TECHNICAL UNIVERSITY OF CRETE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING
TELECOMMUNICATIONS DIVISION

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
TECHNICAL UNIVERSITY OF CRETE

Phase-based Localization of Low Power Bluetooth Tags with Multi-antenna Receivers and Comparison with RFID Technology

by

Georgios Andreadis

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DIPLOMA OF

ELECTRICAL AND COMPUTER ENGINEERING

Chania, July 2024

THESIS COMMITTEE

Professor Thrasyvoulos Spyropoulos, *Thesis Supervisor*
Professor Aggelos Bletsas
Professor Athanasios Liavas

Abstract

Localization technologies are essential for applications ranging from asset tracking to navigation systems. This study explores minimizing Bluetooth Low Energy (BLE) multi-antenna receivers (locators) to reduce energy consumption, hardware usage, and installation costs. It introduces a hyperbolic localization technique based on phase differences, using linear approximations of hyperbolas to calculate 3D direction of arrival (DoA) and estimate BLE tag positions. On a single locator, DoA estimation achieves a Mean Absolute Error (MAE) under 10° for azimuth and under 7° for elevation. However, a single locator cannot determine the tag's position due to system limitations. Therefore, the study explores using multiple locators. On multiple locators, the proposed localization method can enhance the prior art localization method, reducing the MAE of localization error by 10%.

The work also compares Radio Frequency Identification (RFID) and BLE for tag localization accuracy under static conditions. While the hyperbolic localization technique is effective, it is sensitive to multipath noise. BLE's spatial and frequency diversity mitigate multipath issues, achieving a 3D localization MAE of 30 cm for the topology with high coverage and 80 cm for the topology with low coverage. In contrast, RFID, lacking such diversity, results in a 3D localization MAE of 1.2 m for the second topology. This comparison suggests developing new algorithms for consistent results across both technologies and the potential for merging BLE and RFID into a single tag to minimize energy consumption and enhance localization accuracy.

To further minimize BLE locators, neural networks were employed. Recognizing BLE single position measurements as sequential data, Recurrent Neural Networks (RNNs) were utilized, achieving a 2D localization MAE of 30 cm with a single locator, a challenge unsolvable by deterministic methods.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to *Prof. Thrasyvoulos Spyropoulos* and *Prof. Athanasios Liavas* of the committee for their cooperation, given the unique situation created due to *Prof. Aggelos Bletsas*, my original supervisor, being on leave at the final stages of my thesis. As he was not eligible to serve as the official supervisor, a change had to be made at the last moment, and I am grateful that it could all happen in time.

I am particularly grateful to *Prof. Aggelos Bletsas* for his guidance and support throughout most of my research. His insights and mentorship were invaluable.

I would also like to sincerely thank *Evangelos Giannelos* and *Panos Alevizos* for their invaluable assistance in completing my thesis. I am also grateful to *ADVEOS* for providing the necessary equipment and the internship that facilitated the completion of my work and enhanced my learning experience.

Additionally, I want to thank my *lab colleagues* for making the time in the lab more enjoyable and for their help with the experimental setups.

Lastly, this journey would not have been possible without the unwavering support of my *friends* and *family*. Thank you for always believing in me and providing your constant encouragement over the years.

Thank you all for your unwavering support and for making this journey a fulfilling and memorable experience.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless communication technologies have fundamentally transformed our daily interactions with devices and systems. From the widespread use of smartphones to the networked devices in smart homes, the capability to transmit data wirelessly has unlocked numerous applications that boost convenience, efficiency, and functionality. One particularly significant application is indoor localization, which has become increasingly important due to the demand for precise location tracking in environments where GPS signals are inadequate.

Phase-based indoor localization has attracted significant interest within the wireless research community, given its crucial role in a wide array of applications, including asset tracking, navigation, and location-based services. These applications demand systems characterized by high accuracy and low power consumption. Among the plethora of technologies available for phase-based indoor positioning, Bluetooth Low Energy (BLE) and Radio-Frequency Identification (RFID) have emerged as favored options.

Different categories of tags (active/battery-operated, semi-passive/battery-assisted, or passive/batteryless) find many applications across different industries due to their unique capabilities, such as real-time tracking and monitoring. These tags are used in healthcare and medical applications, retail and inventory management, supply chain logistics, and smart packaging. These applications share the common need for indoor localization and require accurate estimation of the tag's location to acquire information about the item's whereabouts or the path it may follow.

1.1 Bluetooth Low Energy (BLE) Protocol

Bluetooth Low Energy (BLE) is a wireless communication protocol designed for short-range communication with an emphasis on low power consumption, introduced as part of the Bluetooth 4.0 specification in 2010 [1]. BLE operates in the 2.4 GHz ISM band and utilizes frequency hopping to minimize interference and enhance communication reliability.

Bluetooth Low Energy is structured as a protocol, functioning as a rulebook that devices must adhere to in order to communicate or exchange information efficiently and reliably. A protocol's general functionality is broken down into smaller, specialized protocols, each responsible for a specific role. These smaller protocols are organized into a layered architecture known as a protocol stack or protocol suite.

The BLE protocol stack is broadly divided into three primary components or subsystems: application, host, and controller. Each of these blocks contains distinct layers that handle various aspects of BLE communication. The main layers in the BLE protocol stack include:

- **The Physical Layer:** This layer handles the radio transmission and reception of raw bitstreams over the air. It defines the modulation and demodulation techniques, frequency channels, and power levels used for communication.

- **The Link Layer:** Responsible for establishing and maintaining the physical link between devices. It manages the device's roles, connection establishment, advertising, scanning, and data packet handling.

- **The Host Controller Interface (HCI):** This layer acts as a standardized interface between the host and controller, enabling communication between the higher layers of the host stack and the lower layers of the controller stack.

- **The Logical Link Control and Adaptation Protocol (L2CAP):** This layer facilitates the multiplexing of data between different higher

layer protocols. It handles the segmentation and reassembly of large data packets and provides quality of service (QoS) features.

- **Attribute Protocol (ATT):** Defines how data is organized and accessed in a BLE device. It provides a framework for defining attributes, which are used to represent data and services.
- **Generic Attribute Profile (GATT):** Built on top of ATT, GATT specifies how data is structured, discovered, and accessed in a BLE device. It defines standard profiles and services, enabling interoperability between devices from different manufacturers.
- **Generic Access Profile (GAP):** Manages the device's advertisement, discovery, connection, and security procedures. GAP defines the roles that devices can adopt (e.g., central, peripheral, observer, broadcaster) and the procedures for establishing connections and exchanging data.

The BLE architecture is based on a master-slave configuration, where the master device (central) can connect to multiple slave devices (peripherals). BLE communication is structured around two main processes: advertising and scanning.

- **Advertising:** In this process, peripheral devices broadcast packets containing information about their capabilities and available services. These packets are sent on predefined advertising channels. The advertising process is crucial for the discovery phase, where devices announce their presence and readiness to connect.
- **Scanning:** Central devices scan for these advertising packets to discover nearby peripherals. Once an advertising packet is detected, the central device can initiate a connection request. Upon acceptance, a connection is established, allowing for data exchange.

After establishing a connection, data transfer between the central and peripheral devices occurs primarily using the GATT protocol. GATT organizes

data into a hierarchical structure composed of services and characteristics, making it easy to discover, access, and manage data efficiently.

1.1.1 Constant Tone Extension (CTE)

The Constant Tone Extension (CTE) is designed to improve direction finding and localization accuracy [2]. CTE is a type of continuous wave that consists solely of a series of symbols, each representing binary 1. Although the CTE is a simple continuous wave, with a variable length of 16 μs to 160 μs, it is divided into specific periods. The first 4 μs of the CTE is called the guard period, followed by an 8 μs reference period. After the reference period, the CTE consists of a sequence of alternating switch slots and sample slots, each either 1 μs or 2 μs long, as specified by the application. The 2-μs slots allow the use of a cheaper RF switch between the antennas that has a longer transition time. The 1-μs slots enable sampling each antenna multiple times, which can help reduce the effect of noise and improve accuracy.

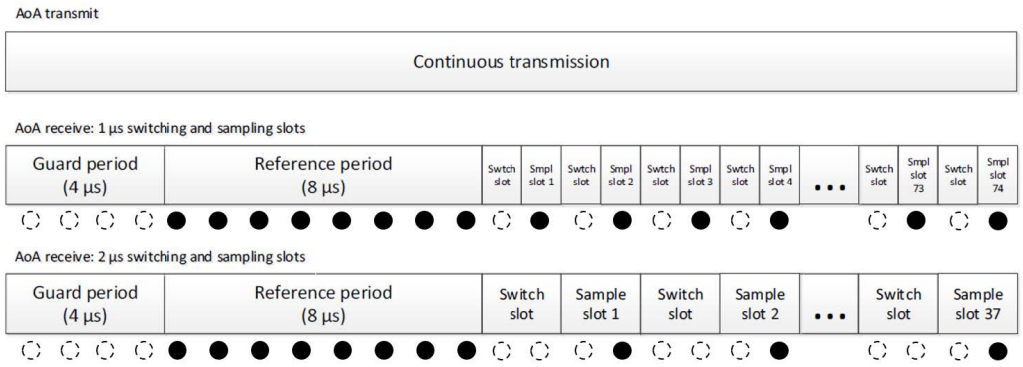


Figure 1.1: Sampling of CTE.

Once a CTE has started, the radio samples the In-phase (I) and Quadrature (Q) components of the baseband signal at its native sample rate. The samples are then downsampled to a 1 sample/μs rate. The first 4 samples (taken during the guard period) are discarded, and the next 8 samples (taken during the reference period) are stored in the sample buffer. Subsequently, every sample taken during switching slots is discarded, and every sample

taken during sample slots is stored in the sample buffer. In the case of 2-μs slots, only one sample is kept for each sample slot. This procedure is illustrated in Fig. 1.1.

Antenna Switching

The goal of CTE is to sample the same continuous wave on different antennas. Since the radio can only sample one antenna at a time, an RF path switching network must be created among the antennas [2]. The antenna switching pattern, which is the order of addressing the different antennas, can be set using the CTE Receiver and CTE Transmitter Bluetooth APIs. The switching pattern can have any length between 1 and 35. It is repeated over the CTE time period; once the end of the pattern is reached, it starts over and repeats until the end of the CTE signal. Note that the first switch happens after the reference period, meaning that the first antenna in the pattern will be sampled 8 times.

Phase Compensation

Ideally, all antennas should be sampled simultaneously to easily calculate the phase difference between them. However, this is not possible with a single radio. The only way to sample more than one antenna is time division, which results in a time shift in sampling.

Although the incoming signal is mixed and downsampled by the radio such that the central frequency of the channel corresponds to a DC signal in the baseband, the CTE will not have zero frequency in the baseband because oscillators are not always perfectly tuned, resulting in an offset between the ideal carrier frequency and the actual carrier frequency.

Since the reference samples are taken on the same antenna, they can be used to measure the actual phase shift over a 1-μs period. This can be done either by averaging the phase differences between the first 8 samples or by applying a median filter on those phase differences. Once a phase shift for 1 μs is calculated, this value should be multiplied and used to compensate for the phase shift between samples after the reference period due to the

non-sampled switching slots.

1.2 RFID Gen2 Protocol

The RFID Gen2 protocol, also known as EPCglobal Class 1 Generation 2, is a standard for UHF (Ultra High Frequency) RFID communication established by the EPCglobal consortium [3].

RFID (Radio-Frequency Identification) systems consist of readers and tags. Readers, also called interrogators, send out radio waves to communicate with tags. Tags, which are attached to objects, respond with their stored data when they receive these signals. The communication between the reader and the tag in the Gen2 protocol is based on backscatter modulation, where the tag reflects the reader's signal to transmit its information.

RFID tags are typically passive, meaning they do not have an internal power source. Instead, they harvest energy from the radio waves emitted by the reader. This harvested energy powers the tag, enabling it to respond. Each RFID tag is equipped with an antenna that can toggle between two states, Z_0 and Z_1. In the low state (Z_0), the antenna operates as an open-circuit, maximizing its reflection coefficient to backscatter the signal to the receiver. In the high state (Z_1),"perfect" matching occurs, setting the reflection coefficient to zero and maximizing power transfer for RF energy harvesting, where the tag absorbs energy from the signal.

Some of the operations that Gen2 RFID protocol can perform are:

- **Inventory Process:** The Gen2 protocol starts with an inventory process where the reader identifies tags within its range. The reader transmits a continuous wave (CW) signal to power up the tags and then sends a QUERY command. This command initiates the inventory round by setting parameters such as the tag data rate and the number of slots in the framed slotted Aloha (FSA) algorithm.

- **Framed Slotted Aloha (FSA) Algorithm:** This algorithm helps manage multiple tag responses and reduce collisions. The QUERY command defines the number of slots available for tag responses. Each

tag randomly selects a slot to respond with a 16-bit random number (RN16). If a tag's chosen slot is collision-free, it sends the RN16 back to the reader.

- **Tag Identification:** Upon receiving the RN16, the reader responds with an acknowledgment (ACK) containing the RN16, confirming successful communication. The tag then transmits its unique identifier, the Electronic Product Code (EPC), which is typically 96 bits long. This EPC identifies the tag and the object it is attached to.

1.3 Localization Prior Art

Phase measurements [4] inherently introduce ambiguity in distance estimation, typically measured in multiples of the wavelength (for one-way wireless communication) or half-wavelength (for round-trip backscatter communication). This ambiguity can be addressed through various means, such as leveraging multiple wavelengths and increased bandwidth [5], or by incorporating mobility, which involves taking measurements at multiple positions with either the tags in motion [6,7], where neural networks are also deployed in the latter case, or the reader in motion [8–11]. In the case of RFID reader mobility, additional challenges arise, including the need to estimate or determine the reader's location beforehand [12].

Research utilizing CSI-methods [13], [14] and DoA methods [15] leverage the 80 MHz bandwidth of BLE to deliver impressive localization results. However, energy consumption limitations can pose challenges for large-scale applications. Alongside large-scale deployment, maintainability throughout the life cycle of BLE beacon systems becomes a critical evaluation criterion, encompassing deployment, battery replacement, repairs, and more. In [16], the authors propose batteryless BLE tags utilizing energy harvesting as a solution.

Even though excessive bandwidth can yield impressive localization results, Gen2 RFID operation (e.g., in European UHF ISM bands) is not compatible with such bandwidth requirements. Nevertheless, this challenge can

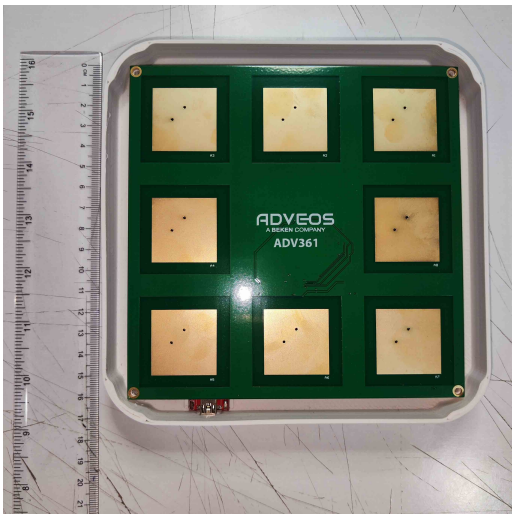
be addressed using alternative methods. Research conducted in [12], [17], [18] utilizes a particle filtering method to facilitate 2D/3D tag localization while [19] improves upon that by introducing a different distance metric of each particle-measurement phase pair, based on geometry arguments, which is more robust to phase measurement noise (e.g., due to multipath) compared to other phase-based localization methods. Moreover, research has explored DoA estimation and localization techniques [20], [21], with the latter employing a bistatic/multistatic Gen2 RFID architecture. Finally, recent research integrates stereo vision, RFID, and BLE to identify individuals within groups in complex outdoor environments [22].

1.4 Equipment

Regarding the Bluetooth receiving equipment, one ore more ADVEOS ADV361 8-element BLE 5.1 locators were employed, as depicted in Fig. 1.2a. The received signal is sampled with each antenna, resulting in the same unknown random phase of the tag ϕ_0 across all antennas. Furthermore, the additional phase due to cabling ϕ_c remains consistent across all antennas due to the construction of the locator. BLE utilizes a total of 40 channels, each with a bandwidth of 2 MHz, resulting in a cumulative bandwidth of 80 MHz spanning from 2.402 GHz to 2.480 GHz. Out of these, 3 channels are dedicated to advertising, while the remaining 37 are for transmitting data packets. However, the BLE waveform utilizes only 1 MHz bandwidth within each channel. As a result, each received packet has a different carrier wavelength, allowing this uniqueness to be leveraged to estimate the position of the tag with better accuracy.

Regarding the RFID equipment, the Impinj Speedway R420 RFID reader was employed, Fig. 1.2b, along with a total of six FlexiRay SF-2110 5 dBi antennas. Finally, the BLE tag that came with the locators and the Alien ALN-9540 (Higgs-2) RFID tag were utilized, depicted in Fig. 1.3.

A Turtlebot2 platform, Fig. 1.4a, was used to precisely collect BLE measurement of grid positions, used to train the Neural Networks in Chapter 5. The TurtleBot2 is a popular, low-cost robot kit with open-source software,



(a) ADVEOS ADV361 Locator.

(b) Impinj Speedway R420 RFID Reader.

Figure 1.2: BLE and RFID Receivers.

