}\boldsymbol{\theta} \\ &\begin{bmatrix} m_{12} \\ m_{23} \\ m_{13} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} r_{12} \\ r_{23} \\ b \end{bmatrix} \end{aligned} \tag{3.17}$$

With these values, it is true that $\mathbf{H}$ is full rank and Least Squares can be applied to estimate r_{12}, r_{23}, and b.

3.3.3 Constraint Least Squares

Another important aspect to consider is that, since our values represent distances that are inherently non-negative, it is necessary to impose a constraint during the estimation process. This constraint, known as Non-Negative Linear Squares (NNLS), ensures that the estimated values remain non-negative, aligning with the physical nature of distances. The goal of NNLS is to find the solution subject to the constraint that all of the variables in the solution are non-negative. By this the problem in 3.14 is extended as follows:

$$\begin{aligned}&\text{Minimize:}\\ &J(\boldsymbol{\theta}) = ||\mathbf{m} - \mathbf{H}\boldsymbol{\theta}||^2\\ &Subject \quad to:\\ &\boldsymbol{\theta} \geq 0.\end{aligned} \tag{3.18}$$

This can be solved using the Python function *scipy.optimize.nnls*(). The algorithm is an active set method, and it solves the KKT (Karush-Kuhn-Tucker) conditions for the non-negative least squares problem and uses a FORTRAN implementation given by [20]. The complexity of the implementation is beyond the scope of this work, and therefore, the existing libraries and functions will be used to estimate the values.

3.3.4 Extension of Objective Function for Repeated Measurements

For a better result, n measurements of the same distance were taken. As our sample size increases, the confidence in the estimate will increase and greater precision is achieved. In order to perform an estimation including all n observations of the values and keep the estimation vector the same size (1×3) an adjustment in the $\mathbf{H}$ should be made. First, the observations can be expressed as:

$$\begin{aligned}\mathbf{m_{12}} &= [m_{12,1}, m_{12,2}, m_{12,3}, \dots m_{12,n}]\\ \mathbf{m_{23}} &= [m_{23,1}, m_{23,2}, m_{23,3}, \dots m_{23,n}]\\ \mathbf{m_{13}} &= [m_{13,1}, m_{13,2}, m_{13,3}, \dots m_{13,n}].\end{aligned} \tag{3.19}$$

Then, the rows of the matrix $\mathbf{H}$ are extended accordingly by n times, and the objective function looks like

$$\begin{bmatrix} m_{12,1}\\ m_{23,1}\\ m_{13,1}\\ m_{12,2}\\ m_{23,2}\\ m_{13,2}\\ \vdots\\ m_{12,n}\\ m_{23,n}\\ m_{13,n}\end{bmatrix} = \begin{bmatrix} 1 & 0 & 1\\ 0 & 1 & 1\\ 1 & 1 & 1\\ 1 & 0 & 1\\ 0 & 1 & 1\\ 1 & 1 & 1\\ & \vdots & \\ 1 & 0 & 1\\ 0 & 1 & 1\\ 1 & 1 & 1\end{bmatrix} \begin{bmatrix} r_{12}\\ r_{23}\\ b,\end{bmatrix} \tag{3.20}$$

and all the previous requirements and constraints are unchanged.

3.3.5 Bias Correction

After the LSE estimation, the θ vector will contain as the third element the value of estimated bias b. Then, the bias-corrected distances are denoted by

$$\hat{\mathbf{r}}_{\mathbf{c}} = \mathbf{m} - b. \tag{3.21}$$

CHAPTER

FOUR

MEASUREMENT SETUP

In order to test the proposed method some sets of data were collected. This was done in an anechoic chamber with the nRF52833 Development Kits. The devices were programmed to perform a ranging and to send the values via UART communication to a laptop computer.

4.1 Hardware Configuration

In order to conform to the Euclidean geometry postulates from the previous sections, a measuring network has to be conformed. Such a network should be able to collect the distances from multiple devices at once. There exists the Nordic Distance Measurement Library and the GitHub repository from [21] that allows obtaining the distances between two devices. To perform network measurements the nRF52833 devices were configured to connect to neighboring devices and perform the MCPD measurements. Using the nRF Connect SDK on version 2.1.0 and the example Script from the Zephyr Project "nrf-dm" as a starting point, the network configuration was achieved. The software works in the following way: There are two parts that run concurrently, one is the main.c code that handles all the Bluetooth advertisement and scanning, device connections and disconnections, and manages the synchronization process for accurate distance measurement. The second is the file peer.c which manages the selection of different devices or peers and is responsible for printing the distance measurements for each device.

The peer.c file was modified in order in order to receive distance measurements in an orderly manner, adding some lines in the *peer_find_closest* function. And to change the output format in the *print_result* function. The full C code can be found in Appendix B1.

A flow diagram explaining the logic of the code is presented in figure 4.1.1. First, a thread is initialized with a starting point in the *peer_thread* function. This is the main function and its job is to continuously wait for measurement results in a message queue. When a result is received, it updates the corresponding peer's information, triggers LED and Bluetooth Low Energy notifications, and prints the result as a CSV file format. The thread also manages a timeout mechanism for peer devices, removing them from the list if they haven't been updated

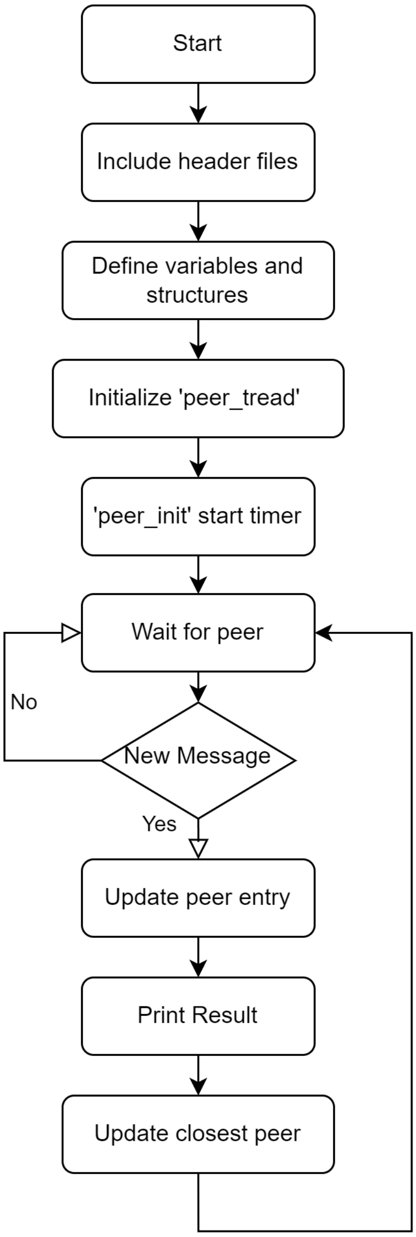

Figure 4.1.1: Configuration program flow diagram

within a certain time period. This thread ensures real-time processing of distance measurements and updates for multiple peers. The *peer_init* function starts a timer for the timeout mechanism and initializes the PWM LED module. The PWM LED module controls an LED that indicates the distance between devices and when the peers are connected. Other functions in the code handle tasks such as finding the closest peer, managing the list of tracked peers, adding new peers, setting the ranging mode (in this case MCPD), and preparing and retrieving the address of the device (used to identify the nodes). These functions provide all the necessary functionality required for peer management and distance measurement in the code.

