Indoor localization using Smartphone-based IEEE 802.11mc WiFi Round Trip Time (RTT) and Android Aware technology

by

Abhishek Arunkumar Kulkarni

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
August 5, 2023

Approved by

Alvin S. Lim, Chair, Professor of Computer Science and Software Engineering
Shiwen Mao, Professor of Electrical and Computer Engineering
Dean Hendrix, Professor of Computer Science and Software Engineering
Cheryl Seals, Professor of Computer Science and Software Engineering

Abstract

Self-localization of mobile nodes is a critical challenge in enabling a wide range of mobile applications that rely on accurate position information. Traditionally, obtaining precise location data has required specialized hardware or dedicated infrastructure, such as GPS, UWB, ultrasounds transceivers, GSM, or WLAN. However, we propose an alternative approach that eliminates the need for such additional components and infrastructure. Our approach revolves around the concept of cooperative and opportunistic data exchanges among mobile nodes, which can significantly enhance and refine the localization process. Consider a scenario where a target node lacks GPS or any position information. By leveraging communication with multiple nearby mobile peer nodes that possess some positioning capabilities, we can achieve indoor localization without the need for specialized hardware or infrastructure support. Specifically, our smartphone-based technique harnesses the power of IEEE 802.11mc WiFi-based fine time measurement (FTM) capabilities for indoor navigation and tracking. This cutting-edge technology utilizes smartphones equipped with WiFi chipsets that support round trip time (RTT) measurements. By leveraging FTM and its hardware-level timestamping of send and receive events, we can accurately estimate the RTT and, consequently, infer precise distance measurements.

An advantageous aspect of our technique is its infrastructure independence. It operates autonomously, leveraging the networking capabilities of smartphone devices and facilitating communication through the formation of device clusters. This is made possible by leveraging the Android Aware technology, allowing for seamless networking and collaboration among smartphones within the same vicinity. By employing this smartphone-based approach, we open up new possibilities for indoor localization and tracking. The technique enables GPS-like operation indoors, overcoming the limitations of traditional positioning methods. It leverages the ubiquity of smartphones and

their embedded WiFi capabilities to achieve accurate and reliable indoor localization without the need for additional hardware or infrastructure support.

In summary, our approach represents the next generation of indoor navigation and tracking. By leveraging the IEEE 802.11mc WiFi-based FTM capabilities of smartphones, we unlock the potential for precise indoor localization. The technique capitalizes on cooperative and opportunistic data exchanges among mobile nodes, enabling accurate position estimation without relying on specialized hardware or dedicated infrastructure. With our infrastructure-independent solution, we pave the way for a new era of indoor positioning technology.

Acknowledgments

I would like to thank my advisor Dr. Alvin Lim for providing me with a great opportunity to join his research group at Auburn University. It would be very remiss of me not to acknowledge the amount of time and effort he has spent on me to shape my professional career. His constant support and guidance have made this dissertation possible. I am immensely grateful to Dr. Xiao Qin and Dr. Dean Hendrix for providing me with an assistantship when I joined Auburn. Without him, this journey wouldn't have begun. I would like to thank my committee members Dr. Shiwen Mao, Dr. Dean Hendrix, and Dr. Cheryl Seals for supporting my research and serving on my defense. I would like to acknowledge Dr. Yin Sun for serving as the graduate school representative for my dissertation.

The friendships and memories from my time in Auburn are something that I will cherish and remember fondly. My friends Bhargav, John, Junaid, Arjun, Akshay, Yufei, and Yue have been a positive influence and helped me in more ways than I can imagine.

I would like to thank my parents- Dr. Arunkumar Kulkarni and Arati Kulkarni, for all their love and unwavering belief in me. Everything I possess and value in my life stems from their sacrifice and relentless support. I am thankful to my fiancee, Sahana, for always being there for me. Her continued motivation has inspired me to be a better person. I dedicate this dissertation to my parents and fiance.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the year 2000, a significant milestone occurred when the Defense Department decided to remove the intentional degradation of GPS accuracy, resulting in a tenfold increase in accuracy compared to the previous version available to civilians [1]. This pivotal event spurred the development of devices aimed at monitoring and tracking various objects. Over the following decade, GPS devices became widely adopted, and as their usage increased, so did the demand for improved accuracy. The latest generation of GPS devices offers significantly enhanced accuracy compared to their predecessors. However, despite the advancements in GPS technology, one significant limitation remains: satellite signals cannot penetrate through buildings, dense forests, or urban canyons. As a result, GPS functionality is largely restricted to outdoor environments with a clear line of sight to satellites. Recognizing this limitation, extensive research has been conducted to address the challenge and develop a new technology capable of providing accurate monitoring and tracking in indoor environments [2, 3, 4, 5]. Unfortunately, during the early stages of this research, the lack of technological support hindered the achievement of acceptable accuracy levels.

In recent years, there has been a resurgence in the importance of monitoring and tracking devices, primarily driven by technological advancements in the industry. With the support of improved technology, the feasibility of achieving accurate indoor monitoring and tracking has become more attainable. This renewed focus on indoor localization has been further fueled by the growing popularity of indoor mobile robots and the increasing demand for AI communication services [6]. The recognition of the limitations of GPS in indoor environments, combined with the advancements in technology and the rising demand for indoor monitoring and tracking, has accelerated the exploration of innovative solutions. Researchers and industry professionals have directed their efforts toward developing robust indoor localization technologies that can overcome the challenges posed by signal attenuation and obstructions. By leveraging alternative approaches, such as WiFi-based techniques, ultra-wideband (UWB) technology, or Bluetooth Low Energy (BLE) beacons, the aim is to achieve accurate and reliable indoor positioning.

Accurate indoor monitoring and tracking have become increasingly significant across various domains, with far-reaching implications for asset management, indoor navigation, facility security, and smart building automation. The ability to precisely locate and track objects, assets, and individuals within indoor environments offers a wealth of benefits and opportunities. In the realm of asset management, organizations can optimize their operations by effectively monitoring the movement and utilization of valuable resources within indoor spaces. This includes tracking equipment, inventory, and supplies, enabling efficient allocation, maintenance, and retrieval of assets. Accurate indoor monitoring also contributes to improved inventory management, minimizing losses and enhancing overall operational efficiency. Indoor navigation is another area greatly influenced by accurate monitoring and tracking technology. Traditional navigation systems, such as GPS, are not reliable within buildings, and this limitation hampers wayfinding and navigation in complex indoor spaces. However, with precise indoor localization capabilities, individuals can seamlessly navigate within large buildings, airports, shopping malls, hospitals, and other indoor environments. This enhances visitor experiences, facilitates efficient movement, and opens up opportunities for location-based services and targeted advertising.

Facility security is yet another domain that benefits from accurate indoor monitoring and tracking. By precisely tracking the movement of people and assets within a building, security teams can enhance surveillance, detect anomalies, and respond promptly to potential threats. Real-time tracking provides valuable insights into security breaches, enabling swift interventions and preventive measures. Moreover, the integration of accurate indoor monitoring and tracking technologies with smart building automation systems holds great potential. By continuously monitoring the location and behavior of occupants, energy usage patterns, and environmental conditions, buildings can dynamically adapt and optimize their operations. This results in improved energy efficiency, occupant comfort, and overall sustainability.

A wireless ad-hoc network is a unique type of computer network that operates without relying on any pre-existing infrastructure or physical wires. Unlike traditional computer networks that require centralized routers or access points, ad-hoc networks allow individual nodes to communicate and establish a network on their own. In an ad-hoc network, each node is responsible for transmitting and forwarding data to other nodes within the network. One of the key advantages of ad-hoc networks is their flexibility and simplicity in maintenance. Since ad-hoc networks do not depend on fixed infrastructure, they can be quickly deployed and reconfigured as needed. This makes them particularly suitable for dynamic or temporary environments where the availability of infrastructure may be limited or impractical.