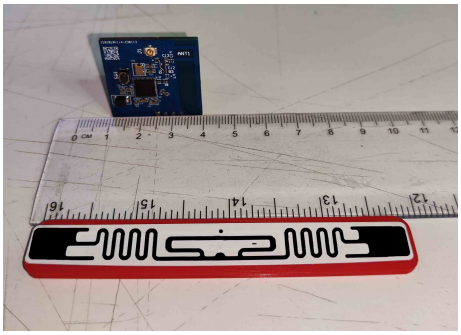


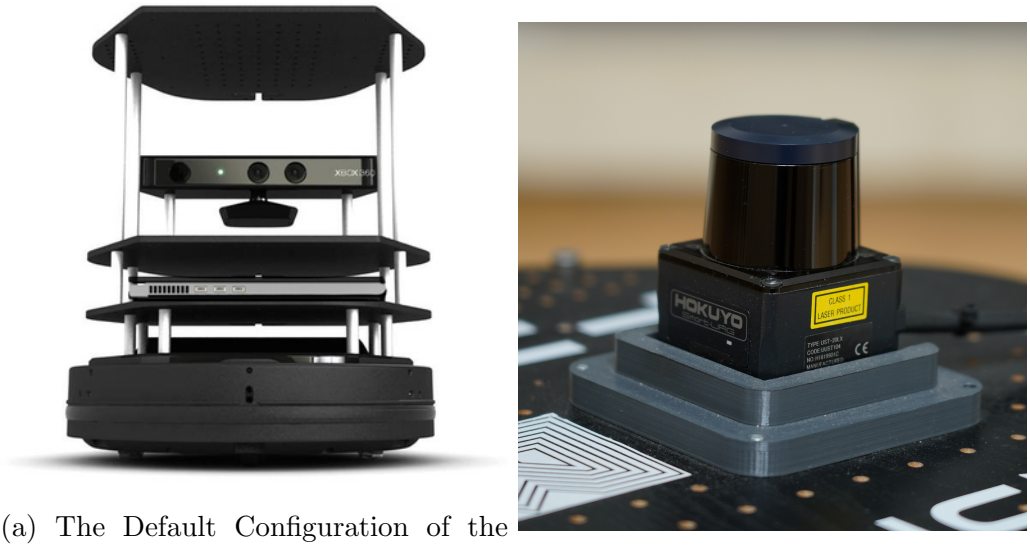
Figure 1.3: BLE and RFID Tags.

commonly used for education and research purposes.

The TurtleBot2 is renowned for its modular design, allowing users to easily customize and extend the robot by adding or removing components to suit specific needs. Its affordability makes it accessible to a wide range of users, from students to researchers. A significant advantage of the TurtleBot2 is its reliance on open-source software, particularly the Robot Operating System (ROS), which facilitates extensive customization and development. Additionally, the strong community support surrounding TurtleBot2 means that users can benefit from shared knowledge, resources, and continuous improvements contributed by other developers.

The core components of the TurtleBot2 include the Kobuki Base, which provides mobility and basic sensor capabilities. A key feature of this base is its built-in rechargeable battery, which powers the entire robot. The TurtleBot2 3D sensor, enabling the robot to perceive its environment in three dimensions. This is complemented by a netbook or single-board computer, such as a Raspberry Pi, which handles data processing and runs the necessary software. The robot also includes customizable mounting plates, allowing users to easily add additional sensors, cameras, or other hardware components to enhance its functionality. These elements together make the TurtleBot2 a versatile and powerful platform for a variety of robotics applications.

For distance measurement and environmental mapping, our robotic system employs a Hokuyo UST-20LX, a sensor that operates based on Light Detection and Ranging (LiDAR) technology, Fig. 1.4b.



(a) The Default Configuration of the Turtlebot2 Platform.

(b) Hokuyo UST-20LX Sensor.

Figure 1.4: Equipment for Collecting Neural Network Datasets.

Chapter 2

Single Locator

2.1 Hyperbolas

Definition 1. *A hyperbola is the set of all points (x, y) in a plane such that the absolute difference of the distances between (x, y) and two fixed points, called the foci, is a positive constant.*

The equation of a hyperbola depends on the position of its foci. The general form of the equation of a hyperbola is:

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1, \tag{2.1}$$

If the two foci are not aligned along the x-axis, a rotation matrix should be applied.

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}, \tag{2.2}$$

where θ is the angle of the counterclockwise rotation required to align the x-axis with the line defined by the two foci. For the entirety of this work, the hyperbola is rotated by $\theta = 90°$, necessitating the application of the rotation matrix to determine the new axes

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = R(\theta) \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -y \\ x \end{bmatrix} \tag{2.3}$$

This results in the new equation

$$\frac{y^2}{a^2} - \frac{x^2}{b^2} = 1, \tag{2.4}$$

However, this hyperbola has its center at the point $O(0,\ 0)$. The equation

of the hyperbola with its center at the point $P(x_0,\ y_0)$ is given by

$$\frac{(y-y_0)^2}{a^2} - \frac{(x-x_0)^2}{b^2} = 1, \tag{2.5}$$

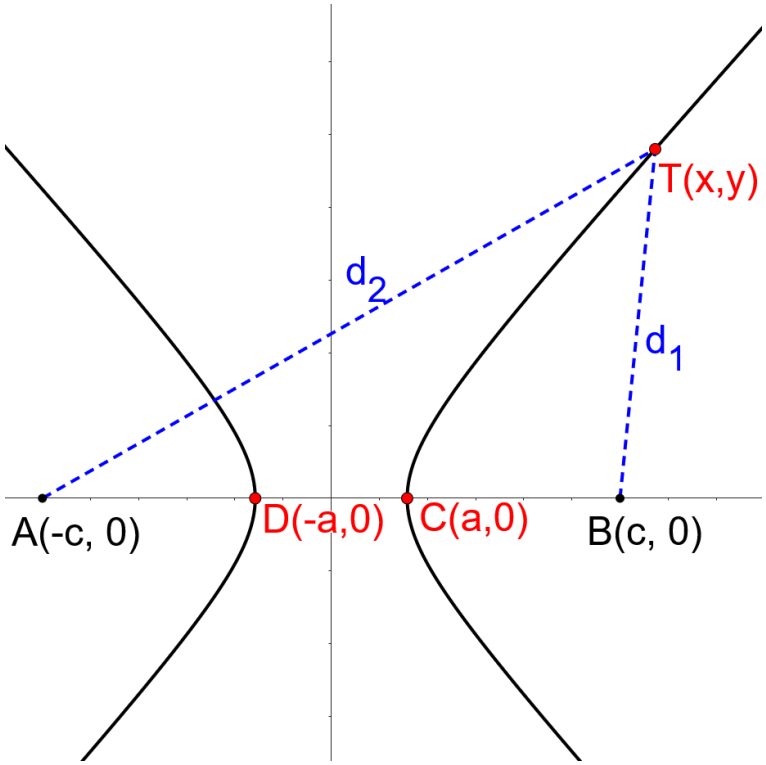


Figure 2.1: Definition of Hyperbola Parameters.

As depicted in Fig.2.1, let a point $T(x,\ y)$ lie on the hyperbola and d_1, d_2 be the distance of this point and the 2 foci, at points $A(-c,\ 0)$ and $B(c,\ 0)$, respectively. Then the parameters of the hyperbola are defined as

- $2a = |d_1 - d_2|$
- $2c = \|A - B\|_2$, the distance of points $A,\ B$
- $b = \sqrt{c^2 - a^2}$

2.1.1 Mathematical Proof of Hyperbolas

A BLE tag at position $\boldsymbol{x}_T \triangleq [x_{tag},\ y_{tag}]^\intercal$ transmits a BLE packet with CTE extension using different carrier frequencies $f_c^{(i)}$. These transmissions occur with an unknown random phase ϕ_0. Upon reception at the locator's location ($\boldsymbol{x}_R$), the received signal is subject to a time delay of

$$\tau = \frac{d_0}{c} = \frac{d_0}{\lambda^{(i)} f_c^{(i)}}, \tag{2.6}$$

where $d_0 \triangleq \|\boldsymbol{x}_R - \boldsymbol{x}_T\|_2$ is the Euclidean distance between the locator and the tag, c is the speed of light, and $\lambda^{(i)}$ is the carrier wavelength. In the case of no multipath and only line-of-sight (LoS) propagation, this is equivalent to a signal received at the locator with a phase of

$$\phi_0 - 2\pi f_c^{(i)} \tau = \phi_0 - 2\pi \frac{d_0}{\lambda^{(i)}}. \tag{2.7}$$

In the case of multipath between the reader and the tag, the (one-way) flat fading channel can be expressed as follows:

$$h = \underbrace{\alpha_0 e^{-jkd_0}}_{\text{direct path}} + \underbrace{\sum_{i=1}^{N_m} \alpha_i e^{-jkd_i}}_{\text{multipath}} \tag{2.8}$$

$$= \underbrace{\alpha_0 e^{-jkd_0}}_{h_0} \underbrace{\left(1 + \sum_{i=1}^{N_m} \frac{\alpha_i}{\alpha_0} e^{-jk(d_i - d_0)}\right)}_{h_m} \tag{2.9}$$

$$= h_0 h_m, \tag{2.10}$$

Here, d_i represents the length of the i-th propagation path (out of N_m), which depends on the locations of the reflectors, as well as the positions of the reader and the tag. Additionally, the complex coefficients $a_i,\ i \in 1, \ldots, N_m$ depend on space geometry, reflector dielectric constants, and antenna gains. Consequently, the phase of the one-way propagation channel h (i.e., from the

tag to the locator) follows:

$$\phi_{\text{prop}} \equiv \angle h = -\frac{2\pi d_0}{\lambda^{(i)}} + \angle h_m, \tag{2.11}$$

where $\angle h_m$ represents the phase introduced by multipath effects. Additionally, there are further delays due to cabling (resulting in a constant phase ϕ_c), as well as phase noise ϕ_n at the locator's receiver chain:

$$\begin{aligned}\phi_R &= \phi_0 + \phi_{\text{prop}} + \phi_c + \phi_n, \\ &= -\frac{2\pi d_0}{\lambda^{(i)}} + \underbrace{\phi_0 + \phi_c}_{\theta} + \underbrace{\angle h_m + \phi_n}_{\phi_n}, \\ &= -\frac{2\pi d_0}{\lambda^{(i)}} + \theta + \phi_n. \end{aligned} \tag{2.12}$$

Receivers typically report a value in the range $[-\pi, \pi)$ or $[0, 2\pi)$. Thus, the measured phase is:

$$\phi_{\text{meas}} = \phi_R \bmod 2\pi, \tag{2.13}$$

$$\overset{1}{=} \left[-\frac{2\pi d_0}{\lambda^{(i)}} \bmod 2\pi + \underbrace{\theta \bmod 2\pi}_{\hat{\Theta}} + \underbrace{\phi_n \bmod 2\pi}_{\hat{\Phi}_n} \right] \bmod 2\pi,$$

$$= \left[-\frac{2\pi d_0}{\lambda^{(i)}} \bmod 2\pi + \hat{\Theta} + \hat{\Phi}_n \right] \bmod 2\pi. \tag{2.14}$$

***Theorem* 1.** *[23, 24] Assuming two antennas A and B located along the same line, if the distance δ_x between the two receiving antennas is such that $\delta_x < \lambda$, then a unique constant distance difference $|d_A - d_B|$, where d_A and d_B represent the Euclidean distances between the tag and antennas A and B respectively, can be determined, regardless of the tag's location.*

Proof. Two antennas denoted as A and B, depicted in Fig.2.2, are considered.

[1] $(\alpha + \beta) \bmod \gamma = [(\alpha \bmod \gamma) + (\beta \bmod \gamma)] \bmod \gamma$.

Their phases are represented as ϕ_A and ϕ_B, respectively, following Eq.2.14:

$$\phi_A = \left(\frac{d_1}{\lambda}2\pi\right) \mod 2\pi \quad \text{and} \quad \phi_B = \left(\frac{d_2}{\lambda}2\pi\right) \mod 2\pi \tag{2.15}$$

Thus,

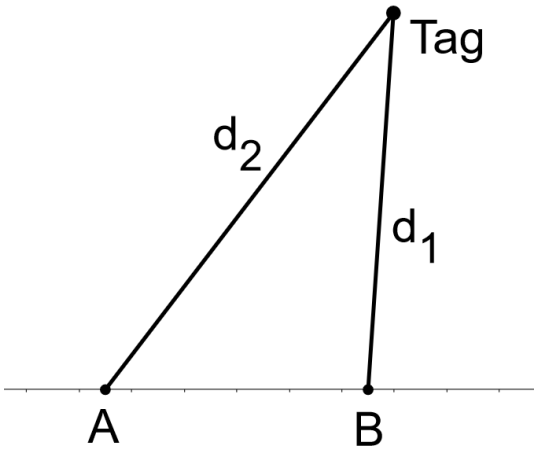


Figure 2.2: Antennas and Tag.

$$\begin{aligned}
\phi_A &= \left(\frac{d_1}{\lambda}2\pi\right) \mod 2\pi \\
&= \left(\frac{d_1 + d_2 - d_2}{\lambda}2\pi\right) \mod 2\pi \\
&= \left(\frac{d_2}{\lambda}2\pi + \frac{d_1 - d_2}{\lambda}2\pi\right) \mod 2\pi \\
&\overset{2}{=} \left(\left(\frac{d_2}{\lambda}2\pi\right) \mod 2\pi + \left(\frac{d_1 - d_2}{\lambda}2\pi\right) \mod 2\pi\right) \mod 2\pi \\
&= \left(\phi_B + \left(\frac{d_1 - d_2}{\lambda}2\pi\right) \mod 2\pi\right) \mod 2\pi
\end{aligned}$$

From the triangular inequality in the triangle defined by the two antennas and the tag, it is observed that $\|d_1 - d_2\| < d_x$.

[2] $(\alpha + \beta) \mod \gamma = [(\alpha \mod \gamma) + (\beta \mod \gamma)] \mod \gamma$.

Suppose that $d_x < \lambda$ and $d_1 > d_2$. Then

$$0 < d_1 - d_2 < \lambda \;\Rightarrow\; 0 < \frac{d_1 - d_2}{\lambda} 2\pi < 2\pi$$

So,

$$\begin{aligned}
\phi_A &= \left(\phi_B + \left(\frac{d_1 - d_2}{\lambda} 2\pi\right) \mod 2\pi\right) \mod 2\pi \\
&= \left(\phi_B + \frac{d_1 - d_2}{\lambda} 2\pi\right) \mod 2\pi \\
&\Rightarrow \phi_B + \frac{d_1 - d_2}{\lambda} 2\pi = \phi_A + 2k\pi \\
&\Rightarrow \phi_A - \phi_B = \frac{d_1 - d_2}{\lambda} 2\pi - 2k\pi
\end{aligned}$$

Since $0 < \phi_A,\ \phi_B < 2\pi$, two cases need to be checked. If $\phi_A > \phi_B$ then

$$0 < \phi_A - \phi_B < 2\pi \Rightarrow 0 < \frac{d_1 - d_2}{\lambda} 2\pi - 2k\pi < 2\pi \Rightarrow \boxed{k = 0}$$

Else if $\phi_A < \phi_B$ then

$$-2\pi < \phi_A - \phi_B < 0 \Rightarrow -2\pi < \frac{d_1 - d_2}{\lambda} 2\pi - 2k\pi < 0 \Rightarrow \boxed{k = 1}$$

The case of $d_1 < d_2$ needs to be examined. Then

$$-\lambda < d_1 - d_2 < 0 \;\Rightarrow\; -2\pi < \frac{d_1 - d_2}{\lambda} 2\pi < 0$$

So,

$$\begin{aligned}
\phi_A &= \left(\phi_B + \left(\frac{d_1 - d_2}{\lambda} 2\pi\right) \mod 2\pi\right) \mod 2\pi \\
&= \left(\phi_B + \frac{d_1 - d_2}{\lambda} 2\pi + \cancel{2\pi}\right) \mod 2\pi \\
&\Rightarrow \phi_B + \frac{d_1 - d_2}{\lambda} 2\pi = \phi_A + 2k\pi \\
&\Rightarrow \phi_A - \phi_B = \frac{d_1 - d_2}{\lambda} 2\pi - 2k\pi
\end{aligned}$$

If $\phi_A > \phi_B$ then

$$0 < \phi_A - \phi_B < 2\pi \Rightarrow 0 < \frac{d_1 - d_2}{\lambda} 2\pi - 2k\pi < 2\pi \Rightarrow \boxed{k = -1}$$

Else if $\phi_A < \phi_B$ then

$$-2\pi < \phi_A - \phi_B < 0 \Rightarrow -2\pi < \frac{d_1 - d_2}{\lambda} 2\pi - 2k\pi < 0 \Rightarrow \boxed{k = 0}$$