All devices are flashed with this code and can act as either initiators or reflectors. The role is defined in the *scan_start* function. Since we obtain a distance report in both devices the role that they take is not an issue. This allows for multiple geometrical configurations, a limitation of the program is that it does not relay messages and only allows for a star network topology.

Another program was also used to capture the unprocessed I/Q measurements. In this case, the devices were configured using the repository from GitHub available in [21]. Both programs use a UART connection to a computer to save the data.

4.2 Data Collection

The data collection was made in two scenarios involving different number of devices inside the anechoic chamber. A total of 2000 samples of each distance were taken, this will bring a high level of confidence in the estimated bias. The first scenario involved three devices and the second scenario involved a network of five devices.

4.2.1 Three-device Network

This scenario follows the configuration of Figure 4.2.1. The setup in the anechoic chamber can be seen in figure 4.2.2. Devices 2 and 3 were positioned in a straight line and the distances and nomenclature are presented in table 4.2.1. The captured data is presented in Appendix C2.

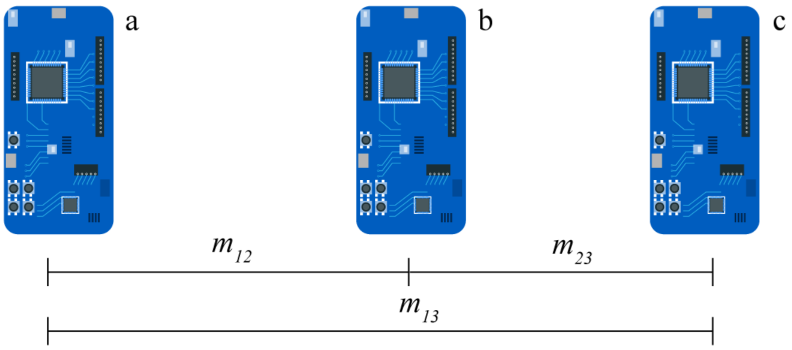

Figure 4.2.1: Setup for data collection

Segment	True Distance [m]
r_{12}	2.07
r_{23}	1.52
r_{13}	3.59

Table 4.2.1: Table of the distances for the network measurements with three devices

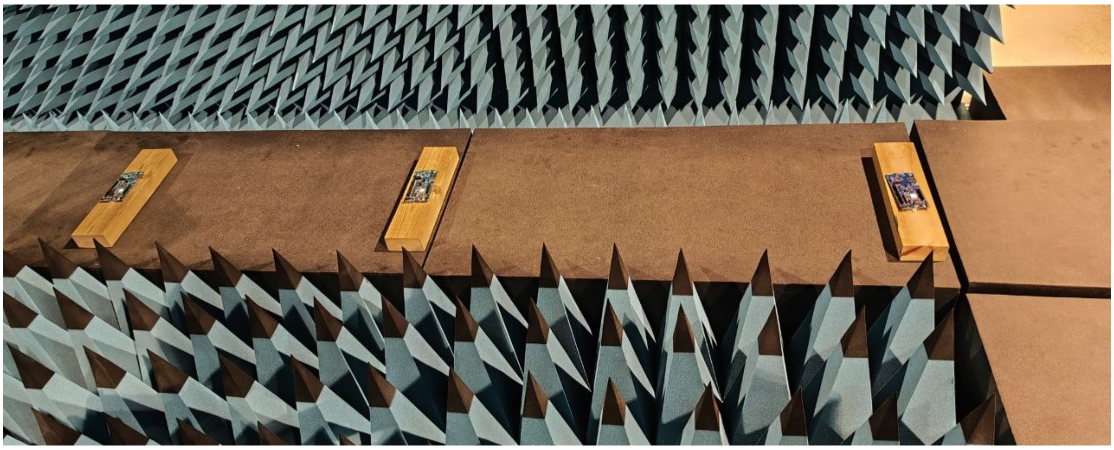

Figure 4.2.2: Measurement setup in the anechoic chamber

4.2.2 Five-device Network

To examine the impact of increased measurements and device count on the values of the least squares estimation, a network consisting of five devices was constructed. By expanding the scope of the network, we aim to gain deeper insights into the behavior and performance of the least squares estimation method in various scenarios. For five devices a total of 10 different distance measurements can be taken. The distances and nomenclature for this setup can be seen in Figure 4.2.3 and Table 4.2.2. The captured data is available in Appendix C3.

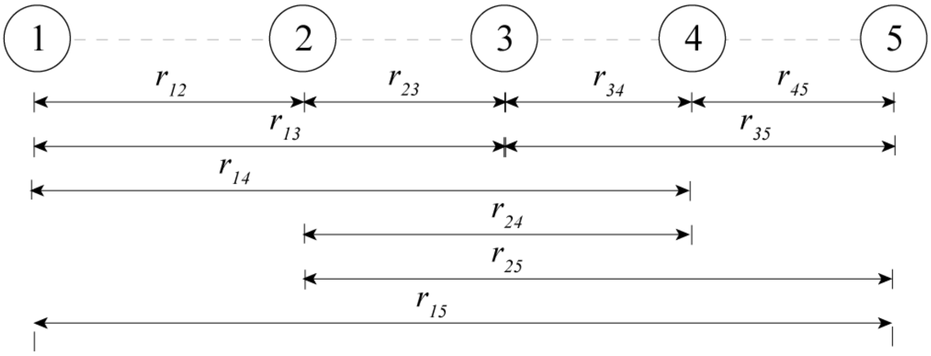

Figure 4.2.3: Second setup for data collection

The real setup can be seen in figure 4.2.4, this set of measurements was also

Segment	True Distance [m]
r_{12}	1.23
r_{13}	2.13
r_{14}	2.76
r_{15}	3.61
r_{23}	0.9
r_{24}	1.53
r_{25}	2.38
r_{34}	0.63
r_{35}	1.48
r_{45}	0.85

Table 4.2.2: Table of the distances for the network measurements

taken in the anechoic chamber. To include all ten measurements on the distance estimation the objective function (3.17) has to be modified accordingly with another set of equations.