While ad-hoc networks can utilize a flooding protocol, where data is forwarded to all neighboring nodes, there are more efficient routing protocols available. These protocols, such as proactive routing, location-based routing, or hybrid routing, optimize the transmission of data by selectively choosing the most suitable paths within the network. By employing these routing protocols, ad-hoc networks can achieve better efficiency and scalability compared to simple flooding approaches. The maintenance of an ad-hoc network is fairly simple and flexible compared to other networks which involve infrastructures [7]. The choice to use an ad-hoc network as our mobile networking system is driven by its flexibility and adaptability. With an ad-hoc network, our mobile nodes can communicate and establish connections with one another without relying on fixed infrastructure or central control. This allows for greater mobility and independence, making it well-suited for scenarios where traditional network infrastructures may not be available or feasible.

In the context of mobile and ad-hoc networks, having knowledge of the position and trajectory of individual nodes is crucial for various purposes. This information can be leveraged to optimize communication protocols, plan efficient paths, and design cooperative tasks among the nodes. The accuracy of localization estimation is highly dependent on the specific environment and the technology employed by the devices to determine their positions.

Some localization techniques rely on inexpensive and widely available technologies that utilize Received Signal Strength Indicator (RSSI) measurements. However, such approaches generally yield poor localization performance, as demonstrated in studies [8]. On the other hand, more expensive hardware that compares the Time-of-Arrival (TOA) of radio signals, as discussed in research [9], can provide better accuracy in localization. However, the adoption of specialized localization hardware can significantly increase the cost of mobile devices, making it less feasible in certain scenarios. As a result, there is a need for localization methods that strike a balance between cost-effectiveness and accuracy. Utilizing low-cost Commercial off-the-shelf (COTS) hardware that can provide comparable accuracy to expensive alternatives becomes a viable alternative. By leveraging existing hardware components that are readily available in the market, the cost of mobile devices can be kept reasonable while still achieving satisfactory localization accuracy. The accuracy of localization estimation depends on the environment and the technology employed. While expensive hardware solutions can provide better accuracy, they may not be cost-effective for widespread adoption. Hence, alternative approaches utilizing low-cost Commercial off-the-shelf hardware with comparable accuracy become more viable options in practice.

Accurate localization or tracking of wireless devices has become a critical requirement in various emerging location-aware systems. These systems find applications in diverse fields such as search and rescue operations, medical care, intelligent transportation, location-based billing, security, home automation, industrial monitoring and control, location-assisted gaming, and social networking. The demand for accurate localization spans across multiple domains, highlighting the need for robust and reliable solutions. While satellite-based navigation systems like GPS have achieved widespread adoption and success in open sky scenarios, localization in challenging environments such as indoors or urban areas remains a persistent challenge. These scenarios pose unique obstacles that hinder the effectiveness of traditional satellite-based approaches. Consequently, there is a growing demand for a new technology that leverages wireless networks to provide self-localization capabilities, filling the gap in harsh environments where GPS signals may not be accessible.

To address these contemporary needs, we have developed a novel method of localization based on opportunistic data exchanges. Our approach takes advantage of the cooperative nature of wireless networks, allowing mobile devices to exchange data and collaborate in self-localization. By leveraging the available network infrastructure and the information shared between devices, our method offers a promising solution to achieve accurate localization even in challenging environments. The current trend in the field of localization is the integration of heterogeneous technologies to ensure global coverage and high accuracy across various scenarios. This integration aims to create a seamless localization system that is accessible anywhere and anytime. By combining different technologies and leveraging the power of wireless networks, we can overcome the limitations of individual approaches and achieve reliable and precise localization in a wide range of situations.

Localization and location-based services have become a worldwide technological need with the increased use of smartphone devices equipped with GPS and inertial sensors. Although GPS provides continuous location information with reasonable location accuracy for outdoor environments, it fails to provide the same for indoor environments. In the past few years, there has been an increased demand for indoor localization techniques to fulfill the gap created by the GPS technique. Many techniques have been developed to fulfill the need for an indoor localization technique in the past decade but most of them are infrastructure dependent and use sophisticated hardware units.

Our indoor localization technique utilizes standard Android smartphones, which are widely used by millions of people, making them accessible and cost-effective. Unlike other localization methods, our approach does not rely on specialized hardware or require additional infrastructure installations. Instead, we leverage the capabilities of the IEEE $802.11mc$ WiFi protocol, which is already present in these smartphones. The core principle of our technique is the use of Round-Trip-Time (RTT) measurements to estimate the distance between two or more devices. By accurately measuring the time it takes for a signal to travel from a reference node to the target smartphone and back, we can calculate the distance between them. These distance measurements, combined with the known position coordinates of the reference node, serve as the input for our sophisticated algorithms.

By leveraging the existing WiFi capabilities of standard smartphones and utilizing sophisticated algorithms, our technique offers a practical and accurate solution for indoor localization. It provides an accessible and cost-effective alternative to specialized hardware-based approaches. With its reasonable accuracy, our smartphone-based IEEE $802.11mc$ fine time measurement (FTM) technique opens up possibilities for a wide range of applications that rely on indoor localization, including navigation, asset tracking, and location-based services.

Chapter 2

Motivations and Applications

In this chapter, we delve into the motivations behind the development of our localization technology and explore its diverse range of applications. By understanding the driving forces and potential use cases, we gain valuable insights into the significance and practical implications of our technology.

2.1 Motivation

The motivation for research in indoor navigation and tracking systems stems from the need to provide accurate and reliable location information for a wide range of applications in indoor environments. While outdoor navigation systems using GPS and other satellite-based technologies have become commonplace, similar solutions for indoor environments are much more challenging to implement. This is due to the fact that GPS signals are typically too weak to penetrate building walls and roofs, resulting in poor signal quality or no signal at all in indoor environments. Although indoor tracking has been an important research area since the early 2000s, it has been difficult to develop a system with the required accuracy. Indoor tracking using Global Positioning System (GPS) does not work as the GPS satellite signals cannot reach the indoor environment. As a result, the accuracy of the position is very poor. Also, other research techniques like Ultra-wideband (UWB) produce acceptable accuracy but it is very expensive to deploy these devices. Other techniques like Received Signal Strength indicator (RSSI), although very easy to develop and maintain produces poor accuracy and are unreliable in most cases. This motivated us to develop a new technique to solve the indoor tracking system which can provide better accuracy and cost-effectiveness to deploy on a large scale.

Indoor navigation and tracking systems can be used in a variety of settings, such as hospitals, airports, museums, shopping malls, and factories, to help people find their way, track assets, and optimize workflows. For example, in hospitals, indoor navigation systems can help patients, visitors, and staff members navigate complex buildings, locate specific departments, and find specific medical equipment. In factories, indoor tracking systems can help optimize supply chain management, reduce inventory loss, and improve worker safety by tracking the location of goods and equipment in real-time. In addition to these practical applications, research in indoor navigation and tracking systems is also motivated by the potential for new location-based services and technologies, such as augmented reality, indoor mapping, and indoor positioning in autonomous robots. As such, there is a growing interest in developing accurate, scalable, and low-cost indoor navigation and tracking systems that can work in a variety of environments and under different conditions.

Another key motivation for research in indoor navigation and tracking systems is the increasing demand for location-based services and the growth of the Internet of Things (IoT) market. As more and more devices become connected to the internet and to each other, the ability to accurately track and locate these devices in indoor environments becomes critical for a wide range of applications, including home automation, smart buildings, and asset tracking.