It can now be concluded that

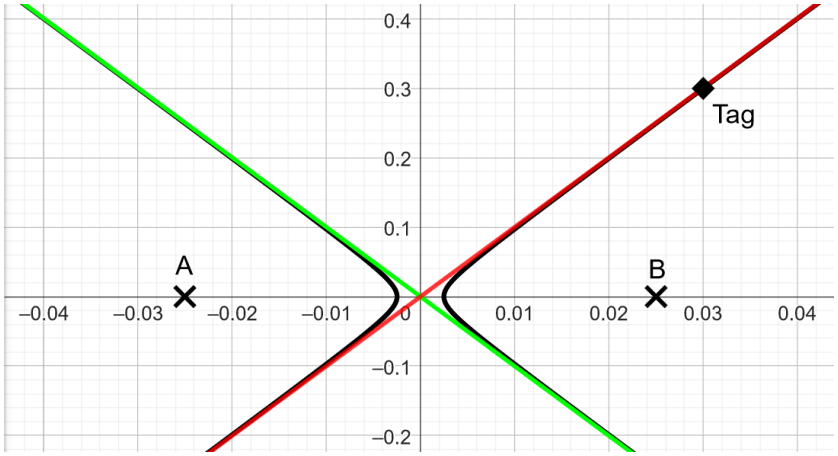
$$\Delta\phi = \phi_A - \phi_B = \begin{cases} \frac{d_1 - d_2}{\lambda} 2\pi & , d_1 > d_2 \text{ and } \phi_A > \phi_B \\ \frac{d_1 - d_2}{\lambda} 2\pi - 2\pi & , d_1 > d_2 \text{ and } \phi_A < \phi_B \\ \frac{d_1 - d_2}{\lambda} 2\pi + 2\pi & , d_1 < d_2 \text{ and } \phi_A > \phi_B \\ \frac{d_1 - d_2}{\lambda} 2\pi & , d_1 < d_2 \text{ and } \phi_A < \phi_B \end{cases} \tag{2.16}$$

One can now solve for $d_1 - d_2$ and conclude that

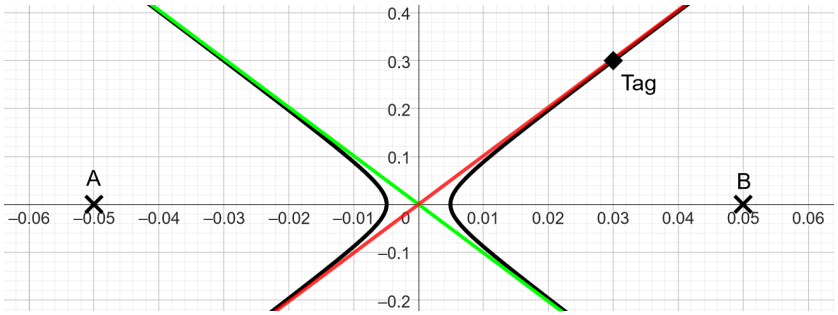
$$d_1 - d_2 = \begin{cases} \frac{\phi_A - \phi_B}{2\pi} \lambda & , d_1 > d_2 \text{ and } \phi_A > \phi_B \\ \frac{\phi_A - \phi_B}{2\pi} \lambda + \lambda & , d_1 > d_2 \text{ and } \phi_A < \phi_B \\ \frac{\phi_A - \phi_B}{2\pi} \lambda - \lambda & , d_1 < d_2 \text{ and } \phi_A > \phi_B \\ \frac{\phi_A - \phi_B}{2\pi} \lambda & , d_1 < d_2 \text{ and } \phi_A < \phi_B \end{cases} \tag{2.17}$$

□

2.1.2 Linear Approximation of Hyperbolas



(a) Antennas Distance = 5 cm.



(b) Antennas Distance = 10 cm.

Figure 2.3: Linear Approximation of Hyperbolas for Different Antenna Distances.

As mentioned before in Section2.1 a constant distance difference can be translated into a hyperbola equation because all the parameters are defined.

***Theorem* 2.** *If the distance between the two foci of a hyperbola is small, then the asymptotic line provides a good approximation of the hyperbola.*

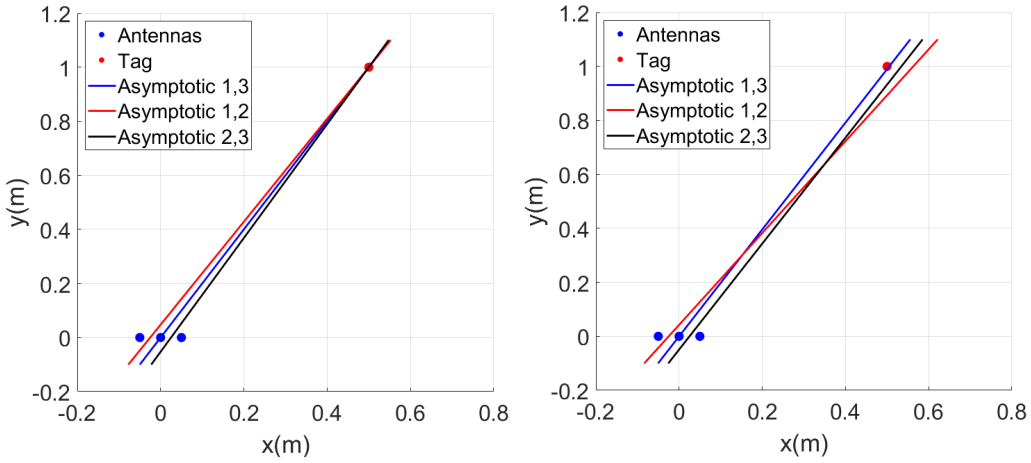
The latter can be visualized in Fig.2.3, where the two foci were at distances of 5 and 10 cm. This implies that we can utilize the asymptotic lines of the hyperbolas to simplify the model. For the hyperbola defined by Eq.2.5,

the asymptotic lines are expressed as:

$$y - y_0 = \pm\frac{a}{b}(x - x_0), \tag{2.18}$$

where $P(x_0,\ y_0)$ represents the center of the hyperbola, and a and b are the previously calculated parameters of the hyperbola.

2.1.3 2D Simulation Results



(a) Intersection Without Noise. (b) Intersection With Noise.

Figure 2.4: Tag's Estimated Position.

If more antennas are deployed under the condition $\delta_x < \lambda$, and in a noise-free environment, the tag's position can be estimated as the common intersection of the hyperbolas, as depicted in Fig.2.4a. However, in real-world noisy environments, finding an accurate estimate becomes challenging due to the doubled noise variance introduced by deploying phase differences. Moreover, when the antennas' pairs are in close proximity, even a slight change in the slope of the lines can lead to intersections very close to the antennas, as depicted in Fig.2.4b. Consequently, this method is not reliable in such scenarios.

2.1.4 2D Experimental Results

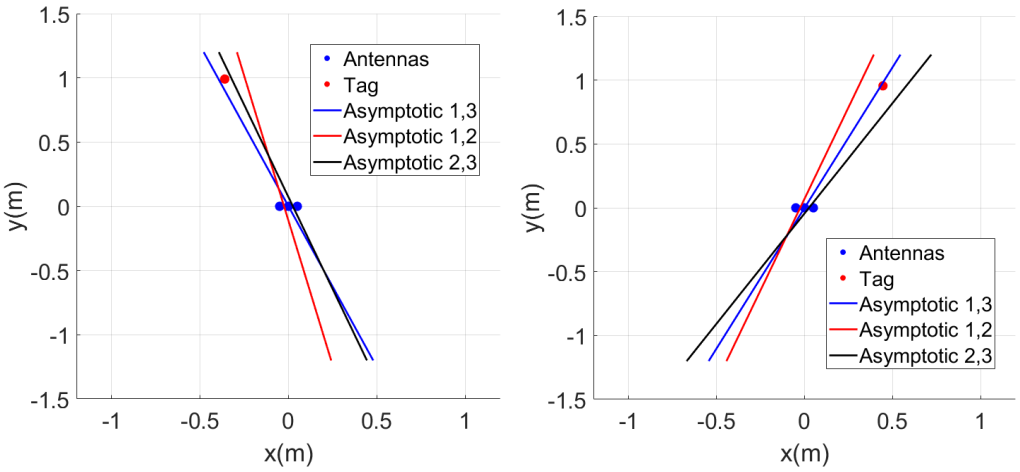
The setup for this experiment requires a single stationary locator and a BLE tag mounted on a stand, rotating around the locator from $-75°$ to $75°$. To measure the change of $5°$ at each position, a printed protractor is used. The tag maintains a constant distance from the locator throughout the rotation. To achieve this, the edges of a 1 m rope are glued to the locator and the stand, as depicted in Fig. 2.5.

(a) Front View of the Setup.

(b) Back View of the Setup.

Figure 2.5: Experimental Setup of Hyperbolas Intersection with One Locator.

The simulation results can be replicated and validated by the experimental findings. In this scenario, obtaining a position estimate is challenging, as illustrated in Fig.2.6



(a) Tag to the Left of the Locator. (b) Tag to the Right of the Locator.

Figure 2.6: Inability to Determine the Tag's Position.

2.2 Direction of Arrival (DoA)

Definition 2. *Direction of Arrival (DoA) refers to the angle at which a propagating wavefront, such as a radio, sound, or electromagnetic wave, arrives at a receiving sensor or antenna array.*

Determining the DoA is crucial in various applications including radar, sonar, wireless communications, and signal processing, as it allows for the localization and tracking of the source of the wavefront.

2.2.1 DoA Estimation Methods

Some of the techniques used to determine the DoA of a signal will be summarized later [25].

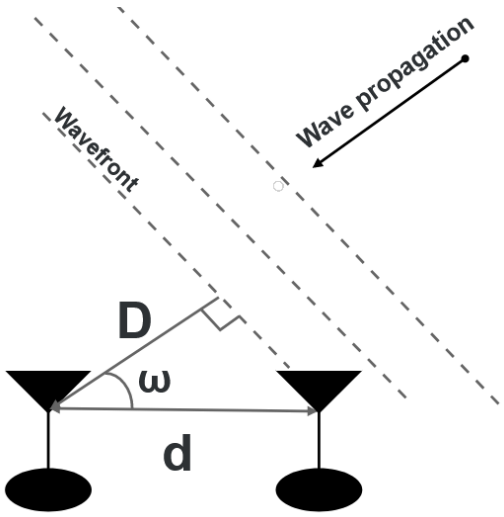


Figure 2.7: Calculate the AoA.

The trivial solution

Assume the use of two receiver antennas as depicted in Fig. 2.7. If the incoming signal hits these antennas at an angle, it results in a phase difference ($\Delta\phi$) because the wavefront propagates at the speed of light (c) and takes more time to reach the farther antenna. This phase shift can be translated into the distance between the wavefront and the farther antenna:

$$D = \Delta\phi \frac{\lambda}{2\pi} \tag{2.19}$$

If the receivers are far enough from the transmitter, it can be assumed that they are in the far field of the transmitter.

Definition 3. *The far field, also known as the Fraunhofer region, is the region of the electromagnetic field surrounding an antenna or scattering object where the angular field distribution is essentially independent of the distance from the source.*

In this region, electromagnetic waves are assumed to propagate in parallel, and the wavefronts are approximately planar. In this case, the angle of the received signal can be determined using trigonometric functions and the

previously calculated distance D:

$$\theta = \arccos\left(\frac{D}{d}\right), \tag{2.20}$$

where d is the distance between the antennas.

Classical Beamformer

Beamforming is a technique utilized to enhance the signal-to-noise ratio (SNR) of received signals, eliminate undesirable interference sources, and concentrate transmitted signals on specific locations. The classical or conventional beamformer method, also known as the delay-and-sum method or Bartlett method, is commonly employed.

Consider an Uniform Linear Array (ULA) consisting of M elements. $A(\theta)$ represents the steering of the antenna array:

$$\mathbf{A}(\theta) = \left[1 \quad \exp\left(\frac{j2\pi d\sin(\theta)}{\lambda}\right) \quad \cdots \quad \exp\left(\frac{j2\pi(M-1)d\sin(\theta)}{\lambda}\right)\right]^{\intercal}, \tag{2.21}$$

where d is the distance between adjacent antennas, λ is the wavelength of the signal, and θ represents the angle of arrival (AoA). The receiving signal as a function of time t is given by:

$$\mathbf{x} = \mathbf{A}(\theta)\mathbf{s}(t) + \mathbf{n}(t), \tag{2.22}$$

where $\mathbf{s}$ is the transmitted signal and $\mathbf{n}$ is the additive noise. The covariance matrix of the received signal can be approximated using Eq.2.22 as:

$$\mathcal{R}_{xx} \approx \frac{1}{N}\sum_{t=1}^{N}\mathbf{x}(t)\mathbf{x}^{H}(t), \tag{2.23}$$

where N is the total time samples of the receiving signal and H represents the Hermitian transpose of a matrix. The objective of classical beamforming

is to maximize the output power as a function of the angle:

$$P_{\text{classical}}(\theta) = A(\theta)\mathcal{R}_{xx}A(\theta)^H. \tag{2.24}$$

The direction that produces the highest output power is the estimate of the desired signal's direction.

Multiple Signal Classification (MUSIC)

The fundamental concept behind the MUSIC algorithm [26] is to perform eigenvalue decomposition on the covariance matrix $\mathcal{R}_{xx}$ of the array output data:

$$\mathcal{R}_{xx} = VUV^{-1}, \tag{2.25}$$

where U is a diagonal matrix containing the eigenvalues and V contains the corresponding eigenvectors of $\mathcal{R}_{xx}$. The objective function that needs to be maximized is:

$$P_{\text{MUSIC}}(\theta) = \frac{1}{A(\theta)VV^HA(\theta)^H}. \tag{2.26}$$

2.2.2 Hyperbolas DoA Estimation

The Hyperbolas Direction of Arrival (DoA) estimation technique relies on the mathematical proof outlined in Section 2.1.1. According to Theorem 2, when the antennas are sufficiently close, the asymptotic lines of the hyperbola offer highly accurate approximations. Hence, determining the DoA entails computing the slope of the asymptotic line. As previously demonstrated, the equation of the asymptotic line is given by:

$$y - y_0 = \pm\frac{a}{b}(x - x_0), \tag{2.27}$$

meaning that the slope of this line is:

$$\text{slope} = \pm\frac{a}{b} \tag{2.28}$$

The default sign for the slope can be set to negative $(-)$ since in Eq.2.17, the value of $d_1 - d_2$ can be either positive or negative. Additionally, because the calculation of the parameter b depends on a^2, c^2, the sign of a can be disregarded. However, utilizing the parameter a with its sign to determine the slope of the asymptotic line can ascertain the correct sign for the slope. Thus, the equation for parameter a and the slope is updated as follows:

$$a = \frac{d_1 - d_2}{2}, \quad \text{and} \quad \text{slope} = -\frac{a}{b} \tag{2.29}$$

To ascertain the DoA/AoA, the angle corresponding to this slope estimate must be computed using trigonometric functions:

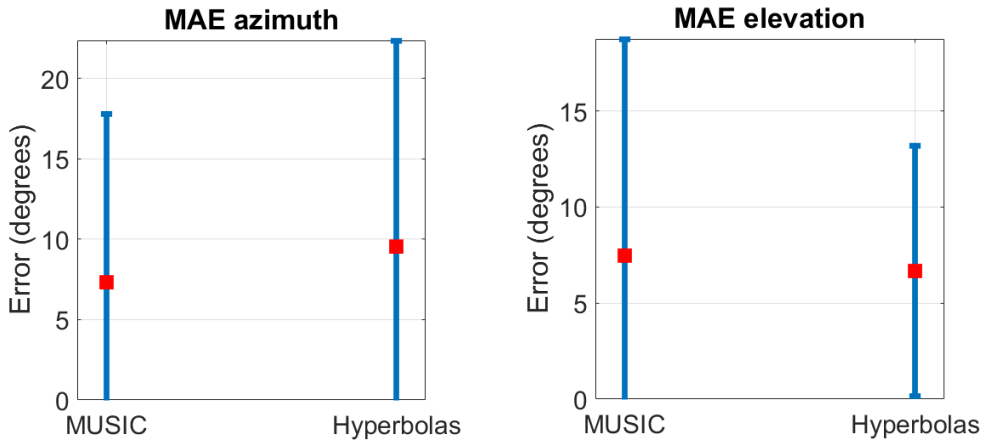
$$\omega = \arctan(\text{slope}) = \arctan\left(-\frac{a}{b}\right). \tag{2.30}$$

2.2.3 Comparison of Methods

It can be proved that 3D DoA can be estimated using the method of hyperbolas and shapes of revolution. Briefly, the hyperbolas method needs to be applied to two pairs of antennas, one at y-axis and one at z-axis. The complete proof can be found in Section 3.2 in the next chapter.

The DoA estimation method using hyperbolas should be compared to an existing DoA estimation method known for its high accuracy. One such method is the MUSIC algorithm. As shown in Fig. 2.8a, the azimuth estimates from the hyperbolas method are very close to those from the MUSIC algorithm, both being under 10° with a difference of only 2°. On the other hand, the elevation estimates exhibit a slightly different behavior; both algorithms result in a mean absolute error (MAE) under 8°, with the hyperbolas method having a slight advantage, as depicted in Fig. 2.8b.

Both algorithms produce excellent results. However, it is important to note that the high sensitivity of the hyperbolas method to noise can lead to slightly higher errors compared to the MUSIC algorithm, which is less sensitive to noise. The full results for the DoA estimation are summarized in Table 2.1.



(a) MAE of Azimuth Estimation. (b) MAE of Elevation Estimation.

Figure 2.8: MAE of DoA Estimation.