The objective function for the second scenario is

$$\mathbf{m} = \mathbf{H}\boldsymbol{\theta}$$

$$\begin{bmatrix} m_{12} \\ m_{13} \\ m_{14} \\ m_{15} \\ m_{23} \\ m_{34} \\ m_{23} \\ m_{25} \\ m_{34} \\ m_{35} \\ m_{45} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} r_{12} \\ r_{13} \\ r_{14} \\ r_{23} \\ r_{24} \\ r_{25} \\ r_{34} \\ r_{35} \\ r_{45} \\ b \end{bmatrix} \tag{4.1}$$

and the rank of **H** is 10 with determinant -3 so it is full rank and LSE can be applied.

4.3 Data Processing

The data processing was done utilizing Python, leveraging the capabilities of essential libraries such as SciPy, NumPy, and Pandas. The structure and logic of the scripts are described in the present section and the algorithm can be seen in

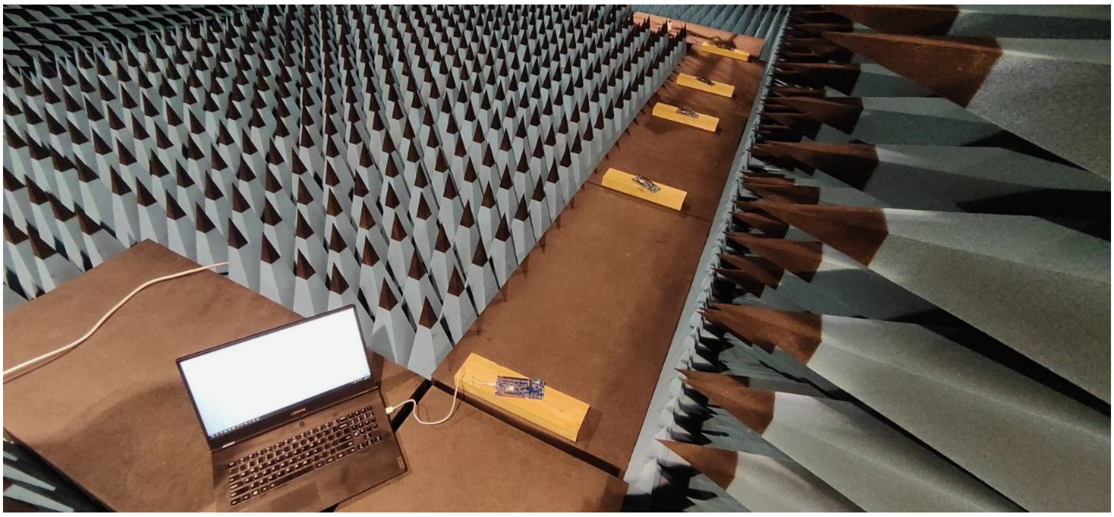

Figure 4.2.4: Measure with five devices

Appendix A3. First, a data frame is created from the CSV file. Each file is a set of measurements taken at each distance, then a filtering of the data is made based on the quality value of the measurement, and an id is assigned to each node based on its address to easily further processing. Then each distance observation is filtered based on a parameter called 'quality' that assures the measurement was performed correctly. Finally, all measurements are stored in individual arrays of 2000 elements each. Then **H** and **m** are created, these are the elements of the objective function we are going to use to apply the Least Squares estimation. With all the parameters the different calculations and estimations are done and the results will be presented in the next chapter.

4.4 Performance Metrics

In order to evaluate the effect of the bias correction in the distance measurements the Mean Absolute Error (MAE) can be applied. MAE is defined as the average variance between paired observations of a dataset [22], in the present case it will be a comparison between the measured distance and the bias-corrected distance. Let the position estimation error for each measurement n is given by the Euclidean distance between the estimated position m_n and the true position r_n, then the MAE is calculated by:

$$\text{MAE} = \frac{\sum_{i=1}^{n} |r_i - m_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n} \tag{4.2}$$

An advantage of using this metric is that the mean absolute error uses the same scale as the data being measured, which in this case is meters. Another metric is the Mean Bias Error (MBE) which captures the average bias in the estimation [22] and is calculated as:

$$\text{MBE} = \frac{1}{n} \sum_{i=1}^{n} (m_i - r_i) \tag{4.3}$$

These performance metrics were calculated using the Python Script in Appendix A3 and the results will be presented in the next chapter.

CHAPTER

FIVE

NUMERICAL RESULTS

The results of the measurements and bias estimation are shown in the present section. First, the results of the three-device network and later the estimations for the Five-device network. The estimations were evaluated using the Mean Absolute Error before and after the bias correction.

5.1 Three-device Network

The collected data is presented in Figure 5.1.1, and the unprocessed measurements can be found in Appendix C2. The values of m_{12}, m_{23}, m_{13} are the measured distances, r_{12}, r_{23}, r_{13} are the real distances, and b is the calculated bias using (3.11) for every set of three samples.

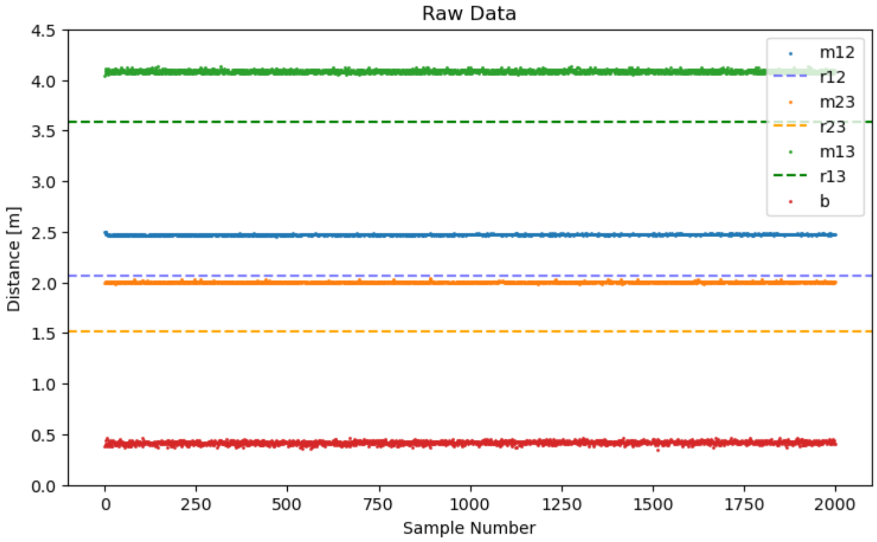

Figure 5.1.1: Captured data and bias in three-device network

A histogram of the bias can be seen in Figure 5.1.2. This graph provides an initial glimpse into the anticipated outcomes of the LSE. Its purpose is to offer an understanding of what can be expected from the estimation process.