Moreover, indoor navigation and tracking systems can also help improve safety and security in indoor environments. For example, in emergency situations, indoor navigation systems can help guide people to safety and enable emergency responders to quickly locate and assist those in need. In addition, indoor tracking systems can be used to monitor the movement of people and assets in real-time, which can be useful for preventing theft, identifying safety hazards, and optimizing building layouts. The development of new technologies, such as Bluetooth Low Energy (BLE), WiFi RTT, ultra-wideband (UWB), and visual-based tracking, has also fueled research in indoor navigation and tracking systems. These technologies have enabled new methods for tracking the location of objects and people in indoor environments with greater accuracy and precision, and have opened up new possibilities for location-based services and applications.

Overall, the motivation for research in indoor navigation and tracking systems are driven by a wide range of factors, including practical applications, the growth of the IoT market, the demand for location-based services, and the development of new technologies. As such, research in this area is likely to continue to grow in importance and impact in the years to come.

2.2 Applications

Indoor tracking systems have the potential to support a variety of applications, including tracking objects and personnel in hospitals, malls, university campuses, and automobile plants. They can be especially useful in extreme conditions where firefighters need to navigate in low-visibility environments. Additionally, indoor tracking systems can provide navigation and directions in complex buildings, helping individuals avoid the frustration of getting lost. Overall, the versatility of indoor tracking systems makes them useful for a wide range of applications in different settings. We will discuss some of these applications in the following sections.

2.2.1 Hospitals, Industrial Warehouses, and University Campuses

Indoor localization application in Hospitals

Indoor localization technology can significantly improve the efficiency and quality of patient care in hospitals. By using location-based services, hospitals can track the movement of patients, medical equipment, and staff within the facility. This can help hospitals optimize resource allocation, improve patient flow, and reduce waiting times. One of the most significant applications of indoor localization technology in hospitals is patient tracking. By equipping patients with location-tracking devices, hospitals can monitor their movement and provide personalized care. This can help medical staff quickly locate patients, administer medication, and provide urgent care in a timely manner.

Another application of indoor localization technology in hospitals is asset tracking. Hospitals can track the movement of medical equipment and supplies, ensuring that they are available when needed. This can help reduce equipment loss and theft, improve inventory management, and save costs by eliminating the need for excess equipment.

Indoor localization technology can also be used to improve hospital security by monitoring access to restricted areas and detecting any unauthorized movement. This can help ensure that only authorized personnel have access to sensitive patient data and medical supplies. Moreover, indoor localization technology can be used to provide real-time navigation and wayfinding services to patients and visitors. This can help them navigate the complex hospital environment and reduce the likelihood of getting lost or delayed.

Indoor localization technology can also be used to improve patient safety in hospitals. By tracking patient movement and interactions with staff and medical equipment, hospitals can identify potential safety issues and take corrective action in real-time. For example, if a patient with a high risk of falls is found to be moving towards a high-risk area, such as a staircase or slippery floor, hospital staff can be alerted to intervene and prevent a fall.

Indoor localization technology can also be used to automate patient check-in and check-out processes, reducing wait times and improving the patient experience. By using location-based services, patients can be automatically checked in when they arrive at the hospital, and their location can be tracked throughout their stay. This can help hospitals optimize patient flow and reduce wait times, which can be a significant source of frustration for patients and their families.

Another potential application of indoor localization technology in hospitals is infection control. By tracking the movement of patients, staff, and medical equipment, hospitals can identify potential sources of infection and take corrective action to prevent the spread of disease. This can help reduce the risk of hospital-acquired infections, which can be a significant source of morbidity and mortality in healthcare settings.

Finally, indoor localization technology can be used to improve hospital operations by providing real-time data on patient flow, staffing levels, and equipment utilization. This can help hospital administrators optimize resource allocation and improve operational efficiency, leading to better patient outcomes and reduced costs.

Overall, the applications of indoor localization technology in hospitals are diverse and far-reaching. By improving patient safety, automating patient check-in processes, enhancing infection control, and optimizing hospital operations, indoor localization technology is becoming an essential tool for hospitals looking to provide better care to their patients.

Indoor localization application in University Campus

Indoor localization technology can provide several benefits for universities and colleges. One of the primary applications is campus navigation. Large university campuses can be complex and difficult to navigate, especially for new students or visitors. By providing real-time location-based services, universities can guide students and visitors to their destinations with ease. This can improve the overall experience of students and visitors and reduce the risk of getting lost.

Indoor localization technology can also be used to enhance campus security. By tracking the movement of students and staff, universities can quickly identify potential security threats and take appropriate action to mitigate them. For example, if a student is found to be in an unauthorized area or outside of the designated campus boundaries, campus security can be alerted to intervene. Another application of indoor localization technology in university campuses is space utilization. By tracking the use of classrooms, lecture halls, and other facilities, universities can optimize resource allocation and improve operational efficiency. This can help universities reduce costs and improve the overall quality of education.

Indoor localization can be used to enhance student engagement on campus. By leveraging indoor localization technology, universities can provide personalized and interactive experiences for students. For example, universities can use location-based notifications to provide students with information about upcoming events or course-related announcements. Indoor localization technology can help universities manage their facilities more efficiently. By tracking the use of different rooms and facilities, universities can identify opportunities to optimize the use of space and reduce energy costs. In addition to enhancing security, indoor localization technology can be used to monitor the health and safety of students and staff. For example, universities can use location data to track the movement of students during emergencies and ensure that everyone is accounted for. Universities can also use indoor localization technology to improve their marketing efforts. By tracking the movement of students and visitors, universities can identify popular areas and tailor their marketing messages accordingly.

Finally, indoor localization technology can be used to support research and development activities on university campuses. By tracking the movement of researchers and equipment, universities can optimize the use of lab space and improve the efficiency of research operations. This can help researchers achieve their goals faster and more efficiently, leading to better research outcomes.

In summary, indoor localization technology has a wide range of applications on university campuses, from improving campus navigation to enhancing security and optimizing resource allocation. As the technology continues to advance, we can expect to see even more innovative use cases emerge in the future.

Indoor localization application in large industrial warehouse

Accurate inventory management is critical for warehouse operations. By using indoor localization technology, warehouses can track the location of goods and products within the facility in real-time. This helps in keeping track of inventory levels, preventing stock-outs, and reducing waste. The technology can also be used to alert warehouse personnel when inventory levels fall below a certain threshold, ensuring that reordering is done in a timely manner. Indoor localization technology can help identify bottlenecks in the workflow and optimize the routing of goods and materials. For example, by tracking the movement of employees and equipment within the warehouse, the technology can identify areas where congestion is happening and re-route traffic to avoid it. This results in a more efficient workflow and faster delivery times.

In addition to tracking inventory, indoor localization technology can be used to track the location of assets such as forklifts, carts, and other equipment. This can help improve efficiency by reducing the time and effort required to locate these assets. The technology can also be used to monitor equipment usage and track maintenance needs, ensuring that equipment is properly maintained and in good working condition. Indoor localization technology can improve safety and security in the warehouse by monitoring employee movement and detecting any unusual or unsafe behavior. The technology can also be used to identify and locate potential hazards in the warehouse, such as spills or dangerous equipment. Additionally, the technology can be used to track the movement of unauthorized personnel within the facility, alerting security personnel to any potential security breaches. Indoor localization technology can help optimize energy use by turning off lights and HVAC systems in areas that are not in use. By tracking the movement of people and equipment within the warehouse, the technology can determine which areas are not being used and adjust energy usage accordingly. This results in cost savings and a more sustainable warehouse operation.