Measurements	MAE (degrees)			
	Azimuth		Elevation	
	MUSIC	Hyperbolas	MUSIC	Hyperbolas
Raw	12.32	12.29	11.88	12
Buffered	7.33	9.53	7.45	6.66
	RMSE (degrees)			
	Azimuth		Elevation	
	MUSIC	Hyperbolas	MUSIC	Hyperbolas
Raw	21.97	21.51	20.84	17.15
Buffered	12.78	16	13.53	9.32

Table 2.1: DoA Comparison Results.

2.3 Localization Method

In the previous section, a proof was presented demonstrating that position estimates of a tag using a single locator are not reliable, but a good DoA estimation was provided. The objective of this section is to determine the distance between the tag and the locator and combine it with the previously calculated DoA line to derive an estimate of the tag's position. Below, a mathematical proof of distance estimation is provided.

2.3.1 Mathematical Proof of Distance Estimation

Measurements obtained only from one antenna at various frequencies (each BLE packet corresponds to a different carrier frequency). Given the stationary nature of both the tag and the antenna, the distance between them remains constant over time. The phase of the received signal at the antenna, corresponding to frequencies f_1 and f_2 (with $f_2 > f_1$), denoted as ϕ_1 and ϕ_2 respectively, follows the Eq.2.14

$$\phi_1 = \left(\frac{2\pi d}{\lambda_1}\right) \mod 2\pi \quad \text{and} \quad \phi_2 = \left(\frac{2\pi d}{\lambda_2}\right) \mod 2\pi \tag{2.31}$$

Thus,

$$\begin{aligned}
\phi_2 &= \left(\frac{2\pi d}{\lambda_2}\right) \mod 2\pi \\
&= \left(\frac{2\pi d}{\lambda_2} + \frac{2\pi d}{\lambda_1} - \frac{2\pi d}{\lambda_1}\right) \mod 2\pi \\
&= \left(\frac{2\pi d(\lambda_1 - \lambda_2)}{\lambda_1\lambda_2} + \frac{2\pi d}{\lambda_1}\right) \mod 2\pi \\
&\overset{3}{=} \left[\left(\frac{2\pi d(\lambda_1 - \lambda_2)}{\lambda_1\lambda_2}\right) \mod 2\pi + \left(\frac{2\pi d}{\lambda_1}\right) \mod 2\pi\right] \mod 2\pi \\
&= \left[\left(\frac{2\pi d(\lambda_1 - \lambda_2)}{\lambda_1\lambda_2}\right) \mod 2\pi + \phi_1\right] \mod 2\pi
\end{aligned}$$

It needs to be checked if $m \equiv \dfrac{2\pi d(\lambda_1 - \lambda_2)}{\lambda_1\lambda_2} \in [0, 2\pi)$.
For $\lambda_1 > \lambda_2 \Rightarrow m > 0$

$$\begin{aligned}
m < 2\pi &\Rightarrow \frac{\cancel{2\pi} d(\lambda_1 - \lambda_2)}{\lambda_1\lambda_2} < \cancel{2\pi} \\
&\Rightarrow d(\lambda_1 - \lambda_2) < \lambda_1\lambda_2 \\
&\Rightarrow d < \frac{\lambda_1\lambda_2}{\lambda_1 - \lambda_2} = \frac{\frac{c^2}{f_1 f_2}}{\frac{c}{f_1} - \frac{c}{f_2}} = \frac{c}{f_2 - f_1}
\end{aligned}$$

[3] $(a+b) \mod 2\pi = (a \mod 2\pi + b \mod 2\pi) \mod 2\pi$

$$d < \frac{c}{\Delta f} \overset{4}{=} \frac{3 \cdot 10^8}{2 \cdot w \cdot 10^6} = \frac{150}{w} \text{ m} \tag{2.32}$$

For $d \leq d_{max}$

$$\frac{150}{w} \leq d_{max} \Rightarrow w^* = \left\lfloor \frac{150}{d_{max}} \right\rfloor$$

The results can be summarized by this equation:

$$\begin{aligned} \phi_2 &= \left[\left(\frac{2\pi d \Delta f}{c} \right) \mod 2\pi + \phi_1 \right] \mod 2\pi \\ &= \left[\left(\frac{2\pi d \cdot w}{150} \right) \mod 2\pi + \phi_1 \right] \mod 2\pi \end{aligned} \tag{2.33}$$

If $w \leq w^*$ then $m \in [0, 2\pi)$

$$\begin{aligned} \phi_2 &= \left(\frac{2\pi dw}{150} + \phi_1 \right) \mod 2\pi \\ &= \frac{2\pi dw}{150} + \phi_1 + 2k\pi, \ k \in \mathbb{Z} \\ \Rightarrow \Delta\phi \equiv \phi_2 - \phi_1 &= \frac{2\pi dw}{150} + 2k\pi \end{aligned}$$

Since $0 < \phi_1, \phi_2 < 2\pi$. If $\phi_2 > \phi_1$ then

$$\begin{aligned} &0 < \phi_2 - \phi_1 < 2\pi \\ &\Rightarrow 0 < \frac{2\pi dw}{150} + 2k\pi < 2\pi \\ &\Rightarrow k = 0 \end{aligned}$$

Else if $\phi_2 < \phi_1$ then

$$\begin{aligned} &-2\pi < \phi_2 - \phi_1 < 0 \\ &\Rightarrow -2\pi < \frac{2\pi dw}{150} + 2k\pi < 0 \\ &\Rightarrow k = -1 \end{aligned}$$

[4] w: frequencies index difference

So,

$$d = \begin{cases} \dfrac{150\Delta\phi}{2\pi w} & , \phi_2 > \phi_1 \\ \\ \dfrac{150\Delta\phi}{2\pi w} + \dfrac{150}{w} & , \phi_2 < \phi_1 \end{cases} \tag{2.34}$$

If $w > w^*$ then $m > 2\pi$

$$\begin{aligned} \phi_2 &= \left(\frac{2\pi dw}{150} + 2k_1\pi + \phi_1\right) \mod 2\pi \\ &= \left(\frac{2\pi dw}{150} + \phi_1\right) \mod 2\pi \\ &= \frac{2\pi dw}{150} + \phi_1 + 2k\pi,\ k \in \mathbb{Z} \\ \Rightarrow \Delta\phi \equiv \phi_2 - \phi_1 &= \frac{2\pi dw}{150} + 2k\pi \end{aligned}$$

Since $0 < \phi_1, \phi_2 < 2\pi$. If $\phi_2 > \phi_1$ then

$$\begin{aligned} &0 < \phi_2 - \phi_1 < 2\pi \\ &\Rightarrow 0 < \frac{2\pi dw}{150} + 2k\pi < 2\pi \\ &\Rightarrow k < 0 \end{aligned}$$

Else if $\phi_2 < \phi_1$ then

$$\begin{aligned} &-2\pi < \phi_2 - \phi_1 < 0 \\ &\Rightarrow -2\pi < \frac{2\pi dw}{150} + 2k\pi < 0 \\ &\Rightarrow k < 0 \end{aligned}$$

Since a specific interval for m cannot be defined, so the equation is as follows:

$$d = \frac{150(\Delta\phi - 2k\pi)}{2\pi w},\ k < 0 \tag{2.35}$$

From (2.34) and (2.35), it is derived that

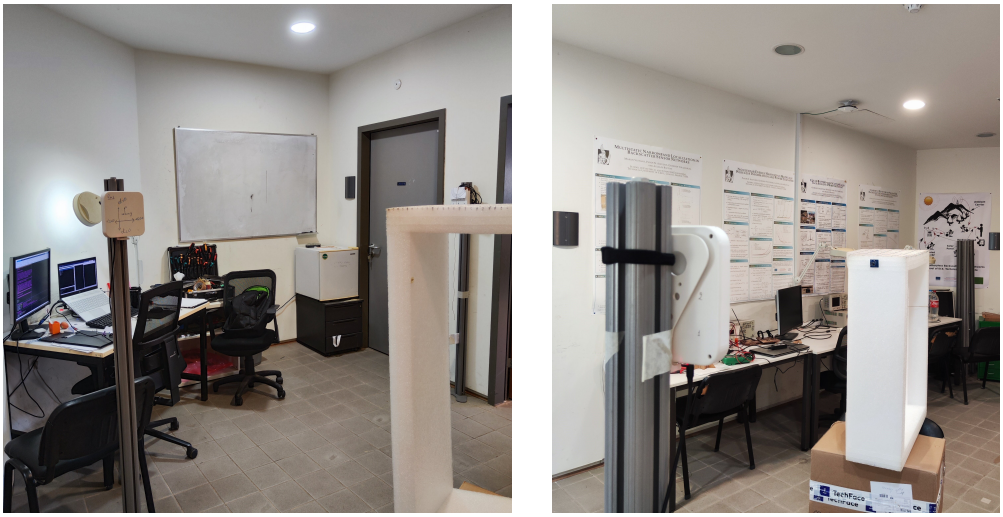
$$d = \begin{cases} \dfrac{150\Delta\phi}{2\pi w} & , w \leq w^* \text{ and } \phi_2 > \phi_1 \\ \dfrac{150\Delta\phi}{2\pi w} - \dfrac{150}{w} & , w \leq w^* \text{ and } \phi_2 < \phi_1 \\ \dfrac{150\Delta\phi}{2\pi w} - \dfrac{150k}{w} & , w > w^*, \quad k < 0 \end{cases} \tag{2.36}$$

This approach offers an alternative proof for the distance estimation equation from [27], which can also be applied to BLE. The two approaches differ fundamentally in the phase model they use; this approach uses a different phase model along with the wrapped phase, whereas [27] uses the unwrapped phase. The equation in [27] corresponds to the first case of distance estimation in this model, but this model also includes two additional cases: one for the negative phase difference and one for $w > w^*$ (though with ambiguity).

2.3.2 Experimental results of CPO removal

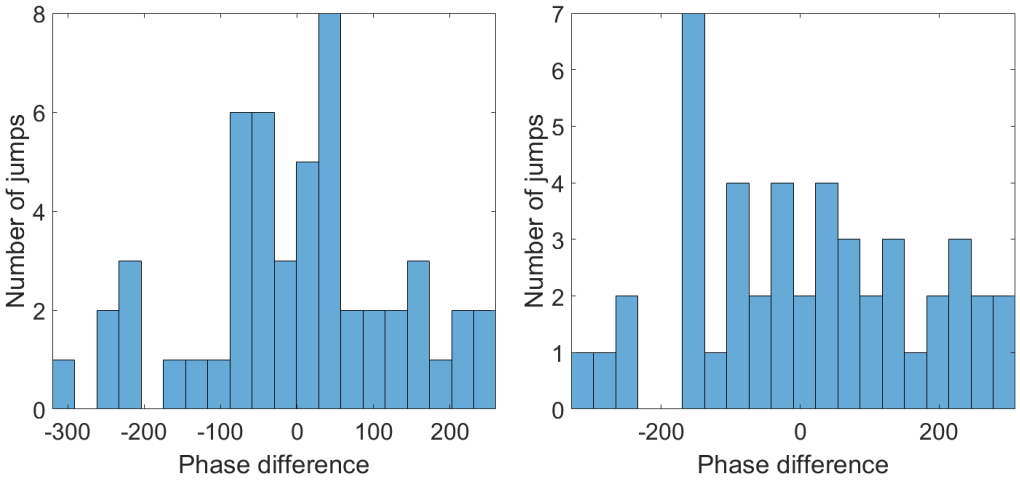
The above estimation technique could not be applied to the current equipment because the BLE tag transmits a BLE packet with an unknown random phase ϕ_0 that varies from packet to packet. One possible reason for this issue is the frequency hopping that occurs in the BLE tag. This frequency hopping results in an additive unknown phase for each packet. In the conducted experiment, the locator and the tag were stationary at a predetermined distance, as shown in Fig. 2.9, to measure the phase resulting from each hop.

The results in Fig.2.10 indicate that a specific hop from f_i to f_j follows a normal distribution (Fig.2.10a), implying randomness that cannot be eliminated without numerous measurements of each hop, or is close to a uniform distribution (Fig.2.10b), implying that randomness cannot be eliminated in general. The problem is that the channels used are 37, resulting in $37^2 = 1369$ different hop combinations, and the hopping is random, making real-time estimations infeasible.

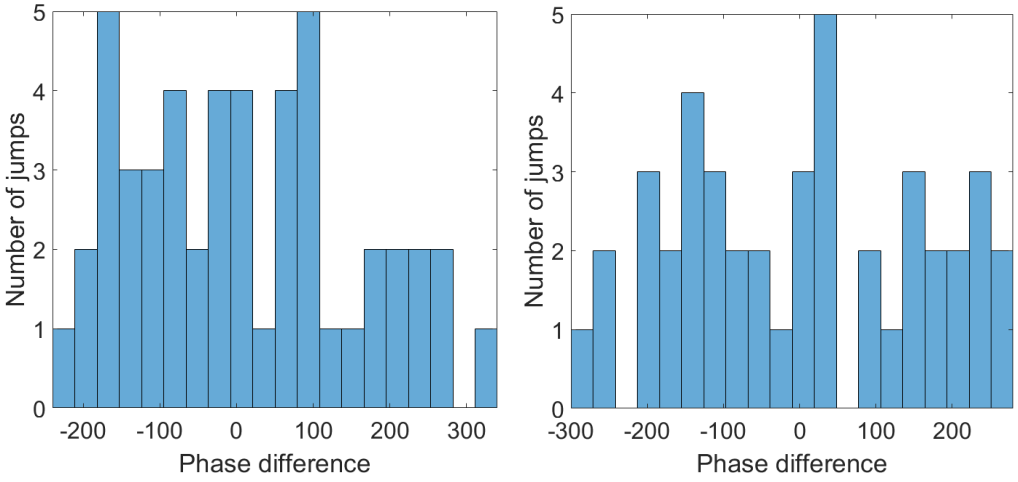


(a) Front View of the Setup. (b) Back View of the Setup.

Figure 2.9: Experimental Setup to Measure the Phase Resulting from Frequency Hopping.



(a) Normal Distributed Phase Differences.



(b) Uniform Distributed Phase Differences.

Figure 2.10: Number of Occurrences of Phase Differences in a Specific Hop.

Chapter 3

Multiple locators

In this chapter, multiple locators are employed to address the limitations of a single locator. The main objective is to obtain an accurate real-time estimate of the tag's position. The BLE tag transmits every 7 ms, allowing for real-time estimation. Additionally, using the "right" topology, the accuracy of these estimates can be significantly improved.

3.1 2D Localization Method

As mentioned in the previous chapter, each locator can produce good DoA estimates. Employing multiple locators means utilizing multiple DoA lines. Each DoA line follows the equation:

$$y - y_0 = -\frac{a}{b}(x - x_0) \Rightarrow y = mx + d, \tag{3.1}$$

where the point $P(x_0,\ y_0)$ is the starting point of the DoA line, m is the slope of the line ($m = -a/b$), and $d = -ax_0/b + y_0$.

Assuming there are N DoA lines, each following Eq. 3.1:

$$y = m_i x + d_i, \quad i = \{1,\ 2,\ \cdots,\ N\} \tag{3.2}$$

A linear system is then formulated as follows:

$$\underbrace{\begin{bmatrix}\mathbf{1} & -\boldsymbol{m}\end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix}y \\ x\end{bmatrix}}_{\boldsymbol{p}} = \boldsymbol{d}, \tag{3.3}$$

where $\mathbf{1}$ is a $N \times 1$ vector with one in every entry, $\boldsymbol{m} = [m_1\ \cdots\ m_N]^{\mathrm{T}}$ and

$\boldsymbol{d} = [d_1 \ \cdots \ d_N]^{\mathrm{T}}$.

The solution to this system, representing the intersection of these N DoA lines, is the estimated tag's position:

$$\boldsymbol{p} = \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{d}. \tag{3.4}$$

3.2 3D Localization Method

The idea of finding the intersection point of N DoA lines to determine the tag's position can be extended into 3D space. Following the proof in Section 2.1.1, a new hyperbola can be defined, which lies on the plane formed by the pair of antennas and the tag, as depicted in Fig. 3.1

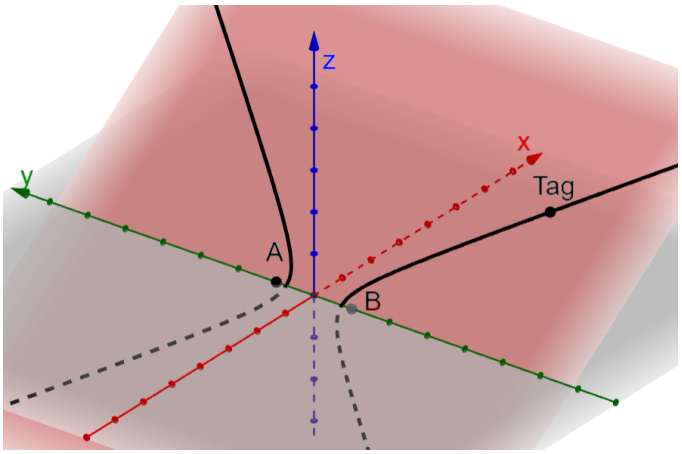


Figure 3.1: Hyperbola in 3D Space.

Rotating this hyperbola around its transverse axis (i.e., the line that connects the two foci) results in a circular hyperboloid (or hyperboloid of revolution) with two sheets, as depicted in Fig. 3.2. Because this hyperboloid is constructed from a specific hyperbola, its intersection with any plane containing the two foci (and thus the line connecting them) produces a hyperbola with the same parameters lying on that plane.