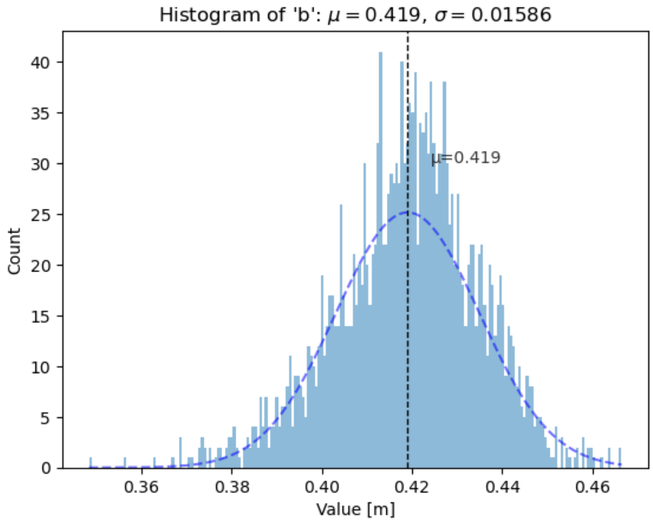

Figure 5.1.2: Histogram of the values of b

The MBE for each measure is presented in table 5.1.1. The values of the MBE represent the bias in the initial measurement.

Measurement	True Distance [m]	MBE [m]
m_{12}	2.07	0.403
m_{23}	1.52	0.482
m_{13}	3.59	0.494

Table 5.1.1: Table of the Mean Bias Error on the measured data

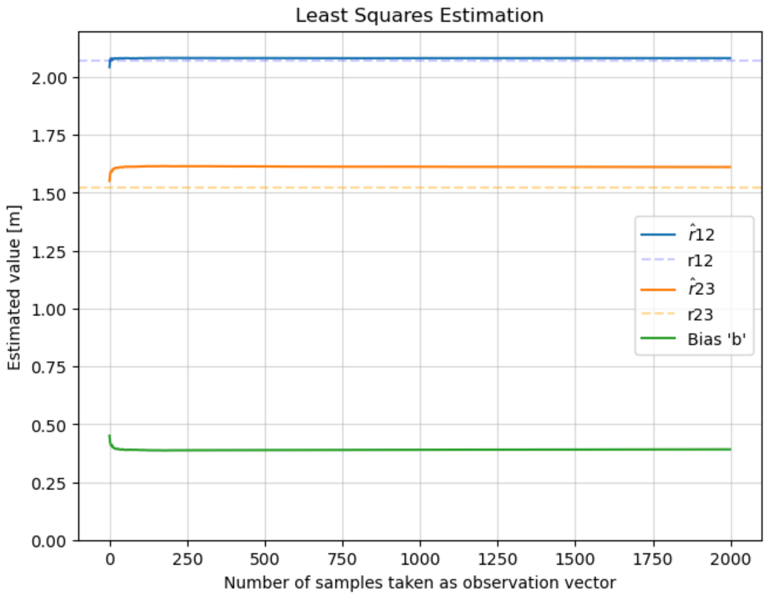

Figure 5.1.3: LSE estimated values in function of the number of samples

Subsequently, a least squares estimation was performed, and the impact of sample size on the estimation of b is presented in Figure 5.1.3. It is evident that as the sample size increases, the estimated value gradually converges to a specific value, illustrating the tendency toward consistency in the estimation process. A correction value of 0.391 m is derived from the analysis of 2000 samples. This value is used to correct the values of the original measured data. The MBE error for the measures in function of the number of samples can be seen in Figure 5.1.4. The MAE is calculated for both the distance values before and after the bias correction and the results are presented in Table 5.1.2. A reduction of the absolute error is present in all the cases.

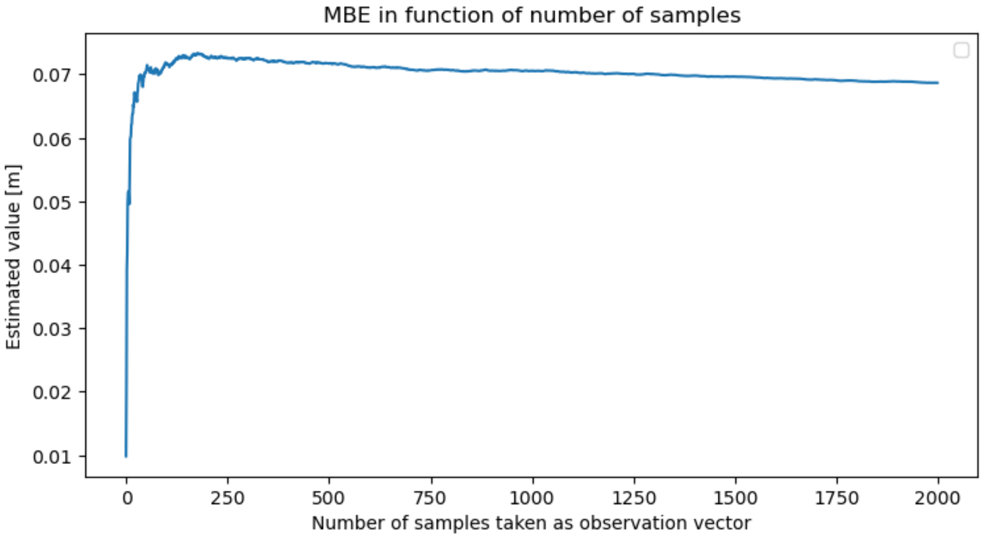

Figure 5.1.4: MBE in function of the number of samples

Measurement	True Distance [m]	MAE of raw data [m]	MAE with bias correction [m]
m_{12}	2.07	0.403	0.012
m_{23}	1.52	0.482	0.091
m_{13}	3.59	0.494	0.103

Table 5.1.2: Table of the Mean Average Error for each distance before and after the bias correction

The effect on the measured values can be seen in Figure 5.1.5. This Figure presents histograms of all the captured data. The bias correction has the effect of moving all measurements to the left. The dashed lines represent the true distances so the closer the histograms are to them the more accurate the measure.

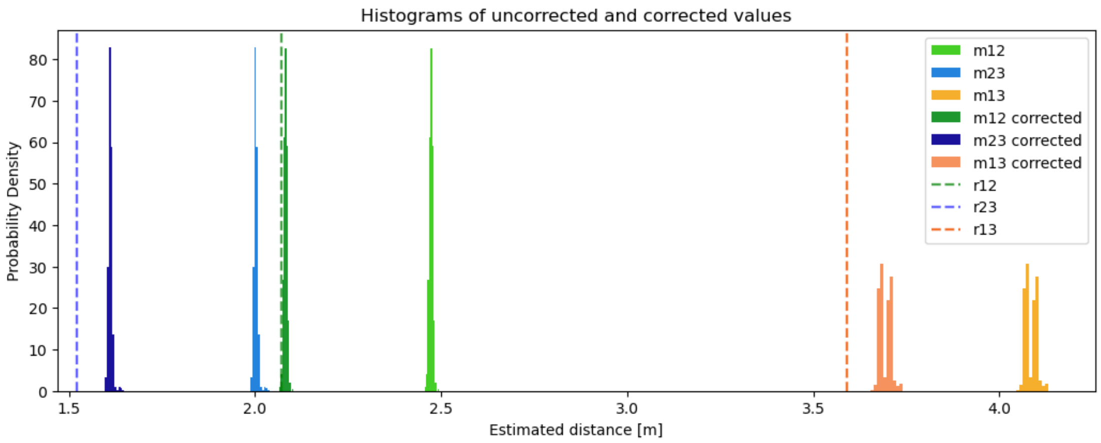

Figure 5.1.5: Measured values before and after the bias correction on three-device network

Finally, a single MBE concerning all the measures was calculated. Before the correction, the overall MBE value was 0.460m, indicating the average bias of the measurements. After the correction, the overall MBE value decreased to 0.069m, reflecting a reduction of the bias in 39cm. A lower MBE value signifies an improvement in accuracy, and a positive value suggests that the estimated distances tend to be larger than the true values on average.