Overall, indoor localization technology has numerous applications in industrial warehouses, from improving inventory management to enhancing safety and security. By leveraging this technology, warehouses can become more efficient, productive, and safe, leading to better business outcomes.

2.2.2 Airports

Indoor localization has several important applications at airports. One of the most important is navigation and wayfinding. Airports can be complex and confusing environments, with multiple levels, terminals, and concourses. Indoor localization can be used to help passengers navigate their way through the airport, providing turn-by-turn directions to their gate, restaurant, or other destination. This can be particularly helpful for passengers who are unfamiliar with the airport or who have visual impairments. Indoor localization can also provide real-time updates on flight information, gate changes, and other important information, which can help passengers stay informed and reduce stress.

Indoor localization can also be used to track the location of airport staff and equipment, which can help improve efficiency and reduce delays. For example, airport operators can use indoor localization to track the location of baggage carts and other equipment, to ensure that they are in the right place at the right time. Indoor localization can also be used to track the location of airport staff, such as cleaning crews, maintenance workers, and security personnel, which can help improve coordination and communication.

Additionally, indoor localization can be used to enhance security at airports, by monitoring the location of passengers and staff and detecting unauthorized personnel in restricted areas. For example, airport operators can use indoor localization to track the movement of passengers and staff through security checkpoints, to ensure that everyone is properly screened before entering restricted areas. Indoor localization can also be used to detect intruders or other unauthorized personnel in restricted areas and to alert security personnel to potential threats.

Indoor localization can be used in airport retail settings to track customer movements and behavior and to provide targeted advertising or promotions based on the customer's location within the airport. For example, airport retailers can use indoor localization to track the location of passengers and offer personalized discounts or promotions based on their travel itinerary. Indoor localization can also be used to optimize store layouts and product placements, based on customer behavior and traffic patterns.

Overall, indoor localization has the potential to improve the airport experience for passengers, staff, and airport operators alike, by providing accurate and real-time information, improving efficiency and coordination, enhancing security, and offering personalized services and promotions. Our indoor navigation system can provide simple directions for navigating inside these buildings for each individual depending on their flight gate. This reduces the anxiety and pain of being lost inside complex environments such as an airport.

2.2.3 Indoor Tracking of Firefighters and First Responders

Firefighters work in one of the most dangerous environments and put their lives on the line to save others. A fire scene usually consists of a lot of chaos and it could be very easy for a firefighter to get indulged inside the building in fighting a fire or saving life and lose track of his teammates. It could be very difficult to navigate inside the building as the vision usually will be almost nil along with the chaos going on the scene. So, firefighters get lost inside the building more than we can imagine. Our technology can provide a solution to this critical problem by providing a device that is capable of tracking all the firefighters on the scene. This way the Chief-in-Office on the scene can keep track of all the firefighters and if anyone of them loses track of their location, it would be easy to track the location and rescue them in time to save their lives.

Indoor localization has become an important tool for firefighters who often face the challenge of navigating through smoke-filled buildings and finding their way to trapped victims. With the help of indoor localization systems, firefighters can determine their precise location within a building and navigate through it more efficiently. These systems can provide real-time tracking of firefighters, enabling incident commanders to keep track of their location and movements and ensure their safety. The system can also provide important data such as the location of hazardous materials, fire exits, and other critical information. Additionally, indoor localization can be used to locate and rescue trapped victims, as well as to improve the overall response time and effectiveness of firefighting operations. Overall, indoor localization has the potential to greatly enhance the safety and effectiveness of firefighting operations, and save lives in the process.

2.2.4 Robot localization

Indoor localization is a critical component in robot localization, which is the process of determining the position and orientation of a robot in an indoor environment. Robot localization is essential for many robotic applications, including autonomous navigation, mapping, and exploration. In these applications, the robot needs to know its location and orientation in order to move safely and effectively in the environment. Pathfinding and mapping are actually an extension of localization and are very essential for robot movement to any destination. Robot localization can be categorized into three groups Global navigation, Local navigation, and Personal Navigation [10]. Among these three robot localization categories, our indoor tracking system can be useful in Global and Local Navigation as these two navigation types require human interaction or with other robots. Our technique is based on opportunistic communication between the nodes so it cannot be applied to personal navigation as in this type the robot is alone and does not interact with anything to navigate around the referenced area.

Robot localization has numerous applications, including search and rescue missions, industrial automation, and delivery services. In search and rescue missions, robots can be used to locate and rescue people in dangerous or inaccessible locations. In industrial automation, robots can be used to transport materials or perform complex tasks in factories. In delivery services, robots can be used to autonomously deliver packages to people's homes or businesses. Overall, indoor localization plays a critical role in enabling these and other applications in robot localization.

2.2.5 Museums

Indoor localization technology has revolutionized the way museums operate and provides a unique experience to visitors. By utilizing location-based services, museums can enhance the visitor experience in many ways. One of the most significant ways is by providing real-time information about exhibits and artifacts. With the use of indoor positioning systems, visitors can access detailed information about the history and significance of a particular exhibit, making the museum visit more engaging and informative.

Another way that indoor localization technology can enhance the museum experience is through crowd management. By tracking visitor movement and analyzing visitor flow, museum staff can optimize exhibit layouts and improve crowd control. This can help reduce congestion in popular areas and provide a more enjoyable experience for visitors.

Indoor localization technology can also be used to improve security within the museum. Valuable artifacts can be equipped with location-tracking devices that allow museum staff to monitor their movement and ensure their safety. Additionally, indoor positioning systems can be used to track visitor movement and detect any suspicious behavior.

Moreover, indoor localization technology can improve accessibility for visitors with disabilities. Museums can provide navigation and wayfinding services to visitors with hearing or visual impairments, making their experience more enjoyable and inclusive.

Overall, the application of indoor localization in museums has revolutionized the way visitors experience exhibits and interact with artifacts. By providing personalized information, improving crowd management, enhancing security, and increasing accessibility, indoor localization technology is becoming a crucial tool for museums worldwide. Our indoor navigation technology can transform each individual mobile device into an interactive tour guide for a museum.

2.2.6 Targeted Advertising

Targeted advertising is a type of advertising to consumers depending on various traits such as behavior, demographics, etc. Targeted advertising has gained a lot of importance in this decade and is currently used by many advertising companies to reach appropriate consumers in order to raise sales. These also help the consumer to get advertisements that are more personalized to them compared to the previous technique of flooding advertisements to all consumers. Indoor localization can be used to support targeted advertising in indoor environments such as shopping malls, supermarkets, and airports. By tracking the location of a user's mobile device and combining it with data on their preferences, shopping history, and demographics, advertisers can deliver personalized ads and promotions to users when they are in close proximity to relevant products or services.

For example, a shopping mall could use indoor localization technology to track the location of a user's mobile device as they move around the mall. Based on the user's previous purchase history and preferences, the mall could then send targeted ads and promotions to the user's mobile device when they are near stores or products that they may be interested in.

In addition to targeted advertising, indoor localization can also be used to improve the overall shopping experience for users. For example, a shopping mall could use indoor localization technology to provide users with real-time directions to stores, promotions, and events. This can help users navigate the mall more easily and find the products or services they are looking for. In the retail sector, indoor localization can be used to provide users with real-time promotions and discounts based on their location within a store. For example, if a user is browsing a specific section of a store, they may receive a targeted promotion for products in that section.