Two angles are required to define a 3D DoA line: azimuth (ϕ) and elevation (θ). Let the antennas lie in the yz-plane and face towards the positive x-semi-axis. Then,

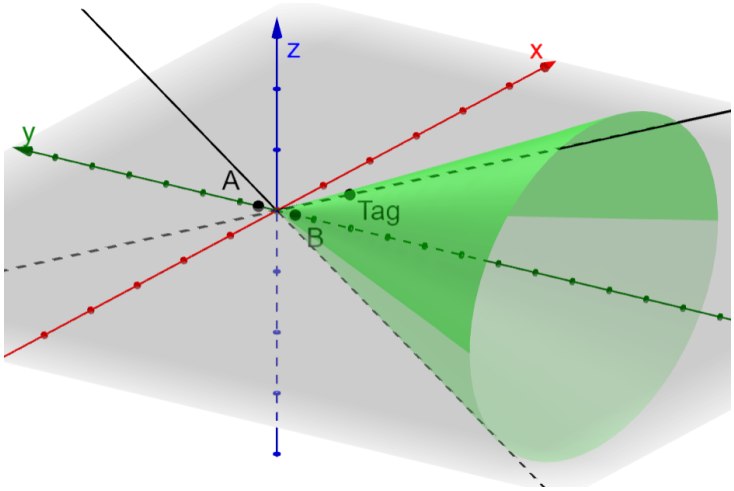


Figure 3.2: Hyperboloid of Revolution of Two Sheets.

- The intersection of a hyperboloid, which is constructed from a hyperbola with its foci on y-axis, and the xy-plane is a hyperbola with the same parameters that lies on xy-plane; the azimuth estimation follows:

$$\hat{\phi} = \arctan\left(-\frac{a_y}{b_y}\right), \tag{3.5}$$

  where a_y, b_y are the parameters of the hyperbola with its foci on y-axis.

- The intersection of a hyperboloid, which is constructed from a hyperbola with its foci on z-axis, and the xz-plane is a hyperbola with the same parameters that lies on xz-plane; the elevation estimation follows:

$$\hat{\theta} = \arctan\left(-\frac{a_z}{b_z}\right), \tag{3.6}$$

  where a_z, b_z are the parameters of the hyperbola with its foci on z-axis.

The final azimuth estimate for each locator is calculated as the mean value of all azimuth estimations obtained from every pair of antennas parallel to the y-axis. Similarly, the final elevation estimation is determined using pairs parallel to the z-axis.

In 3D space, there is no guarantee that the DoA lines will intersect, as they may lie on different planes. To resolve this, the point that minimizes

the distance to the lines needs to be calculated. This involves minimizing the distance from the point to each line, formulating a least-squares problem; let $\boldsymbol{n}_i$ be the unit direction vector of the i-th 3D DoA line:

$$\boldsymbol{n}_i = \begin{bmatrix} \cos(\hat{\theta}_i)\cos(\hat{\phi}_i) \\ \cos(\hat{\theta}_i)\sin(\hat{\phi}_i) \\ \sin(\hat{\theta}_i) \end{bmatrix}, \tag{3.7}$$

and $\boldsymbol{s}_i$ the origin on the i-th 3D DoA line. Then, the point that solves the least-squares problem is [28]:

$$\boldsymbol{p} = \left(\sum_i \left(\mathbb{I} - \boldsymbol{n}_i \boldsymbol{n}_i^{\mathrm{T}} \right) \right)^{-1} \sum_i \left(\mathbb{I} - \boldsymbol{n}_i \boldsymbol{n}_i^{\mathrm{T}} \right) \boldsymbol{s}_i, \tag{3.8}$$

where $\mathbb{I}$ is the identity matrix.

3.2.1 3D Simulation Results

For the 3D simulation results, a comparison with an existing method is necessary. The existing method, provided as a demo code with the locator, is a custom solution developed by ADVEOS, a company based in Athens.

Definition 4. *The Rician K-factor is the ratio between the received power of the strongest path and the sum of powers of all other paths.*

In this simulation, the positioning error needs to be measured across different K-factors. The results are summarized in Fig. 3.3, indicating that hyperbolas are highly sensitive to phase noise compared to the DoA estimation method used in prior art.

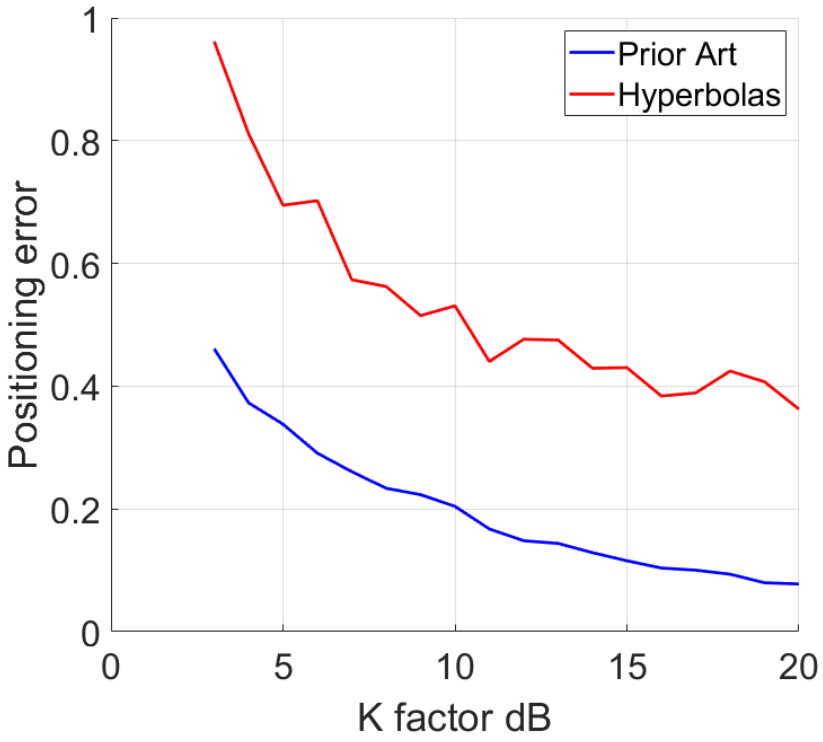


Figure 3.3: Positioning Error over Different K-factors

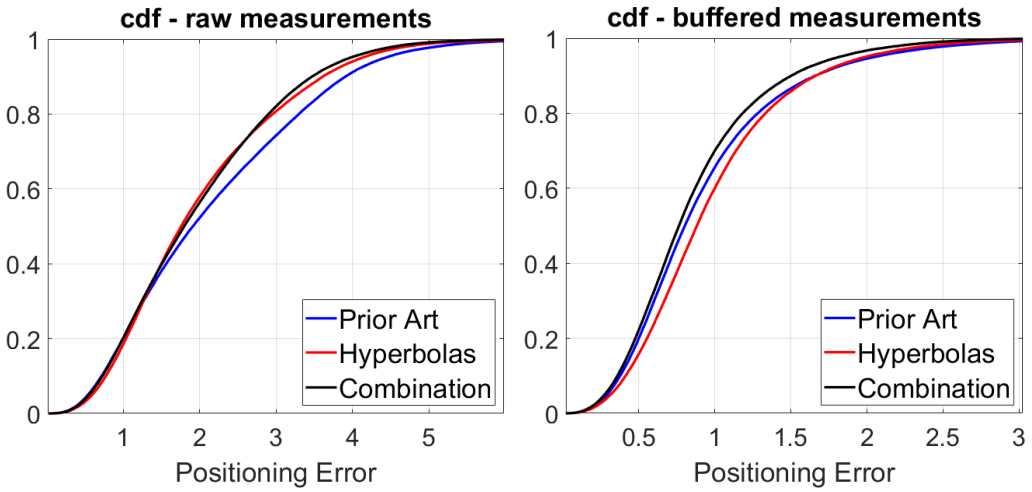
3.2.2 3D Experimental Results

For the 3D experimental results, two different topologies were employed. The first topology utilized four locators positioned at the corners of a room, rotated to converge on a single point at the center of the room. The second topology involved two locators oriented forward. In both configurations, the objective was to measure the positioning error. Additionally, in the experimental results, a hybrid method was implemented, combining the DoA estimates of the prior art algorithm with the hyperbolas.

Additionally, the comparison was conducted using both raw and buffered measurements. Buffering is a smoothing technique designed to reduce the standard deviation of estimators and enhance estimation accuracy. While buffering was applied to the DoA estimates in this case, it could also be applied to the received phases or the hyperbola parameters. The choice of buffering was based on its performance.

First Topology

The measurements for the first topology were collected at ADVEOS and comprised a complete set of measurements on a 9×5 m^2 2D grid at a height of 1.3 m, used to compare the methods. Neither of the algorithms is aware of any of the above parameters of the grid.

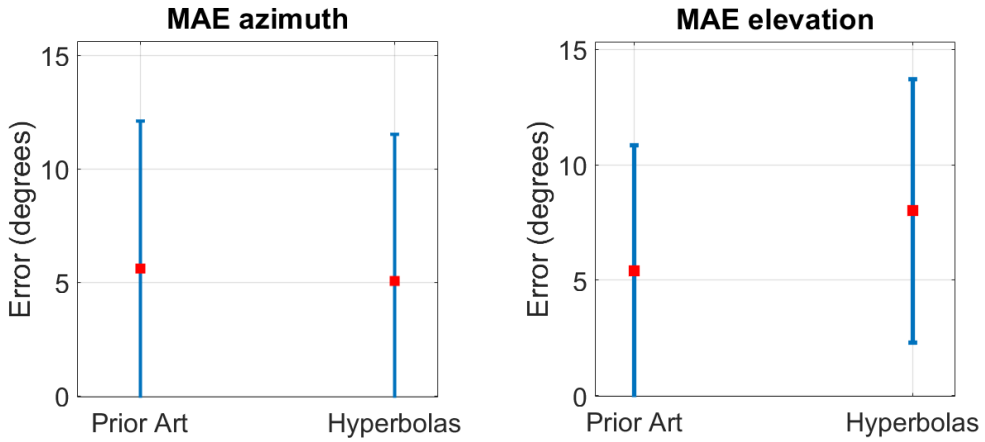


(a) CDF of Raw Measurements. (b) CDF of Buffered Measurements.

Figure 3.4: CDFs of Positioning Error.

As seen in Fig. 3.4a, the hyperbolas' position estimation performs better for raw measurements and has the same performance as the combination of the Prior Art and the hyperbolas. On the other hand, if buffered measurements are deployed, as depicted in Fig. 3.4b, the position estimation of the combination performs better than the other two methods, and the Prior Art algorithm is slightly better than the hyperbolas.

However, CDFs are not the only criterion for comparing methods and discussing their accuracy. This is why the methods were compared based on their accuracy in DoA estimation and position estimation. As seen in Fig. 3.5a, the mean absolute error (MAE) of the azimuth estimate of the two algorithms is very close, only around 5°. On the other hand, the elevation estimate of the hyperbolas is slightly higher in comparison to the Prior Art algorithm, as seen in Fig. 3.5b, but the MAE of the estimation remains small at 8°. The full results for the DoA estimation are summarized in Table 3.1.



(a) MAE of Azimuth Estimation. (b) MAE of Elevation Estimation.

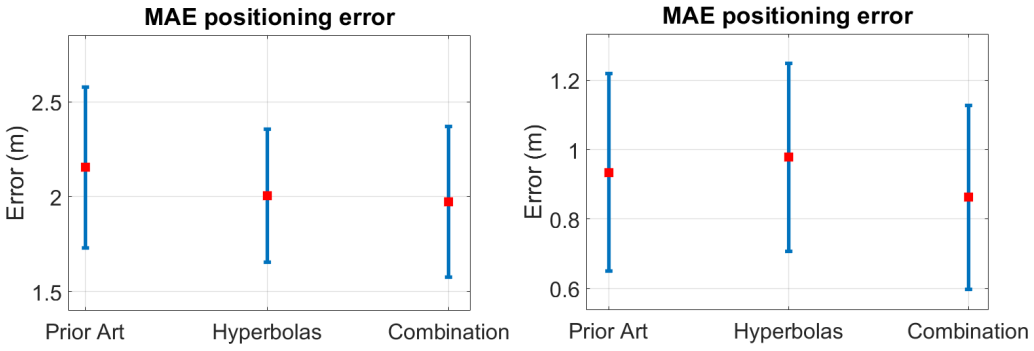
Figure 3.5: MAE of DoA Estimation.

Measurements	MAE (degrees)			
	Azimuth		Elevation	
	Prior Art	Hyperbolas	Prior Art	Hyperbolas
Raw	13.63	10.4	12.48	13.33
Buffered	5.61	5.07	5.39	8.01
	RMSE (degrees)			
	Azimuth		Elevation	
	Prior Art	Hyperbolas	Prior Art	Hyperbolas
Raw	23.88	17.62	21.3	17.15
Buffered	8.57	8.2	7.68	9.84

Table 3.1: Summarized Results for the DoA Estimation of the Two Algorithms.

For the position estimation, the combined method of the hyperbolas and the Prior Art algorithm needs to be included. Similar to the comparison of the CDFs for the position estimation of raw measurements, it is evident that the combination of the methods and the hyperbolas alone provide very similar estimations, both of which are better than the prior art algorithm. In the case of buffered measurements, the positioning error of all methods is lower than before. Here, the combination of the two methods offers the best estimate, while the estimate of the prior art algorithm falls between the other two. It is important to note that the estimations of all three algorithms are

very good for the extensive grid. The full results for the position estimation are summarized in Table 3.2.



(a) MAE of Raw Measurements. (b) MAE of Buffered Measurements.

Figure 3.6: MAE of Position Estimation.

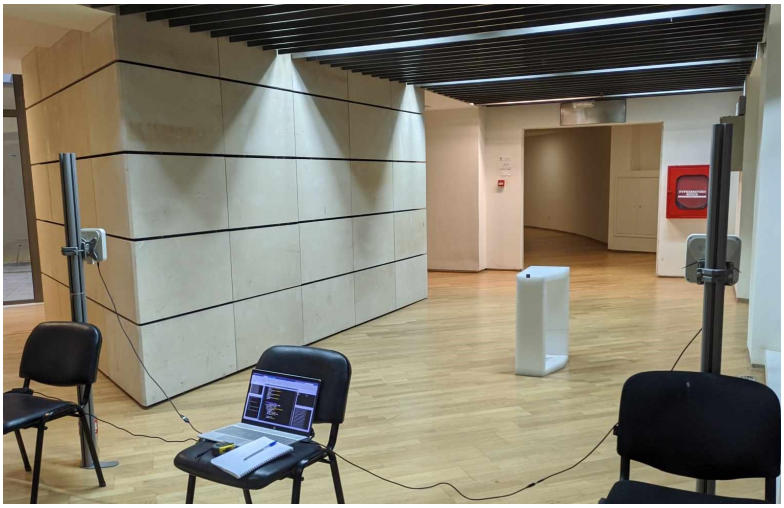
Measurements	MAE (m)		
	Prior Art	Hyperbolas	Combination
Raw	2.16	2.01	1.98
Buffered	0.94	0.98	0.86
	RMSE (m)		
	Prior Art	Hyperbolas	Combination
Raw	2.5	2.3	2.25
Buffered	1.09	1.11	0.99

Table 3.2: Summarized Results for the Position Estimation of the Three Algorithms.

Second Topology

The measurements for the second topology were collected on two different days in different rooms, with the antennas at the same distance apart but at different heights. The setup, as seen in Fig. 3.7, has the two locators standing still with 3 meters between them, facing forward. All the tag's positions were either in front of one locator or between them. The 2D grid in these experiments was 4×2 m^2 with the tag at a height of 0.75 m and 5×3 m^2 with the tag at a height of 1.15 m, on the two respective days. Again, the algorithms do not know the parameters of the grid.

(a) Front View of the Setup.

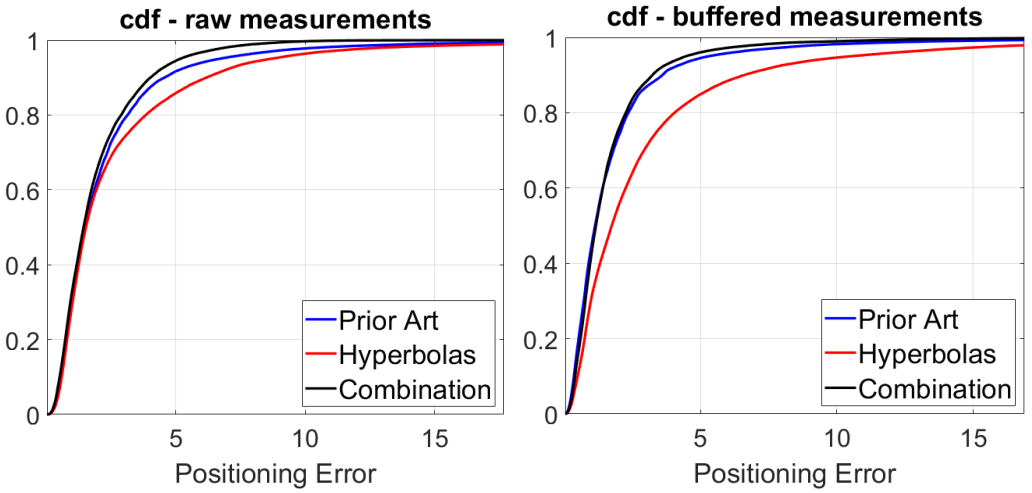


(b) Back View of the Setup.