5.2 Five-device Network

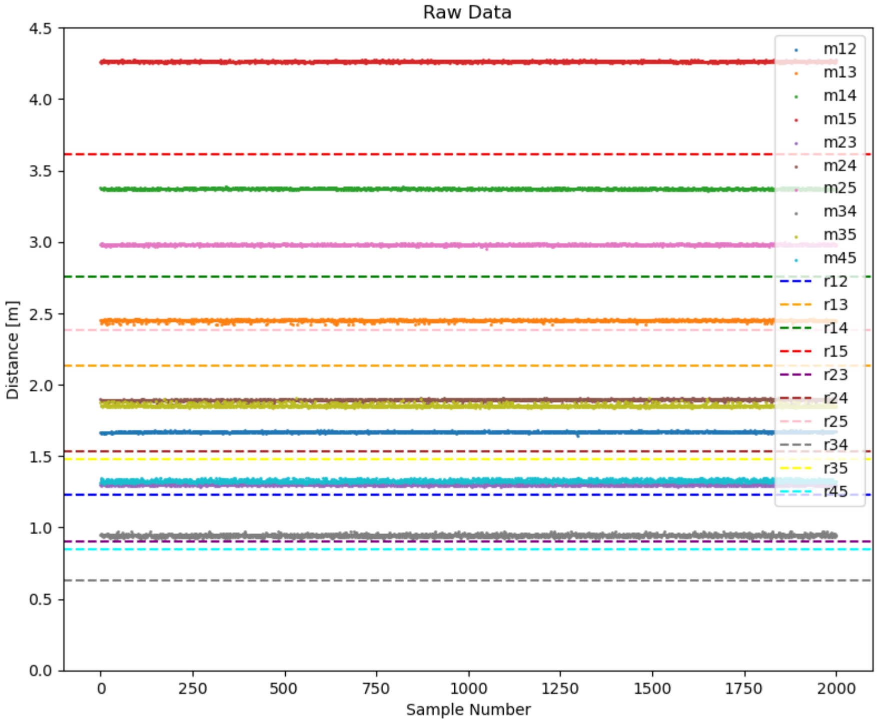

Figure 5.2.1: Captured data of five device network

The results from the second experiment are shown in this section. All the measured distances are presented in Figure 5.2.1. In the same way as the previous example, a total of 2000 samples of each distance were taken.

Measurement	True Distance [m]	MBE [m]
m_{12}	1.23	0.439
m_{13}	2.13	0.321
m_{14}	2.76	0.612
m_{15}	3.61	0.653
m_{23}	0.9	0.398
m_{24}	1.53	0.364
m_{25}	2.38	0.600
m_{34}	0.63	0.315
m_{35}	1.48	0.373
m_{45}	0.85	0.476

Table 5.2.1: Table of the Mean Bias Error of the network measurements

The true distance and the bias error of each measurement are shown in Table 5.2.1.

The least squares estimation of b can be seen in Figure 5.2.2. A correction value of 0.326m is estimated.

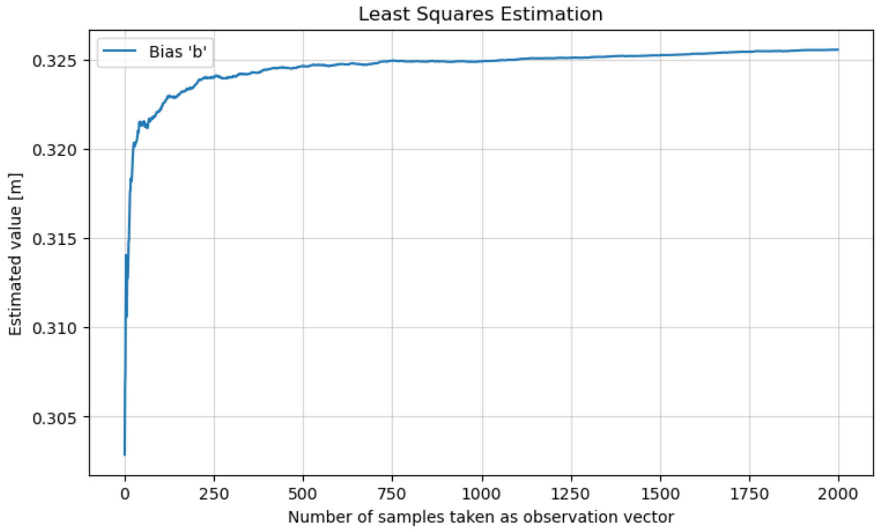

Figure 5.2.2: LSE estimated value in function of the number of samples

In a similar manner, the MAE is calculated for each measurement. The results before and after the bias correction are presented in Table 5.2.2. An error reduction is evident in all the measurements and the correction value is congruent with the previous findings.

Measurement	**True Distance [m]**	**MAE of raw data [m]**	**MAE with bias correction [m]**
m_{12}	1.23	0.439	0.114
m_{13}	2.13	0.321	0.006
m_{14}	2.76	0.612	0.286
m_{15}	3.61	0.653	0.328
m_{23}	0.90	0.398	0.073
m_{24}	1.53	0.364	0.038
m_{25}	2.38	0.600	0.275
m_{34}	0.63	0.315	0.011
m_{35}	1.48	0.373	0.048
m_{45}	0.85	0.477	0.151

Table 5.2.2: Table of the Mean Average Error on the data before and after the bias correction

The histograms of the corrected values are shown in Figure 5.2.3. There are two cases where the median of the histogram is at the left side of the true distance, this means that for these cases the estimated values tend to be smaller than the true distance after the correction.