Overall, indoor localization has the potential to transform targeted advertising in indoor environments, enabling advertisers to deliver personalized ads and promotions to users at the right time and in the right place. Indoor localization has the potential to revolutionize targeted advertising in indoor environments, providing users with personalized promotions and information while also helping advertisers gather valuable data on user behavior and preferences. At the same time, it can also enhance the overall shopping experience for users by providing them with real-time information and directions.

Chapter 3

Problem Statement

3.1 Introduction

The need for indoor navigation and tracking systems has grown significantly across diverse industries, including healthcare, logistics, retail, and manufacturing. These systems play a crucial role in achieving precise localization and tracking of objects or individuals within indoor environments, leading to enhanced safety, efficiency, and productivity. Nonetheless, the development of a dependable and accurate indoor navigation and tracking system based on wireless communication technology poses a multifaceted challenge that demands innovative solutions.

3.2 Problem Description

The complexity of indoor environments presents a significant challenge in the development of indoor navigation and tracking systems. Factors such as signal interference, multi-path propagation, and shadowing effects can introduce inaccuracies in localizing and tracking objects or individuals, which can have critical implications in safety-focused domains like healthcare and emergency response.

To address this challenge, existing indoor navigation and tracking systems leverage diverse wireless communication technologies, including WiFi, Bluetooth, RFID, and Ultra-Wideband (UWB). Each technology offers distinct advantages and limitations that necessitate careful consideration. For instance, WiFi signals provide broad coverage but are susceptible to signal interference, while UWB signals enable precise localization at the cost of increased infrastructure and power requirements.

Ensuring real-time and high-accuracy performance is another crucial aspect of developing indoor navigation and tracking systems. In applications like healthcare and emergency response, any delays or inaccuracies in localization and tracking can have severe consequences. Therefore, the system must be meticulously designed to deliver reliable and timely information to users, prioritizing the requirements of these critical scenarios.

3.3 Research Challenges

The problem of developing an indoor navigation and tracking system using wireless communication technology is challenging due to several reasons:

1. **Complex indoor environments**: Accurately locating and tracking users in indoor environments can be challenging due to the presence of various obstacles like walls, doors, and furniture. These physical structures can obstruct or reflect wireless signals, leading to signal degradation and interference. Consequently, the performance of indoor localization and tracking systems can be compromised, making it difficult to achieve precise and reliable results. Overcoming the effects of these obstacles requires innovative techniques and algorithms that can mitigate signal interference and effectively navigate through complex indoor environments.

2. **Interference and noise**: Interference and noise from various electronic devices can significantly impact wireless signals used for indoor localization and tracking. Common devices like WiFi routers, Bluetooth devices, and microwaves emit signals that can interfere with the signals used for positioning. This interference introduces errors and inaccuracies in the location and tracking data, making it challenging to obtain precise and reliable information about the user's position. Developing robust algorithms and signal processing techniques that can mitigate the effects of interference is essential for improving the accuracy and performance of indoor navigation and tracking systems.

3. **Limited accuracy and range**: The current wireless communication technologies, including Bluetooth and WiFi, have certain limitations in terms of accuracy and range when it comes to indoor location and tracking. These technologies are primarily designed for general-purpose communication rather than precise positioning applications. As a result, they may not provide the level of accuracy required for detailed indoor localization. The accuracy of Bluetooth and WiFi-based systems can be affected by factors such as signal strength, signal propagation characteristics, and environmental conditions. The range of these technologies is also limited, which means that their signals may not propagate well through walls or other obstacles commonly found in indoor environments. These limitations can lead to reduced accuracy and reliability in indoor locations and tracking systems based solely on Bluetooth or WiFi.

4. **Privacy and security concerns**: The deployment of indoor navigation and tracking systems raises valid concerns regarding privacy and security, particularly in sensitive environments like hospitals or military facilities. It is crucial to address these concerns and implement measures to ensure the security and privacy of user data. To safeguard privacy, data collection should be conducted in a transparent and consent-driven manner. Users should have control over the collection, storage, and usage of their personal information. Implementing strong data encryption techniques and secure communication protocols can help protect sensitive data from unauthorized access or interception.

5. **Real-time performance requirements**: Real-time performance is a critical requirement for many indoor navigation and tracking applications. The system must be capable of providing timely and up-to-date information to users, especially in time-sensitive scenarios such as emergency response or logistics operations. However, achieving real-time performance while ensuring accuracy and reliability is a significant challenge that needs to be addressed.

3.4 Research Objectives

The main objective of this study is to develop an indoor navigation and tracking system using wireless communication technology that provides accurate and reliable localization and tracking in complex indoor environments. The specific objectives are:

- To review the literature and identify the current state-of-the-art indoor navigation and tracking systems using wireless communication technology.
- To evaluate the strengths and weaknesses of different wireless communication technologies for indoor navigation and tracking.
- To develop new algorithms and techniques to improve the accuracy and reliability of indoor navigation and tracking systems in complex indoor environments.
- To test the performance of the indoor navigation and tracking system in real-world indoor environments and evaluate its effectiveness.
- To explore the potential applications of indoor navigation and tracking systems in different fields and identify the practical benefits.
- To design the indoor navigation and tracking system to provide real-time and high-accuracy performance.

3.5 Significance of the Study

The development of an indoor navigation and tracking system using wireless communication technology holds immense practical significance across diverse fields, including healthcare, logistics, retail, and manufacturing. By enhancing the accuracy and reliability of indoor navigation and tracking, this research endeavor can greatly contribute to the improvement of safety, efficiency, and productivity within indoor environments. The outcomes of this study have the potential to offer valuable insights into the strengths and limitations of various wireless communication technologies employed for indoor localization and tracking purposes. These insights can serve as guiding principles for the development of future systems in this domain. Moreover, this study adds to the existing body of academic literature on indoor navigation and tracking systems, establishing a solid foundation for further research endeavors in this rapidly evolving field. Through collaborative efforts, advancements in wireless communication technology, and continuous exploration of innovative solutions, the potential for revolutionizing indoor navigation and tracking systems is substantial.

Chapter 4

Related Work

In indoor localization, two approaches are commonly used: machine learning and filter-based methods. Machine-learning methods involve the application of supervised and unsupervised learning techniques. Traditional supervised machine learning algorithms such as Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Naive Bayes, and decision trees effectively address data extraction, matching, and indoor localization classification challenges. These methods require a training phase where a model is built based on labeled data. The model learns the mapping relationship between input features (e.g., RSSI measurements) and the corresponding output (e.g., location coordinates) by analyzing hidden layers within the network architecture. This trained model can then be used for accurate localization during online testing. With the increasing complexity of indoor networks and the availability of larger datasets, more advanced supervised machine learning techniques based on neural networks (NN) have been proposed.

Architectures such as Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), Deep Neural Networks (DNN), and Recurrent Neural Networks (RNN) offer improved capabilities for learning complex relationships in data. These models excel at feature extraction and representation learning, enhancing localization accuracy. The training phase, which involves optimizing network parameters, allows for rapid learning and the development of reliable models for subsequent online testing. Fingerprinting techniques have also gained prominence in machine learning-based localization.

These methods utilize collected Received Signal Strength Indicator (RSSI) data or employ RSSI prediction techniques to create a fingerprint database. During the testing phase, the collected RSSI measurements are matched against the pre-built database to estimate the position accurately. Fingerprinting techniques have shown superior performance in complex network scenarios where traditional trilateration methods may be limited. In contrast to supervised learning, unsupervised methods are employed for dynamic updates of network weights or biases in real-time online phases, reducing the need for extensive training.