Figure 3.7: Setup for the Second Topology.

As seen in Fig. 3.8, the combination of the methods improves the position estimates of the prior art algorithm in both cases (raw and buffered measurements). However, the hyperbolas do not have as good estimates as before. This is because this topology is noisier than the previous one, and the hyperbolas are very sensitive to phase noise.

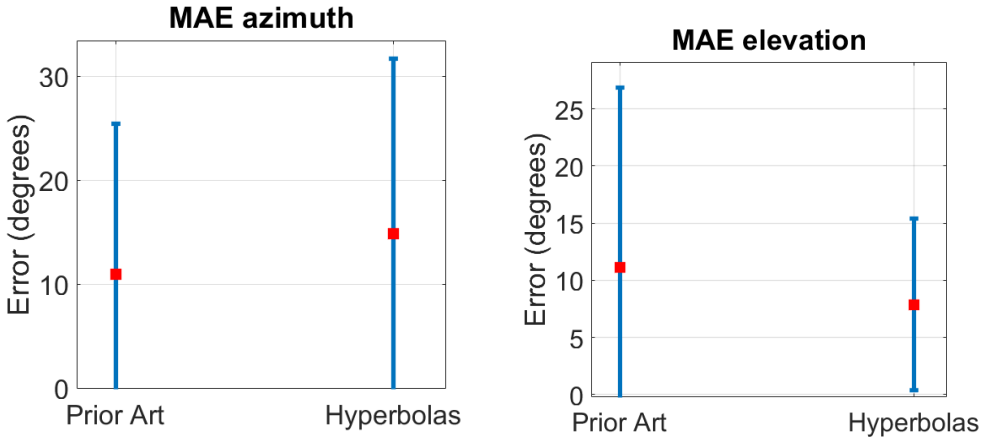
As before, the CDFs are not the only criterion for comparing the algorithms, so accuracy in DoA estimation and position estimation will be used as additional criteria.



(a) CDF of Raw Measurements. (b) CDF of Buffered Measurements.

Figure 3.8: CDFs of Positioning Error.

In this experiment, the behavior of the MAE for azimuth and elevation estimation changes. As seen in Fig. 3.9a, the mean absolute error (MAE) of the azimuth estimate for the Prior Art algorithm is clearly better than that for the hyperbolas. Conversely, the elevation estimates for the hyperbolas are noticeably better than those for the Prior Art algorithm, as shown in Fig. 3.9b. The full results for the DoA estimation are summarized in Table 3.3.

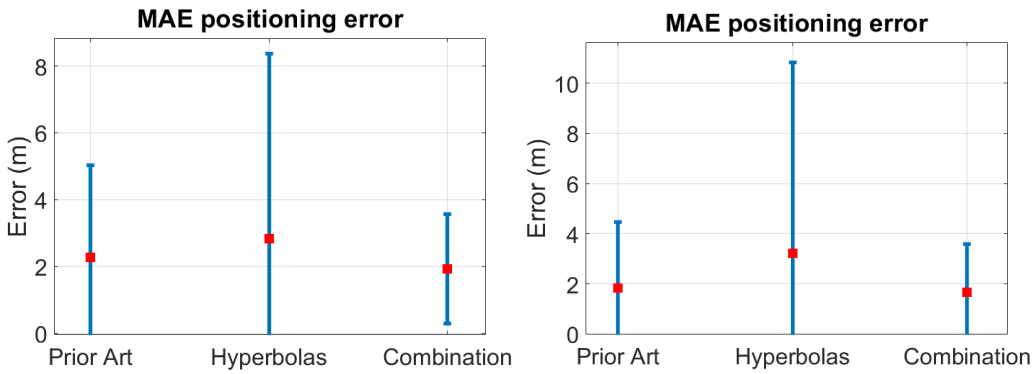


(a) MAE of Azimuth Estimation. (b) MAE of Elevation Estimation.

Figure 3.9: MAE of DoA Estimation.

Measurements	MAE (degrees)			
	Azimuth		Elevation	
	Prior Art	Hyperbolas	Prior Art	Hyperbolas
Raw	17.27	16.86	16.42	13.31
Buffered	10.97	14.87	11.11	7.88
	RMSE (degrees)			
	Azimuth		Elevation	
	Prior Art	Hyperbolas	Prior Art	Hyperbolas
Raw	30.32	27.61	28.05	18.65
Buffered	18.14	22.42	19.25	10.89

Table 3.3: Summarized Results for the DoA Estimation of the Two Algorithms.



(a) MAE of Raw Measurements. (b) MAE of Buffered Measurements.

Figure 3.10: MAE of Position Estimation.

For the position estimation, the combined method of the hyperbolas and the Prior Art algorithm will again be included. Similar to the comparison of the CDFs for the position estimation, it is evident that the hyperbolas have greater error and are noisier, as seen in Fig.3.10a. However, the combination of the methods in raw measurements is clearly better than the prior art, but for the buffered measurements, the error of these two is very similar, as seen in Fig. 3.10b. It is important to mention that the errors in this topology are increased, which occurs because of the locators' gain decreasing as we move to the side of the locator. Another reason is that the two rooms have strong reflectors. In Fig. 3.7 the wall is made of a very reflective material, and

the ceiling has a unique shape that is very harmful to the reliability of the measurements. The other room has fewer reflectors, but the measurement space was closer to the walls as the room was smaller. The full results for the position estimation are summarized in Table 3.2.

Measurements	MAE (m)		
	Prior Art	Hyperbolas	Combination
Raw	2.28	2.84	1.93
Buffered	1.84	3.22	1.68
	RMSE (m)		
	Prior Art	Hyperbolas	Combination
Raw	3.57	6.21	2.53
Buffered	3.21	8.26	2.53

Table 3.4: Summarized Results for the Position Estimation of the Two Algorithms.

Chapter 4

Comparative Analysis of BLE and RFID

In this chapter, RFIDs are deployed to compare their DoA and position estimation accuracy with that of BLE. A shorter version of this work was presented in the "18th Annual International Conference on RFID" [29]. The motivation for using RFIDs is their very low cost and low energy consumption compared to BLE.

Before comparing the experimental results, the mathematical proof of the hyperbolas needs to be revised to align with the two-way/roundtrip nature of propagation.

4.1 Revised Proof of Hyperbolas

As before, an RFID tag is deployed at position $\boldsymbol{x}_T \triangleq [x_{tag},\ y_{tag}]^\intercal$. The RFID reader transmits a signal with carrier frequency f_c and phase ϕ_0.

Following the derivation of the mathematical proof in Section 2.1.1, the phase of the received signal in the two-way/roundtrip propagation channel h_m^2 is:

$$\phi_{\text{prop}} \equiv \angle h_m^2 = -\frac{4\pi d_0}{\lambda} + 2\angle h_m, \tag{4.1}$$

Apart from the delays caused by cabling ϕ_c and phase noise ϕ_n, the RFID tag introduces a term of ϕ_{tag} into the phase measurement, which varies depending

on the impinged RF power at the tag [30]:

$$\begin{aligned}\phi_R &= \phi_0 + \phi_{\text{prop}} + \phi_{\text{tag}} + \phi_c + \phi_n, \\ &= -\frac{4\pi d_0}{\lambda} + \underbrace{\phi_0 + \phi_c}_{\theta} + \underbrace{\phi_{\text{tag}} + 2\angle h_m + \phi_n}_{\phi_n}, \\ &= -\frac{4\pi d_0}{\lambda} + \theta + \phi_n. \end{aligned} \tag{4.2}$$

Thus, the measured phase in this case is given as follows:

$$\phi_{\text{meas}} = \phi_R \bmod 2\pi, \tag{4.3}$$

$$\begin{aligned}&\overset{1}{=} \left[-\frac{4\pi d_0}{\lambda} \bmod 2\pi + \underbrace{\theta \bmod 2\pi}_{\hat{\Theta}} + \underbrace{\phi_n \bmod 2\pi}_{\hat{\Phi}_n} \right] \bmod 2\pi, \\ &= \left[-\frac{4\pi d_0}{\lambda} \bmod 2\pi + \hat{\Theta} + \hat{\Phi}_n \right] \bmod 2\pi. \end{aligned} \tag{4.4}$$

Following the same steps of the mathematical proof in Section 2.1.1, it can be proven that the new equation for the unique distance difference is given by:

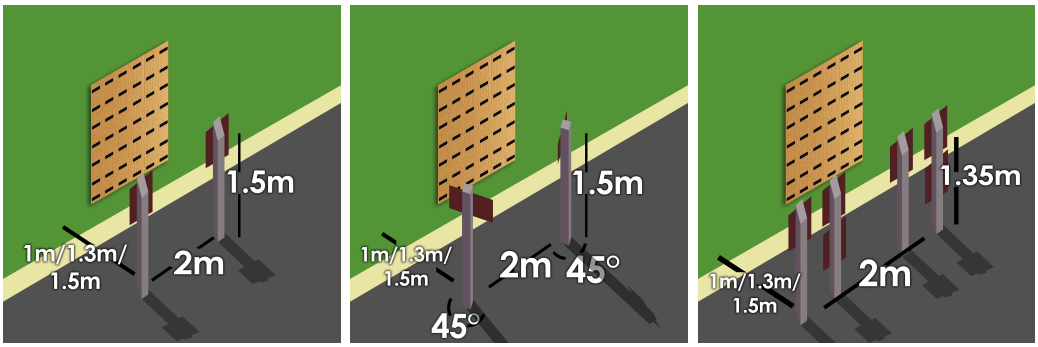
$$d_1 - d_2 = \begin{cases} \dfrac{\phi_A - \phi_B}{4\pi}\lambda, & d_1 > d_2 \text{ and } \phi_A > \phi_B, \\ \dfrac{\phi_A - \phi_B}{4\pi}\lambda + \dfrac{\lambda}{2}, & d_1 > d_2 \text{ and } \phi_A < \phi_B, \\ \dfrac{\phi_A - \phi_B}{4\pi}\lambda - \dfrac{\lambda}{2}, & d_1 < d_2 \text{ and } \phi_A > \phi_B, \\ \dfrac{\phi_A - \phi_B}{4\pi}\lambda, & d_1 < d_2 \text{ and } \phi_A < \phi_B. \end{cases} \tag{4.5}$$

The only difference between Eq. 2.17 and Eq. 4.5 is a factor of 2 at the denominator, since one-way propagation is half of the roundtrip/two-way propagation. Additionally, while BLE utilizes multiple carrier wavelengths, RFID uses only one. Thus, in either RFID or BLE phase measurements, it

is possible to exploit the phase difference between two antennas in order to define a hyperbola.

4.2 Experimental Results

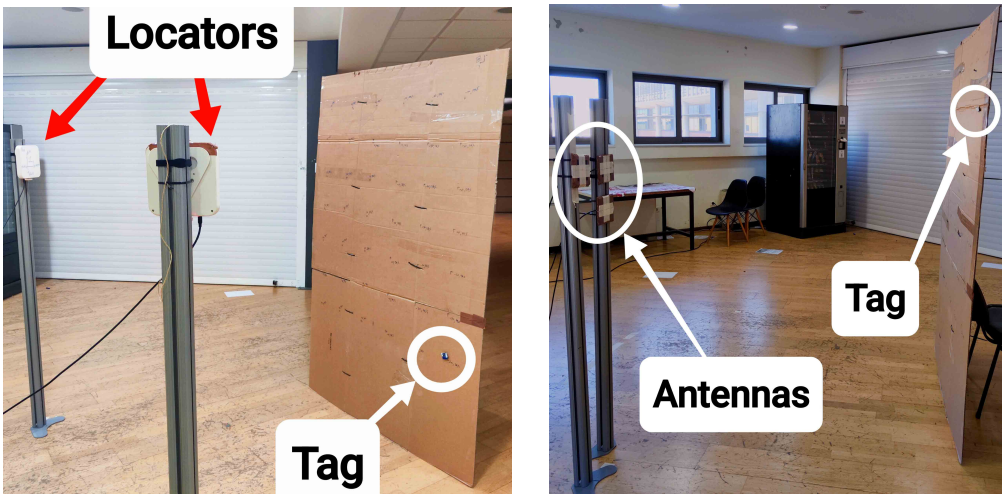
Three experiments were conducted using two Bluetooth locators and an Impinj Speedway R420 RFID reader along with two triads of FlexiRay SF-2110 5 dBi antennas. The two Bluetooth locators and the two triads of FlexiRay antennas, were positioned 2 meters apart, away from the tag wall, where a single tag (BLE or RFID) is located, at distances of 1 meter, 1.3 meters, and 1.5 meters, as depicted in Fig. 4.1. Two Bluetooth locator topologies were employed: one facing the tag wall (BLE Scenario 1 ≡ BLE1), and one rotated 45° towards the wall (BLE Scenario 2 ≡ BLE2). In Fig. 4.2b, only one set of antennas is visible because the reader has 4 antenna ports, requiring the antennas to be relocated to complete the measurements. Again, the comparison will include both raw and buffered measurements for all experiments.



(a) BLE Scenario 1. (b) BLE Scenario 2. (c) RFID.

Figure 4.1: Experimental Setups.

The comparison of the two technologies should be fair and conducted under the same conditions. Therefore, the new hyperbola equation was used to calculate the hyperbola parameters. By having the hyperbola parameters, the algorithms will be the same, ensuring a fair comparison.



(a) Bluetooth Setup. (b) RFID Setup.

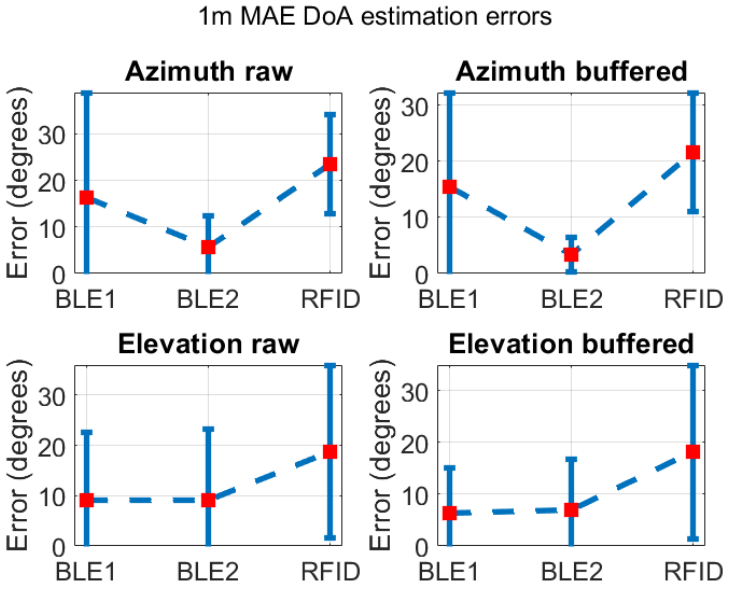
Figure 4.2: Experimental Setups.

4.2.1 DoA Estimation

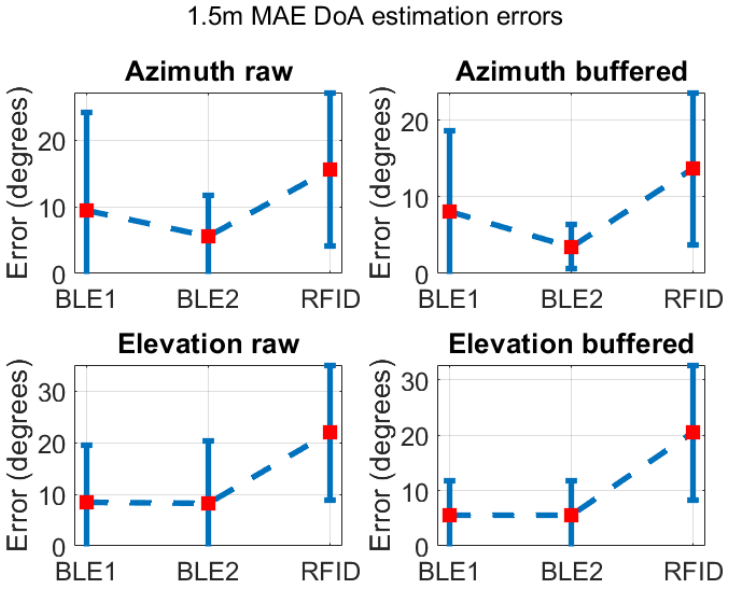
As seen in Fig. 4.3, BLE Scenario 2 outperforms BLE Scenario 1 for azimuth estimation but has very similar errors for elevation estimation. This was expected because the rotated scenario offers higher antenna gain and more accurate parameter estimation, as discussed in Chapter 3. The elevation estimation was anticipated to be similar for the two BLE scenarios because the relative positions of the tags and the locators on the z-axis remained constant. This means the locators were not tilted up or down to focus on the center of the tag wall in Scenario 2. For azimuth estimation, however, the locators were rotated to focus the direction of maximum gain on a point at the center of the tag wall. Additionally, an important observation is that as the locators moved closer to the tag wall, the azimuth estimations for BLE Scenario 1 worsened. This was expected because the tag wall was positioned where the locators' gain was lower, resulting in a noisier received signal.

Additionally, the DoA estimation for RFID is higher compared to the BLE scenarios. This was expected because RFID in static scenarios is not reliable unless measurements from different positions are employed.