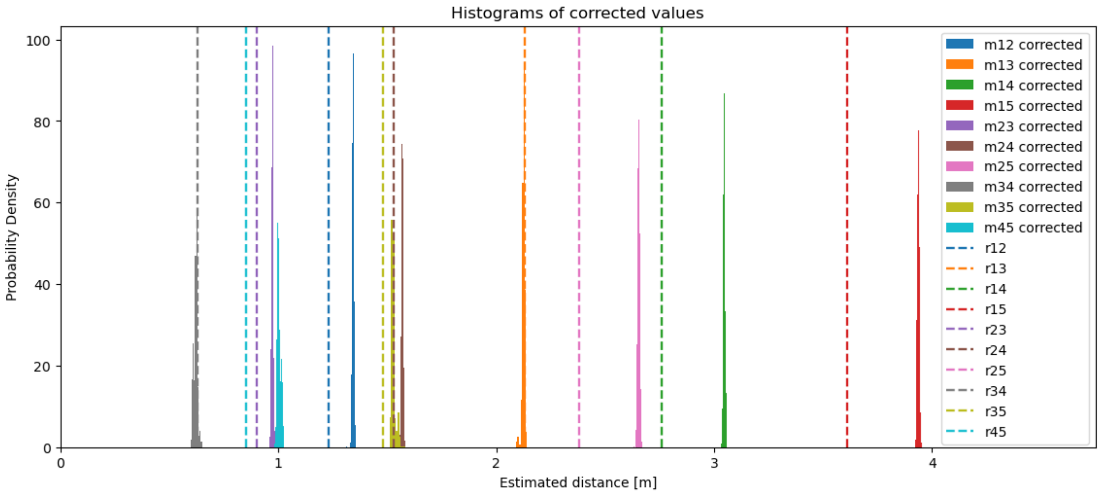

Figure 5.2.3: Histogram of corrected values and relation to their true distances for the five-device network

Measurement	**True Distance [m]**	**MBE [m]**
m_{12}	1.23	0.114
m_{13}	2.13	-0.005
m_{14}	2.76	0.286
m_{15}	3.61	0.328
m_{23}	0.9	0.073
m_{24}	1.53	0.038
m_{25}	2.38	0.275
m_{34}	0.63	-0.01
m_{35}	1.48	0.047
m_{45}	0.85	0.151

Table 5.2.3: Table of the Mean Bias Error of the corrected measurements for the second scenario

The exact values of the mean bias error for the corrected distances are visible in Table 5.2.3. The overall bias error is also calculated with these results given a value of 0.455m before and a value of 0.130m after the correction, given a reduction of the bias of 32cm on average.

5.3 Mean Bias Error

The MBE of the uncorrected measures is presented in this section, this includes both scenarios. The bias error in respect of the true distance is presented in Figure 5.3.1. In this graph, some interesting behavior of the bias can be observed. It appears that the bias value tends to be larger with the distance. This contradicts some of the assumptions made in Chapter 3 and its implications will be discussed in the next Chapter.

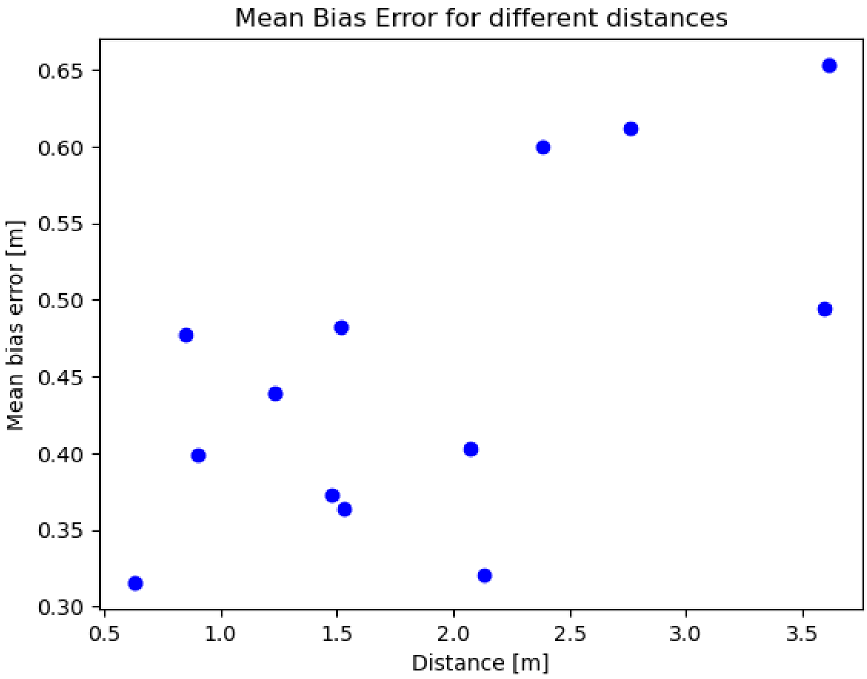

Figure 5.3.1: Scatter graph of the MBE values

CHAPTER

SIX

DISCUSSION

The data presented in the previous section indicates that the estimation process was able to return a value that reduces the mean error in all measurements, providing a more accurate distance on the ranging process. This method is performed after obtaining the distance estimates, in this thesis these values were estimated by using the MCPD-LR algorithm. Since the estimation is calculated after the estimation it can be applied with any other of the aforementioned algorithms.

In the case of the first scenario with three devices, the accuracy is improved largely due to the fact that the bias values were very similar in all three distances. In the case of the second scenario where more devices (and measurements) were involved there was an improvement in the accuracy but it was not as high as in the previous case, this is the result of the mean bias error being less correlated. In some cases, it can be seen that the bias value could be double in magnitude than the rest, this behavior can attract some problems if the estimation is performed with only three devices. However, the method estimates a value that in the end returned an improvement in accuracy. In general terms, any sensitive positive value for the bias will improve the distance measurements but as was observed an overestimation will cause the appearance of negative MBE values. This effect can be problematic at close distances because it can make the corrected estimations appear as negative.

The effects of the number of samples in the LSE estimation for each scenario can be seen in Figures 5.1.4 and 5.2.2. From this behavior can be determined that selecting a value on a few samples is not recommended and can cause higher error metrics and the presence of outliers has more impact. But the opposite is also true, it is not necessary a very high number of observations to obtain an acceptable value. In the case of the three-device network there is a difference of 3mm between the value estimated with 500 samples and the value with 2000 samples. In the case of the five-device network, the values converge even faster due to the higher information available and a value obtained with 200 samples differs by 2mm of the final estimated value. It is important to note that these graphs tend to be noisy with few observations, but a value obtained with reasonable samples will achieve results with acceptable levels of accuracy.

The multiple measurements at different distances gave room for additional findings. There appears to exist certain variability on the mean bias error that depends on the distance, this new information is contrary to the suppositions made for the bias model that it does not change. Some rapid conclusion can be made, and it seems that the bias tends to have an increase proportional to the distance, however, a deeper analysis is necessary perhaps at multiple distances. Because of this behavior, the proposed geometric considerations will not work in some cases. For example, if we think on the three-device scenario, if the bias value on m_{13} is double the value of m_{12} and m_{23} then our LSE estimator will return a value of zero or very close to zero since for this case the euclidean postulate holds but we are still observing erroneous distances.