These unsupervised methods, such as clustering algorithms like K-means or expectation-maximization and anomaly detection techniques like isolation forest, allow for autonomous adaptation and learning from data during online testing. They eliminate the necessity for manual parameter updates and are well-suited for scenarios where the network environment is subject to changes or where labeled training data is scarce. Machine-learning methods, both supervised and unsupervised, offer powerful tools for indoor localization. They enable accurate positioning by leveraging training data, pre-built fingerprint databases, and dynamic online learning, contributing to enhanced localization accuracy, adaptability, and autonomy in indoor environments.

Role of Machine Learning

Machine learning (ML) methods play a crucial role in indoor localization and typically involve an offline training phase followed by a validation or testing process. In the training phase, a substantial amount of collected data is utilized to update and optimize the parameters of the ML model, including weights and biases. This iterative process helps improve the model's performance by learning patterns and relationships within the data. The remaining data is reserved for system verification and position prediction, ensuring the effectiveness of the trained system in real-world scenarios. ML methods can be categorized into supervised and unsupervised approaches. In supervised learning, algorithms such as artificial neural networks (ANN) [11], K-nearest neighbors (KNN) [12], decision trees [13], and support vector machines (SVM) [13, 14, 15] are employed. These methods utilize labeled data, where the input features like received signal strength indication (RSSI) or distance measurements, are associated with known positions or classes. The ML model learns to classify or regress the data, refining the RSSI or distance information to improve the accuracy of trilateration or other positioning techniques.

Unsupervised learning methods, on the other hand, do not rely on labeled data. Instead, they leverage clustering algorithms like K-means [16] or expectation maximization (EM) to uncover patterns and structures within the unlabeled data. These methods enable real-time trilateration and positioning without human intervention, making them suitable for dynamic environments or scenarios where labeled training data is limited. Using ML techniques in indoor localization offers promising solutions for robust and efficient positioning. By leveraging the power of supervised and unsupervised learning, ML methods can enhance the accuracy and reliability of trilateration, facilitate real-time localization without human intervention, and adapt to changing environments or network conditions. This enables more robust and efficient indoor localization systems that can meet the demands of various applications and scenarios.

Neural Networks

Neural networks (NN) have emerged as a prominent branch of machine learning (ML) methods extensively explored for indoor localization. These NN architectures, including feed-forward/artificial neural networks (ANN), convolutional neural networks (CNN), recurrent neural networks (RNN), and deep neural networks (DNN), have shown great promise in improving the accuracy and robustness of indoor localization systems. Feed-forward neural networks (ANN) consist of interconnected layers of neurons, where information flows in a unidirectional manner from input to output. They can learn complex relationships between input features and output predictions, making them suitable for tasks such as regression and classification in indoor localization.

Convolutional neural networks (CNN), on the other hand, are specifically designed for processing structured grid-like data, such as images or sensor data like RSSI features. They utilize convolutional layers to automatically extract spatial features from the input data, enabling effective feature representation and localization in scenarios where spatial information plays a crucial role. Recurrent neural networks (RNNs) are well-suited for processing sequential data and capturing temporal dependencies. They have been successfully applied in indoor localization tasks where the temporal evolution of sensor data, such as time-series RSSI measurements, is essential for accurate positioning. However, it can encounter issues like the vanishing gradient problem and difficulty processing lengthy sequences.

Deep neural networks (DNN) refer to neural networks with multiple hidden layers. Deep learning techniques, enabled by DNN architectures, have revolutionized the ML field and have been extensively used in indoor localization. By incorporating more hidden layers, DNNs can learn hierarchical representations of the input data, allowing for better feature extraction and higher-level abstractions. Consequently, DNN is often combined with other ML methods like CNN and SVM for complex localization scenarios [6].

Ultrasonic and InfraRed

In recent times, there has been a growing interest in accurately estimating the indoor position of mobile devices, given their wide range of applications in today's world. Numerous technologies have been employed to address this challenge, including ultrasonic [17, 18], InfraRed (IR) [19], ultra-wideband (UWB, IEEE 802.15.4a) [20], WiFi (wireless local area network IEEE 802.11) [21] and Bluetooth [22].

Ultrasonic and infrared (IR) approaches offer a cost-effective solution with high precision but usually only provide proximity detection and rely on a direct line of sight (LOS) between the transmitter and receiver. In contrast, technologies like WiFi and Bluetooth, commonly used for proximity beacons, have limitations. Bluetooth, for example, has a shorter range of typically 5-10 meters, necessitating a dense deployment of nodes to cover larger nodes.

Ultra-Wideband (UWB) technology provides excellent ranging accuracy but suffers from a low data rate and a limited installed base. On the other hand, WiFi positioning holds significant appeal due to the widespread deployment of WiFi-enabled devices. Substantial existing WiFi devices make WiFi-based positioning particularly attractive and promising for indoor location estimation [23]. Indoor localization techniques encompass a variety of methods, some of which include wifi-based localization, Bluetooth-based localization, inertial measurement unit (IMU) localization, visual-based localization, ultrasonic-based localization, magnetic field-based localization, radio frequency identification (RFID) localization, dead reckoning localization, and optical-based localization. The underlying principles of these techniques vary and can be used alone or in combination to achieve accurate indoor localization.

Digital Fingerprinting

Digital fingerprinting is undoubtedly a popular technique used to tackle the challenge of indoor localization. In a fingerprinting-based approach, a comprehensive database is initially constructed by capturing detailed measurements of the indoor environment. Subsequently, real-time location inference is acquired by comparing newly collected measurements with the data stored in the database [24]. The process typically involves the following steps: Database Construction, Fingerprinting, and Comparison and Inference. By employing this digital fingerprinting approach, indoor localization systems can leverage the previously collected data to estimate the real-time location of mobile devices within the indoor environment.

Numerous indoor fingerprinting systems currently use WiFi-received signal strength (RSS) values as the basis for their fingerprints. This approach is popular due to its simplicity and minimal hardware requirements. An example of such a system is Radar, the first fingerprinting system to utilize RSS and employ a deterministic method for location estimation [25].

RSS Methods

Radar's use of WiFi RSS fingerprints and deterministic location estimation provides a practical and efficient approach to indoor localization. However, it's worth noting that other fingerprinting systems may employ variations or incorporate additional factors to enhance accuracy or overcome limitations associated with WiFi RSS-based techniques. An example of an RSS-based approach is Horus, which utilizes a probabilistic method to achieve better localization accuracy compared to Radar [26]. However, RSS-based methods suffer from two main disadvantages. Firstly, RSS values exhibit high randomness, and their correlation with propagation distance is weak due to effects such as shadowing fading and multipath interference. Secondly, RSS values provide only coarse information obtained by averaging the amplitudes of all incoming signals without utilizing channel information from different subcarriers. Consequently, localization based solely on RSS values often leads to suboptimal performance [24].

Compared to RSS-based methods, fingerprinting techniques that utilize CSI offer potential advantages in higher localization accuracy. By leveraging the detailed channel information, such as amplitude, phase, and frequency response, the system can overcome the limitations associated with RSS-based methods, including the randomness of RSS values and the lack of fine-grained distance estimation.

RSSI (Received Signal Strength Indication) is widely used as a significant indicator for indoor positioning. It is a valuable metric for estimating the distance between anchor points and the current position, particularly in distance-based localization models like trilateration. Accumulating RSSI measurements from multiple anchor points can help the system estimate the distance or range between each anchor point and the target device. These distance estimates are then used to calculate the device's position using trilateration algorithms or other distance-based localization models. While RSSI measurements are susceptible to various challenges in indoor environments, such as interference, multipath effects, noise, and changing channel conditions, recent research has focused on complementing RSSI with CSI (Channel State Information) to enhance the precision of indoor localization [27]. CSI provides more detailed information about the wireless channel, including amplitude, phase, and frequency response. With the incorporation of CSI alongside RSSI, indoor localization systems can benefit from a richer dataset that captures the various complexities of the wireless channel.