The full results for the DoA estimation are summarized in Table 4.1



(a) 1 m Distance



(b) 1.5 m Distance

Figure 4.3: DoA Estimations

Table 4.1: DoA Estimation Error.

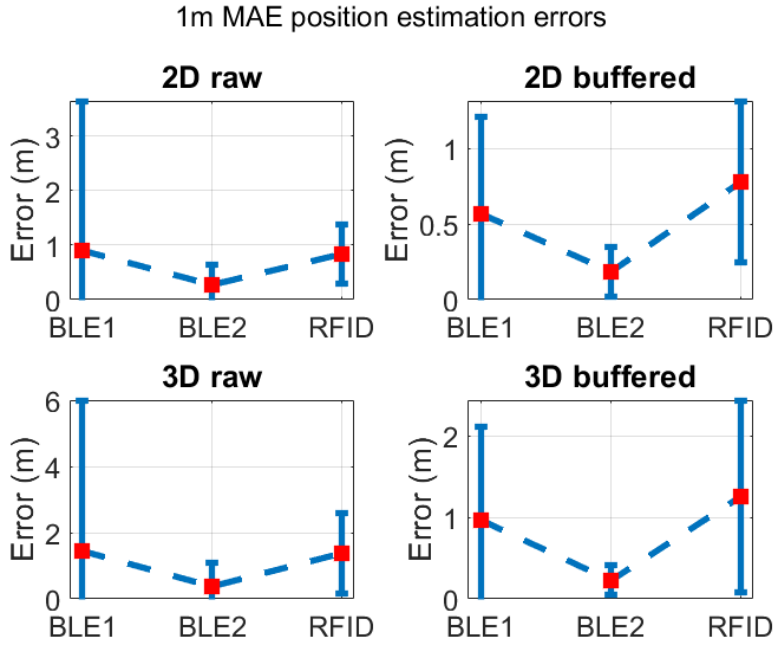
	Azimuth (degrees)			Elevation (degrees)		
$\|x_{\text{ant}} - x_{\text{tag}}\|_2$	1m	1.3m	1.5m	1m	1.3m	1.5m
BLE Scenario 1 (raw)	16.23	11.07	9.45	9.19	7.96	8.43
BLE Scenario 2 (raw)	5.74	6.14	5.61	9.13	8.48	8.22
RFID (raw)	23.57	19.33	15.63	18.83	14.05	22.03
BLE Scenario 1 (buf)	15.47	9.86	8.10	6.30	5.34	5.56
BLE Scenario 2 (buf)	3.32	3.35	3.49	6.97	5.40	5.63
RFID (buf)	21.64	18.66	13.63	18.14	13.35	20.49

4.2.2 Localization Accuracy

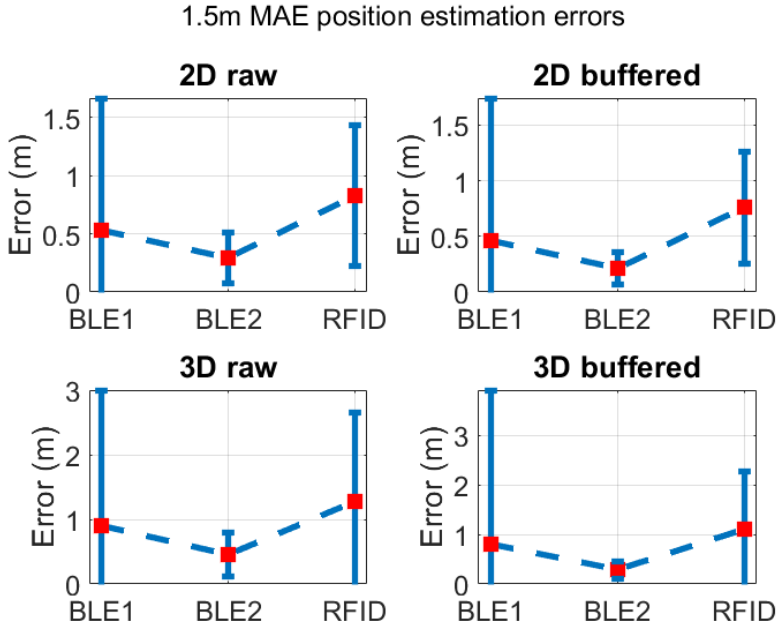
The higher DoA estimation errors of RFID translate into higher position estimation errors. As seen in Fig. 4.4, the error of RFID in raw measurements at 1 meter is close to the error of BLE Scenario 1. However, in buffered measurements, the error of BLE Scenario 1 decreases, while the error of RFID does not. This occurs because the RFID measurements are much closer to each other (more stable) than the BLE's, and buffering cannot significantly improve the estimation accuracy. A noteworthy observation is that the RFID error is more stable, with a smaller standard deviation, compared to BLE. Additionally, it can be observed that BLE Scenario 2 performs better in every case. Another notable point is that the positioning error of BLE Scenario 1 and RFID becomes more comparable as the distance between the tag wall and the antennas/locators decreases. The full results for the position estimation are summarized in Table 4.2

4.2.3 Summary

Multipath is the primary factor contributing to the increased error observed in RFID scenarios. This effect can be mitigated through several methods, including the use of multiple carrier frequencies (i.e., frequency diversity),



(a) 1 m Distance



(b) 1.5 m Distance

Figure 4.4: Position Estimations

Table 4.2: Tag Localization Error.

2D Localization Error						
	MAE (m)			RMSE (m)		
$\|x_{\text{ant}} - x_{\text{tag}}\|_2$	1m	1.3m	1.5m	1m	1.3m	1.5m
BLE Scenario 1 (raw)	0.90	0.57	0.53	2.89	1.48	1.25
BLE Scenario 2 (raw)	0.27	0.30	0.29	0.46	0.40	0.37
RFID (raw)	0.84	0.68	0.83	1.00	0.79	1.03
BLE Scenario 1 (buf)	0.57	0.47	0.46	0.86	0.71	1.36
BLE Scenario 2 (buf)	0.18	0.18	0.21	0.25	0.23	0.26
RFID (buf)	0.78	0.65	0.76	0.95	0.73	0.91
3D Localization Error						
	MAE (m)			RMSE (m)		
$\|x_{\text{ant}} - x_{\text{tag}}\|_2$	1m	1.3m	1.5m	1m	1.3m	1.5m
BLE Scenario 1 (raw)	1.44	0.96	0.90	4.79	2.75	2.29
BLE Scenario 2 (raw)	0.36	0.44	0.46	0.80	0.59	0.57
RFID (raw)	1.38	1.05	1.27	1.83	1.41	1.88
BLE Scenario 1 (buf)	0.97	0.82	0.81	1.50	1.41	3.21
BLE Scenario 2 (buf)	0.23	0.24	0.28	0.29	0.29	0.33
RFID (buf)	1.26	1.10	1.11	1.73	1.69	1.60

obtaining measurements from various locations, or increasing the number of antennas (i.e., spatial diversity). However, in the case of RFID, only one carrier frequency was used, measurements were taken from two points, and the minimum required, by the algorithm, number of antennas was utilized. In contrast, BLE operates differently; it utilizes a bandwidth of 80 MHz divided into 40 channels, providing frequency diversity. This is due to frequency hopping in BLE, resulting in each packet having a different carrier wavelength, allowing the entire available bandwidth to be utilized. Additionally, each BLE locator is equipped with eight antennas, significantly more than RFID.

Chapter 5

Neural Networks for Single Locator

The focus of this chapter remains on solving the localization of a BLE tag with a single locator. The next step involves deploying neural networks to test their ability to address the problem of the initial phase introduced by the tag. Since this specific issue has not been previously tackled, there is no existing architecture to start with. Therefore, it is necessary to determine the optimal architecture, including the number of hidden layers, the number of neurons in each layer, the most effective optimizer, and the best activation function.

Recent work [31] that applies neural networks in Elliptical DoA estimation and localization suggests using the ADAM optimizer and tanh as an activation function. These findings will be utilized to advance the results, but comprehensive research is required to identify the architecture that is most compatible with this problem.

While the research and results of this chapter are not yet complete, they pave the way for future work and show promising results. Before presenting these results, it is necessary to provide some general information about neural networks and the approach taken to develop an architecture capable of solving a problem with no deterministic solution.

5.1 Basic Concepts of Neural Networks

Neural networks are computational models inspired by the structure and function of the human brain. They are renowned for their ability to learn complex and non-linear relationships between input data and desired outputs

by processing vast amounts of data. These networks consist of interconnected nodes, or neurons, organized into layers:

- **Input Layer:** Neurons in the input layer receive the input data, with each neuron corresponding to a feature of the input.
- **Hidden Layers:** These layers perform computations and extract features from the data. Multiple hidden layers can exist, each containing numerous neurons.
- **Output Layer:** The output layer produces the final result of the network, with the number of neurons matching the output classes or the output's dimensional requirements.

Connections between neurons are weighted, with each weight adjusted during training to minimize the error between predicted and actual outputs. Additionally, each neuron has a bias, enabling adjustment of neuron activation thresholds.

Activation functions introduce non-linearity by transforming the output of each neuron, allowing the network to capture complex relationships within the data. The most common activation functions are:

- **Rectified Linear Unit (ReLU)**

$$\text{ReLU}(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases} \tag{5.1}$$

- **Standard Logistic Function (Sigmoid)**

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \tag{5.2}$$

- **Hyperbolic Tangent (tanh)**

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{5.3}$$

5.1.1 Learning Process

The learning process in neural networks involves iteratively updating the weights and biases to minimize the loss function. The loss function measures the difference between the network's predictions and the actual target values. Common loss functions include Mean Absolute Error (MAE) and Mean Squared Error (MSE) for regression tasks and Cross-Entropy Loss for classification tasks. This is typically done through a method called backpropagation, combined with an optimization algorithm like gradient descent.

- **Forward Pass:** In the forward pass, the input data is propagated through the network to produce the output. The loss function then calculates the error between the predicted output and the actual target values.
- **Backpropagation:** Backpropagation is an algorithm used to calculate the gradients of the loss function with respect to each weight and bias in the network. These gradients are then used to update the weights and biases to minimize the loss function.

5.1.2 Optimization Techniques

An optimization algorithm is a method designed to minimize a loss function by updating the weights and biases of the model in response to the loss function output.

Gradient Descent (GD)

Gradient descent is an optimization algorithm used to minimize a function by iteratively moving in the direction of the steepest descent as defined by the negative of the gradient. The parameter update rule for Gradient Descent is given by:

$$w_{t+1} = w_t - \eta \nabla \mathcal{L}_t(w_t), \tag{5.4}$$

where η is the learning rate, and $\nabla \mathcal{L}_t(w_t) = \frac{\partial \mathcal{L}_t}{\partial w_t}$ is the partial derivative of the loss function $\mathcal{L}$ with respect to a weight w at a particular round of

the algorithm, denoted as t. This update rule tells us that we adjust the parameters w_i in the direction opposite to the gradient $\nabla\mathcal{L}_t(w_t)$, scaled by a factor η, to minimize the loss function $\mathcal{L}$.

Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent (SGD) is a variant of gradient descent where the gradient is computed using a single random sample (or a small subset) from the training set rather than the entire dataset. This approach introduces randomness into the parameter updates but can significantly speed up the convergence process, especially for large datasets. The parameter update rule for SGD is given by:

$$w_{t+1} = w_t - \eta\nabla\mathcal{L}_{t,i}(w_t), \tag{5.5}$$

where $\nabla\mathcal{L}_{t,i}(w_t)$ is the gradient of the loss function $\mathcal{L}_t$ with respect to w_t, computed on a single sample i.

Adaptive Moment Estimation (ADAM)

Adaptive Moment Estimation (ADAM) adapts the learning rate for each parameter based on exponentially decaying averages of past gradients and squared gradients. The update rule for ADAM involves computing biased estimates of the first moment (mean) and the second moment (uncentered variance) of the gradients, and then using these estimates to update the parameters:

$$m_{t+1} = \beta_1 m_t + (1-\beta_1)\nabla\mathcal{L}_t(w_t), \tag{5.6}$$

$$v_{t+1} = \beta_2 v_t + (1-\beta_2)(\nabla\mathcal{L}_t(w_t))^2, \tag{5.7}$$

$$\hat{m}_{t+1} = \frac{m_{t+1}}{1-\beta_1^{t+1}}, \tag{5.8}$$

$$\hat{v}_{t+1} = \frac{v_{t+1}}{1-\beta_2^{t+1}}, \tag{5.9}$$

$$w_{t+1} = w_t - \eta\frac{\hat{m}_{t+1}}{\sqrt{\hat{v}_{t+1}}+\epsilon}, \tag{5.10}$$

where β_1 and β_2 are decay rates for the moment estimates, η is the learning rate, ϵ is a small constant to prevent division by zero, m_t and v_t are the first and second moment estimates respectively, and $\nabla\mathcal{L}_t(w_t)$ is the gradient of the loss function $\mathcal{L}_t$ with respect to w_t.

5.2 Types of Neural Networks

Different types of neural networks are designed to handle various types of data and tasks. Here are some common types of neural networks, along with their pros and cons:

Feedforward Neural Networks (FNN)

Feedforward Neural Networks (FNNs) are the simplest type of neural network, where information moves in only one direction—forward from the input nodes, through the hidden nodes (if any), and to the output nodes. There are no cycles or loops in the network.

- **Benefits**
  - Simple to implement and train.
  - Effective for tasks where the relationship between input and output is relatively straightforward.
- **Drawbacks**
  - Cannot handle sequential data well.
  - Limited ability to capture temporal dependencies.

Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNNs) are specialized for processing grid-like data, such as images. They use convolutional layers to automatically and adaptively learn spatial hierarchies of features from input data.

- **Benefits**

  - Excellent for image and video recognition tasks.
  - Reduces the number of parameters in the network by sharing weights.

- **Drawbacks**
  - Requires a large amount of data and computational power.
  - Not suitable for handling sequential data.

Recurrent Neural Networks (RNN)

Recurrent Neural Networks (RNNs) are designed to handle sequential data by maintaining a hidden state that captures information from previous time steps. This makes them ideal for tasks where context or order matters.

- **Benefits**
  - Capable of processing sequences of varying lengths.
  - Effective for tasks involving temporal dependencies, such as time series forecasting and natural language processing.

- **Drawbacks**
  - Can suffer from vanishing and exploding gradient problems, making training difficult.
  - May struggle with long-term dependencies due to limited memory.

Long Short-Term Memory (LSTM) Networks

Long Short-Term Memory (LSTM) networks are a type of RNN designed to overcome the limitations of traditional RNNs by incorporating memory cells that can maintain information over long periods.

- **Benefits**
  - Capable of learning long-term dependencies.

  - Effective for tasks requiring long-range context, such as language modeling and machine translation.

- **Drawbacks**
  - More complex and computationally intensive than traditional RNNs.
  - Requires careful tuning of hyperparameters.

5.3 Construction of Neural Network Model

The construction of the neural network depends on data preparation, where the input is preprocessed to ensure its suitability for the model. Additionally, it is necessary to determine the best architecture to match these inputs so they can produce the desired output.

5.3.1 Data Preparation

To preprocess the measurements of the BLE tag, it is essential to remove the CFO using the method described in Section 1.1.1. Additionally, to ensure the data is suitable for a neural network, either normalization or standardization must be employed.

Normalization typically scales feature values to a range of $[0, 1]$ or $[-1, 1]$, preserving the relationship between the minimum and maximum values of each feature. On the other hand, standardization transforms the data so that the resulting values have the properties of a standard normal distribution, which can ensure faster convergence and greater algorithmic stability.

5.3.2 Architecture of Feedforward Neural Network

The design of the neural network for BLE tag localization encompasses several key components:

Network Design

The neural network consists of an input layer, multiple hidden layers, and an output layer. The specific architecture includes:

- **Input Layer:** The input layer contains neurons corresponding to the preprocessed features of the BLE measurements. These measurements include 8 phase measurements from the 8 different antennas of the locator, 28 phase differences from all possible pairs of antennas, and the frequency of each packet. From these measurements, various input configurations can be defined (e.g., only phases, only phase differences, phases and frequency, differences and frequency, phases and differences, and all combined).
- **Hidden Layers:** The optimal number of hidden layers is not yet definitively determined, as research is ongoing. However, the current optimal number is 4, due to the highly correlated nature of the input data and the complexity introduced by the initial phase of the tag.
- **Output Layer:** The output layer consists of 2 neurons, corresponding to the 2D estimated position of the tag.

Activation Functions, Optimization, and Learning Rate

Activation functions introduce non-linearity into the network, enabling it to learn complex relationships:

- **Hidden Layers:** Recent findings in [31], the most suitable activation function for neural networks in localization tasks is tanh.
- **Output Layer:** A linear activation function is employed in the output layer to provide the estimated 2D position.

The network is trained using the ADAM optimizer with a learning rate of 10^{-3}. This choice balances convergence speed and stability.

Loss Function

The mean absolute error (MAE) is used to measure the error between the predicted and actual positions of the tag. This aligns with the metric used to evaluate the localization accuracy.