CHAPTER

SEVEN

CONCLUSION

This thesis has achieved important objectives in addressing the bias observed in the distance measurements obtained with MCPD algorithms. Firstly, a comprehensive bias model was developed, allowing for a better understanding of the underlying problem. The model was constructed by incorporating a set of assumptions. These assumptions were carefully chosen to strike a balance between simplifying the model and ensuring its realism.

Secondly, a network setup was successfully implemented to collect the necessary data for estimating the bias value. The application of the Least Squares approach through regression analysis yielded valuable insights and approximations of the bias value.

Furthermore, the incorporation of the estimated bias into the measurements showed good results in terms of improving accuracy. The Mean Average Error and Mean Bias Error values obtained from the dataset demonstrated the effectiveness of this approach in reducing bias-induced error. These findings highlight the importance of addressing bias in MCPD algorithms and emphasize the potential for enhancing the reliability and precision of distance measurements.

Overall, this study provides a better insight into the bias cause and its impact on distance estimation. It opens avenues for further research and development in improving the performance of MCPD algorithms, ultimately leading to more accurate and reliable distance measurements in various applications.

7.1 Future work

A possible solution for the bias value changing for different distances could be to modify the model presented and adapt the equation system accordingly. More data is necessary to find out the true behavior of the bias. A testing scenario on greater distances could be interesting. In the present work, a limiting factor was the size of the anechoic chamber and more measurements could have been taken in a bigger room, however, it is important to note that the distance estimation is prone to more errors and inaccuracies in scenarios with multipath and fading.

Some improvements could be done in the estimation algorithms to be less affected by these effects. One solution could be to introduce other parameters in the calculation of the distances, for example, the use of AoA information that the BLE radios are already capable of retrieving. This will allow for more robust and accurate estimation algorithms.

There are some ideas that could be tried during the Geometric considerations of Chapter 3. This thesis has been mainly focused on a set of equations that are congruent in one dimension but it is very rare to have such a scenario in real applications. The following ideas could be tested:

- A network configuration for two and three dimensions. It is possible that with a bias estimation in three dimensions, any network could use the estimation method to make a correction process once deployed. This could be used as an initial calibration process and will allow for more accurate applications for measuring and also tracking and navigation.
- With an increasing number of devices the data collection will be more challenging. In the present thesis, measurements were taken on each device one by one, this is not feasible in a greater network nor it is scalable. A configuration of the devices in order to achieve the routing of messages will allow for the creation of a mesh network solving the scalability problem and allowing for the possibility to receive all the distances in a central device that perhaps has more computational power to perform the estimations.

REFERENCES

[1] *GPS.gov: GPS Accuracy.* URL: https://www.gps.gov/systems/gps/performance/accuracy/.

[2] Germán Martín Mendoza-Silva, Joaquín Torres-Sospedra, and Joaquín Huerta. "A Meta-Review of Indoor Positioning Systems". In: *Sensors* 19.20 (Oct. 2019), p. 4507. ISSN: 1424-8220. DOI: 10.3390/s19204507.

[3] Gabriele S. de Blasio et al. "Bluetooth Low Energy Technology Applied to Indoor Positioning Systems: An Overview". In: 2020, pp. 83–90. DOI: 10.1007/978-3-030-45093-9{_}11.

[4] Faheem Zafari, Athanasios Gkelias, and Kin K. Leung. "A Survey of Indoor Localization Systems and Technologies". In: *IEEE Communications Surveys and Tutorials* 21.3 (2019), pp. 2568–2599. ISSN: 1553877X. DOI: 10.1109/COMST.2019.2911558.

[5] Pouria Zand et al. "A high-accuracy phase-based ranging solution with Bluetooth Low Energy (BLE)". In: *IEEE Wireless Communications and Networking Conference, WCNC.* Vol. 2019-April. Institute of Electrical and Electronics Engineers Inc., Apr. 2019. ISBN: 9781538676462. DOI: 10.1109/WCNC.2019.8885791.

[6] Pepijn Boer et al. "Performance of High-Accuracy Phase-Based Ranging in Multipath Environments". In: *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring).* IEEE, May 2020, pp. 1–5. ISBN: 978-1-7281-5207-3. DOI: 10.1109/VTC2020-Spring48590.2020.9128721.

[7] Mathias Pelka, Christian Bollmeyer, and Horst Hellbruck. "Accurate radio distance estimation by phase measurements with multiple frequencies". In: *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN).* IEEE, Oct. 2014, pp. 142–151. ISBN: 978-1-4673-8054-6. DOI: 10.1109/IPIN.2014.7275478.

[8] Tan Kim Geok et al. "Review of Indoor Positioning: Radio Wave Technology". In: *Applied Sciences 2021, Vol. 11, Page 279* 11.1 (Dec. 2020), p. 279. ISSN: 2076-3417. DOI: 10.3390/APP11010279. URL: https://www.mdpi.com/2076-3417/11/1/279/htm%20https://www.mdpi.com/2076-3417/11/1/279.

[9] Zan Li et al. "TDOA for narrow-band signal with low sampling rate and imperfect synchronization". In: *2014 7th IFIP Wireless and Mobile Networking Conference (WMNC).* 2014, pp. 1–8. DOI: 10.1109/WMNC.2014.6878849.

[10] Davide Dardari, Pau Closas, and Petar M. Djuric. “Indoor tracking: Theory, methods, and technologies”. In: *IEEE Transactions on Vehicular Technology* 64.4 (Apr. 2015), pp. 1263–1278. ISSN: 00189545. DOI: 10.1109/TVT.2015.2403868.

[11] Zheng Yang, Zimu Zhou, and Yunhao Liu. “From RSSI to CSI: Indoor Localization via Channel Response”. In: *ACM Comput. Surv.* 46.2 (Dec. 2013). ISSN: 0360-0300. DOI: 10.1145/2543581.2543592. URL: https://doi.org/10.1145/2543581.2543592.

[12] Martin Woolley. *Bluetooth Core Specification v5.1 Feature Overview.* Tech. rep. 2020. URL: https://www.bluetooth.com/bluetooth-resources/bluetooth-core-specification-v5-1-feature-overview/.

[13] *Bluetooth Technology Overview | Bluetooth® Technology Website.* URL: https://www.bluetooth.com/learn-about-bluetooth/tech-overview/.

[14] Yannic Schröder, Dennis Reimers, and Lars Wolf. “Accurate and Precise Distance Estimation from Phase-Based Ranging Data”. In: *IPIN 2018 - 9th International Conference on Indoor Positioning and Indoor Navigation* (Nov. 2018). DOI: 10.1109/IPIN.2018.8533871.