Incorporating additional measurement parameters alongside RSSI and CSI like RTT, DOA (Direction of Arrival)/AOA (Angle of Arrival), TDOA, and TOA [28, 29] can provide a more comprehensive and multi-dimensional view of the wireless signals in the indoor environment. Localization algorithms that account for these parameters can leverage the extra information to enhance accuracy and robustness, especially in challenging scenarios with multipath propagation, interference, and dynamic channel conditions.

Inertial Measurement Units

Indoor location tracking can be achieved using inertial sensor units (IMUs), which provide motion and orientation information. While IMUs can offer comparable or higher accuracy compared to WiFi-based techniques, they come with a higher cost due to specialized hardware requirements. IMUs also suffer from error accumulation over time and require periodic recalibration. In contrast, WiFi-based techniques utilizing RSSI or CSI are more cost-effective, leveraging existing infrastructure and devices. The choice between IMU-based and WiFi-based approaches depends on specific requirements, budget constraints, and the desired trade-off between accuracy and cost in the given indoor localization scenario.

Indoor localization techniques using off-the-shelf smartphone sensors, like SmartPDR [30], have inconsistent accuracy due to sensor limitations and environmental factors. Efforts are underway to improve reliability through sensor fusion and advanced algorithms. SmartPDR utilizes smartphone sensors like accelerometers, gyroscopes, and magnetometers for traditional dead reckoning. It employs step event detection, heading direction estimation, and step length estimation to compute displacement. However, the accuracy of SmartPDR's indoor localization model can be affected by sensor limitations and environmental factors. Enhancements are being explored to improve reliability and accuracy through sensor fusion and advanced algorithms.

Trilateration Method

Trilateration, a classic geometric-based approach for indoor localization, faces challenges due to multipath effects, nonlinear interference, and noise in raw data (e.g., RSSI measurements). Additional processing and classification are required to minimize localization errors caused by these factors. The survey encompasses techniques for LOS (light-of-sight) and NLOS (non-line-of-sight) scenarios, considering diverse indoor network structures and channel conditions. The authors investigate various input measurement data types, including RSSI, TDOA, DOA, and RTT, and their applicability to different application scenarios.

The survey discusses RSSI-based fingerprinting techniques employing supervised machine learning methods such as SVM, KNN, and NN, primarily focusing on the offline training phases. Additionally, utilizing unsupervised methods like isolation forest, k-means, and expectation maximization enhances localization accuracy during online testing phases. The survey extensively explores Bayesian filtering methods, encompassing linear Kalman filters (LKF) and nonlinear stochastic filters like extended KF, cubature KF, unscented KF, and particle filters. The paper emphasizes the suitability of nonlinear methods for dynamic localization models. It goes beyond localization accuracy to discuss significant performance features including scalability, stability, reliability, and algorithmic complexity. The paper takes a comprehensive perspective to compare existing techniques and practical localization models, aiming to improve localization accuracy while simultaneously reducing system complexity [6].

Filter-Based Methods

Filter-based methods, such as the particle filter (PF) and Kalman filter (KF), are widely used in indoor localization and offer practical solutions to estimate the position and trajectory of a mobile device. These methods typically involve three main steps: prediction, measurement, and assimilation. The Kalman filter is a widely adopted filter-based method that operates under the assumption of a linear system model and Gaussian noise. It uses a uni-modal Single Gaussian Model (SGM) to represent the state estimate and provides an optimal solution for linear systems. The Kalman filter relies on linear functions and matrix operations to update the state estimate based on prediction and measurement information. However, its performance can be limited in strongly non-linear scenarios where the linear assumption does not hold.

The extended Kalman filter (EKF) is commonly used in indoor localization to address the non-linearity challenge. The EKF approximates non-linear functions through linearization, enabling the estimation of non-linear state dynamics. It operates by propagating the mean and covariance of the state estimate through a series of linear transformations. While the EKF can handle non-linear systems, it may suffer from performance degradation if the linearization introduces significant errors. An alternative to the EKF is the unscented Kalman filter (UKF), which offers a sub-optimal solution for non-linear indoor localization. The UKF avoids the linearization step and directly approximates the probability distribution using a set of carefully chosen sigma points. These sigma points capture the mean and covariance information and propagate through non-linear functions, resulting in more accurate state estimation compared to the EKF. The UKF is particularly suitable for scenarios with moderate non-linearities, where the linearization assumption of the EKF may not hold [6].

Indoor Positioning

With the recent surge in interest regarding indoor localization, a multitude of research endeavors have emerged, seeking to leverage the currently available technology to tackle this challenge. In contrast to outdoor localization, where GPS-assisted methods have proven effective, they fall short in providing accurate indoor positioning. This disparity primarily arises from the restricted coverage range of indoor networks and the complexities associated with channel fading issues. As a result, several studies are exploring alternative approaches and innovative techniques to overcome these limitations and improve the accuracy of indoor localization systems. The field of indoor localization has witnessed the emergence of various technologies [31, 32, 33] playing significant roles in achieving accurate positioning. Prominent among these technologies are WiFi, Bluetooth, Zigbee, UWB (Ultra-wideband), RFID (Radio-identification), Ultrasound, and iBeacons.

These technologies have garnered attention and are actively utilized in diverse indoor localization applications, each offering unique advantages and addressing specific requirements based on factors like range, precision, power consumption, and deployment flexibility. The availability of multiple options allows for choosing the most suitable technology depending on the specific needs and constraints of the indoor localization scenario. Indoor positioning can be classified into two distinct categories: Line of Sight (LOS) and Non-Line of Sight (NLOS), based on the deployment and coverage range of access points (APs) within the indoor environment.

When analyzing localization in these scenarios, it becomes crucial to consider a myriad of attenuation and channel models influenced by factors such as the presence of obstacles like walls, including their number, thickness, and material properties. To accurately estimate distances, it is essential to collect distance indication data while accounting for these attenuation factors within different fading channel scenarios. This enables more robust and reliable indoor localization by accounting for the impact of obstacles and channel conditions on the propagation of wireless signals.

Our Indoor Localization technology

Our indoor localization technique is based on the standard IEEE 802.11 WLAN protocols, particularly the IEEE $802.11mc$ amendment, published as an amendment to the $802.11-2012$ protocol and eventually included in the $802.11-2016$ WLAN standard protocol. The IEEE $802.11mc$ protocol was proposed to enhance the time measurement technique previously introduced in IEEE $802.11v$ and was named the fine time measurement (FTM) technique. The FTM technique revolutionizes distance measurement in indoor localization by addressing the limitations of the traditional received signal strength indication (RSSI) technique, known for its susceptibility to various environmental factors and inaccuracies. By incorporating time-of-flight measurements, FTM enables WiFi devices to determine their distance from an access point with improved precision.

The FTM technique measures the duration of a frame to travel through the air between a WiFi device and an access point. This time measurement is then utilized to calculate the distance between the device and the access point. By performing these distance measurements with multiple access points whose locations are known, the precise location of the WiFi device can be determined [1] through techniques such as trilateration or fingerprinting. This breakthrough in indoor localization has been implemented in Android smartphones, including the flagship smartphone *Google*

Pixel 2, which leverages the WiFi-based FTM feature. This capability enables smartphones to accurately measure the distance between the device and an access point or between two smartphones. By utilizing FTM and the standardized IEEE $802.11mc$ protocol, our indoor localization technique offers enhanced accuracy and reliability for various applications and scenarios.