5.3.3 Architecture of Recurrent/Long Short-Term Memory Neural Network

The design of the neural network for BLE tag localization encompasses several key components:

Network Design

The neural network consists of an input layer, multiple hidden layers, and an output layer. The specific architecture includes:

- **Input Layer:** The input layer contains neurons corresponding to the preprocessed features of the BLE measurements. These measurements include 8 phase measurements from the 8 different antennas of the locator, 28 phase differences from all possible pairs of antennas, and the frequency of each packet. The data is structured as a sequence of measurements of each position over time.
- **Hidden Layers:** The network incorporates a mix of 2 LSTM layers and 1 fully connected hidden layer to connect to the output layer and produce the required output.
- **Output Layer:** The output layer consists of 2 neurons, corresponding to the 2D estimated position of the tag.

Activation Functions, Optimization, and Learning Rate

Activation functions introduce non-linearity into the network, enabling it to learn complex relationships:

- **Hidden Layers:** Various activation functions were tested in the hidden layer to identify the best fit for the problem.
- **Output Layer:** A linear activation function is employed in the output layer to provide the estimated 2D position.

The network is trained using the ADAM optimizer with a starting learning rate of 10^{-2}, which is then decreased using the `ReduceLROnPlateau` callback in Python with a factor of 0.2 and patience of 5 epochs. This choice balances convergence speed and stability.

Loss Function

The mean absolute error (MAE) is used to measure the error between the predicted and actual positions of the tag. This aligns with the metric used to evaluate localization accuracy. Additionally, both MAE and mean squared error (MSE) are monitored as metrics during training to capture the model's performance comprehensively.

5.4 Experimental Results

For this experiment, around 500 phase measurements were taken from various positions on a 5×4 m^2 grid. To ensure sufficient data, the robot described in Section 1.4, was employed, moving in increments of 10 cm along both x-axis and y-axis. As seen in Fig. 5.1, the data was collected in a large indoor area (without a ceiling) to maximize the grid space and minimize multipath interference.

First, a feedforward neural network (FNN) was deployed to determine the optimal number of hidden layers and the number of neurons for each layer. For each neural network, utilizing various inputs, the number of neurons in each hidden layer was increased to determine the optimal configuration, followed by increasing the number of hidden layers. The number of neurons found to be optimal remained consistent as the layers were increased. For each additional layer, the number of neurons needed to be determined. This procedure aimed to identify the best network for this problem.

(a) Front View of the Setup.

(b) Back View of the Setup.

Figure 5.1: Experimental Results For Neural Networks.

As seen in Fig. 5.2 and Fig. 5.3, the mean absolute error (MAE) of both the testing and validation sets remains approximately the same, at over 80 cm, regardless of the increase in the number of layers and neurons. However, the MAE of the training set drops significantly, indicating that the model is either overfitting to the training data or that the optimal number of layers and neurons required is extremely high.

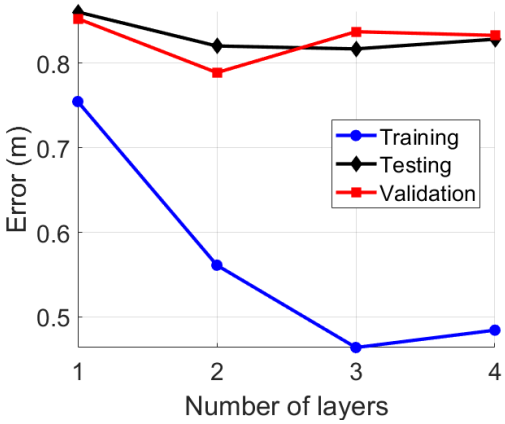
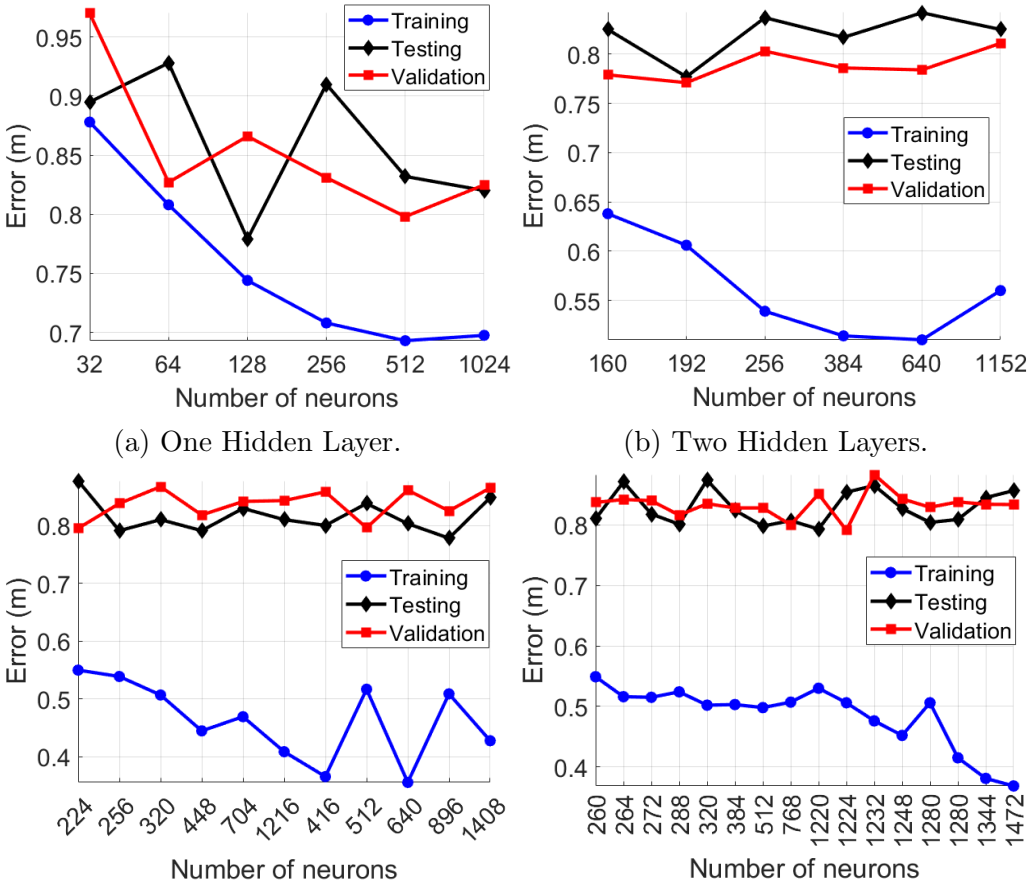


Figure 5.2: Error over the Number of Hidden Layers.



(a) One Hidden Layer.

(b) Two Hidden Layers.

(c) Three Hidden Layers.

(d) Four Hidden Layers.

Figure 5.3: Error over the Total Number of Neurons in the Network.

The second approach involved dividing the 500 measurements of each position into groups of 10 measurements. The data for each position can be represented as a time series because all measurements from each position are collected before moving to a new position. Additionally, the frequency hopping of the BLE tag results in different wavelengths for each BLE packet, making the measurements correlated to a certain degree. Consequently, these groups of 10 can also be described as sequential data. By making this assumption, recurrent neural networks (RNNs) can be used to address the problem. As shown in Table 5.1, the use of RNNs yields significantly better results.

As previously mentioned, the research in this chapter is unfinished. Future work should address the localization problem using more complex architectures to minimize test error and adapt the model for a 3D space.

Activation Function	MAE (m)			MSE (m)		
	Train	Validation	Test	Train	Validation	Test
tanh	0.0772	0.3351	0.3075	0.0127	0.2277	0.1960
relu	0.1205	0.3186	0.3085	0.0283	0.1912	0.1807
sigmoid	0.1014	0.3345	0.3525	0.0217	0.2270	0.2494
softmax	0.8131	0.8596	0.8626	1.0082	1.1089	1.1419
softplus	0.1048	0.3353	0.3615	0.0222	0.2210	0.2527
softsign	0.0872	0.3539	0.3435	0.0163	0.2508	0.2311
selu	0.1113	0.3288	0.3433	0.0253	0.2141	0.2354
elu	0.0667	0.3335	0.3405	0.0089	0.2240	0.2352

Table 5.1: MAE and MSE of RNNs

Chapter 6

Conclusion

6.1 Conclusion

The proposed linear approximations of hyperbolas demonstrate high accuracy for 3D DoA estimation, offering linear complexity and reducing both computational cost and running time. However, this method is highly sensitive to noise from multipath. The random initial phase included by the BLE tag cannot be overcome with deterministic approaches, making it impossible to use the theoretical proof of distance estimation with accurate direction of arrival. Consequently, a single locator can provide accurate results for DoA estimates but cannot deliver reliable localization estimates in either 2D or 3D.

To achieve reliable results in both DoA estimation and localization, the use of multiple locators is essential. Prior Art algorithms require a specific topology, with locators positioned to observe a specific point in the room to improve coverage. The use of hyperbolic DoA estimation can enhance these algorithms, providing excellent results across various topologies and decreasing both DoA and localization errors.

BLE utilizes locators with eight antennas, providing spatial diversity, and an 80 MHz bandwidth divided into 40 channels, offering frequency diversity. This is due to frequency hopping in BLE, which results in each packet having a different carrier wavelength, allowing the entire available bandwidth to be utilized. These features enable BLE to mitigate the multipath effect, the primary factor contributing to increased error in RFID scenarios. In contrast, RFID uses only one carrier frequency, takes measurements from just two points, and employs the minimum required number of antennas. This lack of spatial and frequency diversity in RFID leads to higher errors

compared to BLE.

The need to minimize the number of locators and the ease of BLE equipment in rapidly collecting many measurements motivated the deployment of neural networks (NNs). Viewing BLE measurements as sequential data rather than isolated measurements allows the use of recurrent neural networks (RNNs) as the optimal NN architecture, minimizing the localization error using only a single locator. However, the proposed architecture is not yet optimal, as the number of layers, the number of neurons, and the hyperparameters have not been fully explored.

6.2 Future Work

This work raises questions about the localization accuracy of BLE and RFID in moving scenarios, where RFID has demonstrated very good results. This comparison suggests the potential of merging BLE and RFID into a single tag and receiver to minimize energy consumption and fully solve the accurate localization problem. However, merging BLE and RFID using current algorithms is not feasible for stationary setups. Future research will explore this possibility by developing algorithms for multipath mitigation while considering limitations on the number of antennas and available bandwidth.

Additionally, extensive research into the use of neural networks, particularly Recurrent Neural Networks, should be conducted to find the optimal configuration that will solve the problem using a single locator and be extended in 3D space.

Bibliography

[1] Silicon Labs, "Core Specification 4.0," 2010, latest release Core Specification 5.4.

[2] ——, "AN1297: Custom Direction-Finding Solutions using the Silicon Labs Bluetooth Stack." [Online]. Available: https://www.silabs.com/documents/public/application-notes/an1297-custom-direction-finding-solutions-silicon-labs-bluetooth.pdf

[3] "EPC Radio-Frequency Identity Protocols, Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz, version 1.2.0 EPC Global," 2008.

[4] P. V. Nikitin, R. Martinez, S. Ramamurthy, H. Leland, G. Spiess, and K. V. S. Rao, "Phase based spatial identification of UHF RFID tags," in *Proc. IEEE Int. Conf. on RFID*, Orlando, USA, Apr. 2010, pp. 102–109.

[5] D. Vasicht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single WiFi access point," in *Proc. USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, Santa Clara, USA, Mar. 2016, pp. 165–178.

[6] A. Parr, R. Miesen, and M. Vossiek, "Inverse SAR approach for localization of moving RFID tags," in *Proc. IEEE Int. Conf. on RFID*, Apr. 2013, pp. 104–109.

[7] S. Megalou, A. R. Chatzistefanou, A. Tzitzis, T. V. Yioultsis, and A. G. Dimitriou, "Passive uhf-rfid hyperbolic positioning of moving tags by exploiting neural networks," *IEEE RFID J.*, vol. 6, pp. 402–412, Mar. 2022.

[8] R. Miesen, F. Kirsch, and M. Vossiek, "Holographic localization of passive UHF RFID transponders," in *Proc. IEEE Int. Conf. on RFID*, Orlando, USA, Apr. 2011, pp. 32–37.

[9] A. Buffi, P. Nepa, and F. Lombardini, "A phase-based technique for localization of UHF-RFID tags moving on a conveyor belt: Performance analysis and test-case measurements," *IEEE Sensors J.*, vol. 15, no. 1, pp. 387–396, Jan. 2015.

[10] P. Tripicchio, M. Unetti, S. D'Avella, A. Buffi, A. Motroni, F. Bernardini, and P. Nepa, "A Synthetic Aperture UHF RFID Localization Method by Phase Unwrapping and Hyperbolic Intersection," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 2, pp. 933–945, Apr. 2022.

[11] Y. Ma, Y. Fu, X. Liang, H. Liu, and K. Chen, "An Efficient Method for Hyperbolic-Based Localization in SAR RFID Systems," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, Mar. 2022.

[12] F. Martinelli, "Simultaneous localization and mapping using the phase of passive UHF-RFID signals," *Journal of Intelligent & Robotic Systems (JINT)*, vol. 94, no. 3-4, pp. 711–725, Jul. 2019.

[13] C. S. Mouhammad, A. Allam, M. Abdel-Raouf, E. Shenouda, and M. Elsabrouty, "BLE indoor localization based on improved RSSI and trilateration," in *7th Int. Japan-Africa Conf. on Electronics, Communications, and Computations, (JAC-ECC)*, 2019, pp. 17–21.

[14] R. Ayyalasomayajula, D. Vasisht, and D. Bharadia, "BLoc: CSI-based accurate localization for BLE tags," in *Proc. of the 14th Int. Conf. on Emerging Networking EXperiments and Technologies*, Heraklion, Greece, Dec. 2018, pp. 126–138.

[15] X. Qiu, B. Wang, J. Wang, and Y. Shen, "AOA-based BLE localization with carrier frequency offset mitigation," in *IEEE Int. Conf. on Communications Workshops (ICC Workshops)*, 2020, pp. 1–5.

[16] Q. Liu, W. IJntema, A. Drif, P. Pawełczak, and M. Zuniga, "BEH: Indoor batteryless BLE beacons using RF energy harvesting for internet of things," Nov. 2019.

[17] E. DiGiampaolo and F. Martinelli, "Mobile robot localization using the phase of passive uhf rfid signals," *IEEE Trans. Ind. Electron.*, vol. 61, no. 1, pp. 365–376, Jan. 2014.

[18] ——, "A robotic system for localization of passive UHF-RFID tagged objects on shelves," *IEEE Sensors J.*, vol. 18, no. 20, pp. 8558–8568, Oct. 2018.

[19] E. Giannelos, E. Andrianakis, K. Skyvalakis, A. G. Dimitriou, and A. Bletsas, "Robust RFID localization in multipath with phase-based particle filtering and a mobile robot," *IEEE RFID J.*, vol. 5, no. 3, pp. 302–310, Jun. 2021.

[20] S. Azzouzi, M. Cremer, U. Dettmar, R. Kronberger, and T. Knie, "New measurement results for the localization of UHF RFID transponders using an Angle of Arrival (AoA) approach," in *Proc. IEEE Int. Conf. on RFID*, 2011, pp. 91–97.

[21] K. Skyvalakis, E. Giannelos, E. Andrianakis, and A. Bletsas, "Elliptical DoA estimation & localization," *IEEE RFID J.*, vol. 6, pp. 394–401, Jan. 2022.

[22] D. F. Llorca, R. Quintero, I. Parra, and M. A. Sotelo, *Recognizing individuals in groups in outdoor environments combining stereo vision, RFID and BLE.* New York, NY: Springer New York, 2017, p. 769–779. [Online]. Available: https://doi.org/10.1007/s10586-017-0764-0

[23] T. Liu, L. Yang, Q. Lin, Y. Guo, and Y. Liu, "Anchor-free backscatter positioning for RFID tags with high accuracy," in *Proc. IEEE Int. Conf. on Computer Communications (Infocom)*, Toronto, Canada, Apr. 2014, pp. 379–387.

[24] E. Giannelos, "Intelligent wireless networks and robots for low-cost battery-less sensing and localization," Ph.D. dissertation, School of ECE, Technical University of Crete, Chania, Greece, 2023, advisor: A. Bletsas.

[25] Silicon Labs, "UG103.18: Bluetooth® Direction Finding Fundamentals." [Online]. Available: https://www.silabs.com/documents/public/user-guides/ug103-18-bluetooth-direction-finding-fundamentals.pdf

[26] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[27] M. Pelka, C. Bollmeyer, and H. Hellbrück, "Accurate radio distance estimation by phase measurements with multiple frequencies," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2014.

[28] J. Traa, "Least-squares intersection of lines," University of Illinois Urbana-Champaign (UIUC), 2013.

[29] G. Andreadis, E. Giannelos, P. N. Alevizos, and A. Bletsas, "Hyperbolic DoA Estimation and Static Localization: Comparative Study of Bluetooth and RFID," in *Proc. IEEE Int. Conf. on RFID*, Mar. 2024.

[30] S. Megalou, A. Bletsas, T. Yioultsis, and A. G. Dimitriou, "Power and phase variation of backscattered rfid signal with respect to the incident power at the tag," in *Proc. IEEE Int. Conf. on RFID-Technologies and Applications (RFID-TA)*, Delhi, India, Oct. 2021, pp. 36–39.

[31] G. Papadopoulos, "RFID Localization with Multistatic Interrogation and Neural Networks," Diploma thesis available in: https://dias.library.tuc.gr/view/99087, School of ECE, Technical University of Crete, Chania, Greece, Mar. 2024, Supervisor A. Bletsas.