[15] Dana Maklada et al. “High Accuracy Distance Measurement Using Frequency Comb”. In: *2021 IEEE International Conference on Microwaves, Antennas, Communications and Electronic Systems, COMCAS 2021.* Institute of Electrical and Electronics Engineers Inc., 2021, pp. 361–366. ISBN: 9780738146720. DOI: 10.1109/COMCAS52219.2021.9629073.

[16] *nRF52833 - Advanced Bluetooth multiprotocol SoC - nordicsemi.com.* URL: https://www.nordicsemi.com/Products/nRF52833.

[17] *Nordic Distance Measurement library — nrfxlib 2.2.99 documentation.* URL: https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrfxlib/nrf_dm/README.html.

[18] Walter Prenowitz and Meyer Jordan. *Basic concepts of geometry.* Rowman & Littlefield, 2012.

[19] Steven M Kay. *Fundamentals of Statistical Signal Processing.* Prentice-Hall, 1993.

[20] Charles L. Lawson and Richard J. Hanson. *Solving Least Squares Problems (Classics in Applied Mathematics, Series Number 15).* 1987.

[21] *GitHub - wulffern/nrfdmiq.* URL: https://github.com/wulffern/nrfdmiq.

[22] Ranadip Pal. “Validation methodologies”. In: *Predictive Modeling.* Elsevier, 2017, pp. 83–107. DOI: 10.1016/B978-0-12-805274-7.00004-X.

APPENDICES

A - PYTHON SCRIPTS

A1 - MCPD function

```
c =299792458
fh=1000000
k=79
def calcMCPD(transfer2,):
    ang = np.unwrap(np.angle(transfer2))
    x=np.arange(4,78,1)
    A = np.vstack([x, np.ones(len(x))]).T
    rvec3 = np.empty(0)
    for i in range(5, len(ang)-1):
        rvec= c*-(ang[i]-ang[i-1])/(4*np.pi*fh)
        rvec3 = np.append(rvec3, rvec)
    MCPDvalue=np.mean(rvec3)
    return MCPDvalue
```

A2 - MCPD-LR function

```python
def calcREG(transfer2):
    ang = np.unwrap(np.angle(transfer2))
    x=np.arange(4,78,1)
    A = np.vstack([x, np.ones(len(x))]).T
    m,off = np.linalg.lstsq(A,ang[4:78],rcond=None)[0]
    REGvalue=(c*-m)/(4*np.pi*fh)
    return REGvalue
```

A2 - Data processing script

Only the function definitions are presented in this appendix. The complete code that also generates the graphs is too extensive for this section and can be found in the attached documents.

```python
import numpy as np
import matplotlib.pyplot as plt
import json
import os
import glob
import pandas as pd
from scipy import stats
import pylab as py

c =299792458
fh=1000000
k=79

def readFromFile(filename):
    filename = filename
    with open(filename) as fi:
        obj = json.load(fi)

    quality = obj["quality"]

    l = obj[’i_local’] +  np.multiply(1j,obj[’q_local’])
    r = obj[’i_remote’] +  np.multiply(1j,obj[’q_remote’])
    transfer2 = np.multiply(l,r)

    return transfer2, quality, l, r

def readfiles(directory):
    files = glob.glob(directory + os.path.sep +"*.json")
    mes = list()
    # locals = list()
    # remotes = list()
    for f in files:
        transfer2, quality, l, r= readFromFile(f)
        if quality > 0:
            continue
        mes.append(transfer2)

    estimations=np.empty(0)
    for i in mes:
        estimations = np.append(estimations, calcREG(i))
    return estimations

def calc_nnls3(n,m13,m35, m15):
    m_1=np.array(m13[0:n])
    m_2=np.array(m35[0:n])
    m_3=np.array(m15[0:n])
    y_n = np.array([m_1, m_2, m_3])
    y_n = y_n.ravel(’F’)

    pd_H = np.matrix(’1 0 1; 0 1 1; 1 1 1’)
    pd_H2 = np.matrix(’1 0 1; 0 1 1; 1 1 1’)
    for i in range(0,n-1):
        pd_H2= np.vstack([pd_H2, pd_H])
    estimate = nnls(pd_H2, y_n)
    return estimate

def calc_nnls_5(n,m12, m13,m14,m15, m23,m24,m25,m34,m35,m45):
    m_12=np.array(m12[0:n])
    m_13=np.array(m13[0:n])
    m_14=np.array(m14[0:n])
    m_15=np.array(m15[0:n])
    m_23=np.array(m23[0:n])
    m_24=np.array(m24[0:n])
    m_25=np.array(m25[0:n])
    m_34=np.array(m34[0:n])
    m_35=np.array(m35[0:n])
    m_45=np.array(m45[0:n])

    y_n = np.array([m_12, m_13,m_14,m_15, m_23,m_24,m_25,m_34,m_35, m_45])
    y_n = y_n.ravel(’F’)

    pd_H = np.matrix(’1 0 0 0 0 0 0 0 0 1; 0 1 0 0 0 0 0 0 0 1; 0 0 1 0 0 0 0 0 0 1; 1 0 0 1 0 0 1 0 1 1; 0 0 0 1 0 0 0 0 0 1; 0 0 0 0 1 0 0 0 0 1; 0 0 0 0 0 1 0 0 0 1; 0 0 0 0 0 0 1 0 0 1; 0 0 0 0 0 0 0 1 0 1; 0 0 0 0 0 0 0 0 1 1’)
    pd_H2 = np.matrix(’1 0 0 0 0 0 0 0 0 1; 0 1 0 0 0 0 0 0 0 1; 0 0 1 0 0 0 0 0 0 1; 1 0 0 1 0 0 1 0 1 1; 0 0 0 1 0 0 0 0 0 1; 0 0 0 0 1 0 0 0 0 1; 0 0 0 0 0 1 0 0 0 1; 0 0 0 0 0 0 1 0 0 1; 0 0 0 0 0 0 0 1 0 1; 0 0 0 0 0 0 0 0 1 1’)
    for i in range(0,n-1):
        pd_H2= np.vstack([pd_H2, pd_H])
    estimate = nnls(pd_H2, y_n)
    return estimate

def mae(y_true, y_pred):
    return np.mean(np.abs(y_true - y_pred))

def mbe(y_true, y_pred):
    return np.mean(y_pred- y_true)
```

B - CONFIGURATION FILES

B1 - nRF52833 C files

This code is extensive and it can be found in the attached files.
main.c
peer.c

C - RAW MEASUREMENTS

C1 - Magnitude response Measurements

The measurements can be found in the attached files.

C2 - Three Device Network Measurements

The measurements can be found in the corresponding folder. Attached Electronically.

C3 - Five Device Network Measurements

The measurements can be found in the corresponding folder. Attached Electronically.

NTNU

Norwegian University of
Science and Technology