Chapter 5

System Design

In this chapter, we will delve into the design of our indoor localization technique, which utilizes smartphone-based IEEE $802.11mc$ fine time measurement (FTM). The significance of indoor navigation and tracking systems in various applications today cannot be overstated. Our indoor localization design encompasses three key techniques: Android WiFi Aware technology, Android WiFi Fine Time Measurement (FTM) based on round trip time computation, and a location estimation algorithm employing the Trilateration/Multilateration technique along with the sophisticated grid method algorithm. The subsequent sections of this chapter will provide a comprehensive overview of each of these techniques, outlining their functionalities and implementation details.

5.1 Android WiFi Aware Technology

Our indoor localization technique was specifically designed for the Android operating system, leveraging the capabilities of standard WiFi Aware and WiFi location with RTT techniques available in the Android 9.0 OS, also known as Android Pie [34]. The WiFi RTT feature, which forms the foundation of our technique, is based on the fine time measurement (FTM) technique standardized by the IEEE $802.11mc$ protocol in the $802.11 - 2016$ WLAN protocol. While the WiFi RTT feature can be utilized independently without the support of WiFi Aware, we integrated the WiFi Aware technique into our approach to enable connectivity between smartphone devices solely through WiFi, without the need for access points or additional infrastructure modifications. By incorporating WiFi Aware, smartphone devices can seamlessly discover and connect with neighboring devices, even operating in ad-hoc mode, without relying on an access point for routing purposes.

Our technique utilizes the WiFi Aware service to facilitate device discovery, while the WiFi RTT technique is employed to measure the round trip times (RTTs) and calculate the distance between each device within the cluster. This combined approach enables our technique to accurately estimate distances and establish connectivity among devices, enhancing the overall effectiveness and functionality of our indoor localization system.

The Android Aware system is a feature within the Android operating system that empowers applications to gather information about the user's surroundings and context. This includes data such as the user's location, activity level, nearby Bluetooth devices, and other environmental factors. By providing a set of application programming interfaces (API), the Android Aware system allows developers to create context-aware apps that can adapt to the user's current situation and deliver personalized experiences. With the Aware API, developers can build apps that dynamically adjust their settings based on the user's location or activity or provide relevant notifications based on nearby Bluetooth devices. The Android Aware system prioritizes user privacy, ensuring that users have control over which apps can access their contextual data and the ability to revoke access at any time.

In terms of networking, the WiFi Aware feature operates by forming clusters with neighboring devices or creating a new cluster if the device is the first in a given area. This clustering behavior is managed by the WiFi Aware system service, with apps having no control over it. The clustering mechanism applies to the entire device, facilitating communication and interaction between devices in the vicinity. By leveraging the Android Aware system and its networking capabilities, developers can create innovative apps that make use of contextual information to enhance the user experience while maintaining user privacy and control over their data [35].

A network cluster is a group of interconnected devices or computers that work together to perform a specific task or provide a specific service. The devices in a network cluster typically communicate with each other to coordinate their activities and share resources. An Android Aware network cluster refers to a group of interconnected devices or computers running the Android operating system, that are able to communicate with each other and share contextual data using the Android Aware feature. Such a network cluster could be used, for example, to enable context-aware services or applications that rely on shared contextual data from multiple devices in the network. In our system, we use android aware network cluster to manage the network and to discover the nearby devices that are within this network cluster. The smartphones in these network clusters are divided into 2 types based on their functionality. A publisher is a smartphone in an android aware network cluster capable of broadcasting messages to all the subscribers within the same network. And a subscriber is a smartphone in an Android aware network cluster capable of receiving broadcast messages from all the publishers within the same network. Once the WiFi Aware technique is turned on, the device is capable of becoming either a publisher or a subscriber of a WiFi Aware cluster network. In our indoor localization system design, we made the reference nodes (smartphone devices that have already obtained their indoor location) publishers and the target nodes (smartphone devices that are seeking their indoor location) as subscribers.

Within the network cluster, there can be multiple publishers and subscribers. Publishers have the ability to broadcast messages to subscribers, allowing them to subscribe and establish a connection. When a subscriber chooses to subscribe to a publisher, a PeerHandle is generated by the publisher. The subscriber can then use this PeerHandle to establish a connection or send messages to the publisher even without a formal connection. In our indoor localization system, we utilize this generated PeerHandle to facilitate round-trip time communication and compute the distance between the publisher and subscriber. The publisher broadcasts messages to make all in-range subscriber smartphones aware of its presence, typically for a duration of 15 to 20 seconds. If a subscriber intends to communicate with the publisher, a PeerHandle is generated to facilitate the connection. Otherwise, the publisher smartphone goes dormant until it is reactivated to broadcast messages. This approach helps conserve battery power as continuous broadcasting would consume a significant amount of energy. However, for our experiments focusing on testing the accuracy of the indoor localization system, battery performance is not a primary concern. Thus, we periodically reactivate the publisher smartphone to continuously broadcast messages during these experiments.

The content of the publisher's broadcast message can vary based on the specific application requirements. For example, it can include a simple greeting message, the name of the network cluster, or information related to a specific hospital ward. In our system, we include the position information of the publisher smartphone within the broadcast message. This position information plays a crucial role in determining the location of the subscriber smartphone using the round trip time distance computed using the Android WiFi Fine Time Measurement (FTM) technique.

Upon receiving a broadcast message from a publisher, the subscriber smartphone has two options: it can choose to subscribe to the publisher, establish a connection for communication, or it can directly perform the WiFi Fine Time Measurement (FTM) technique to compute the distance to the publisher smartphone. In our system, we opt for the latter approach, calculating the distance between the subscriber and publisher directly without establishing a connection. This method allows us to avoid the time delays associated with connection handshaking.

A subscriber smartphone has the capability to perform the FTM technique with multiple publishers simultaneously, enabling distance calculations between the subscriber and each respective publisher smartphone. Once the subscriber has performed the FTM with at least three publisher smartphones, it can proceed with the localization grid method to compute the location. It is important to note that only three publishers in-range are required to achieve an acceptable level of location accuracy. Increasing the number of publishers in range can further enhance the accuracy of localization. However, there is a tradeoff between the number of publishers and the performance of the system. As the number of publishers increases, the time required for the subscriber to collect all the FTM distances also increases, potentially leading to a decrease in overall system performance. Thus, finding the right balance is crucial to ensure optimal accuracy and system efficiency.

In our system design, we have established criteria to determine whether a smartphone will function as a publisher or a subscriber. Within an Android WiFi Aware network cluster, smartphones that have already determined their locations are designated as publisher smartphones. These publishers play a crucial role in assisting the subscriber smartphones in computing their own locations using the WiFi Fine Time Measurement (FTM) technique and the localization grid method algorithm. On the other hand, smartphones that have yet to determine their locations within the network cluster are classified as subscriber smartphones. These subscribers opportunistically communicate with the publisher smartphones and leverage the FTM technique to compute their own locations. By distinguishing between publishers and subscribers, our system optimizes the utilization of available resources and ensures efficient collaboration among smartphones in the network cluster. This division of roles facilitates the accurate determination of locations for all smartphones involved in the indoor localization process.

When a subscriber smartphone establishes communication with a publisher smartphone in our system, the publisher smartphone generates a PeerHandle for identification purposes. This choice of using a PeerHandle instead of the MAC address of the smartphone is driven by privacy concerns. MAC addresses are unique to each smartphone and cannot be changed, making them potentially traceable to individual users. In contrast, PeerHandles offer a higher level of privacy protection. They are periodically updated each time they are generated, which is an inherent feature of the Android Aware network system. This mechanism helps to safeguard the privacy of users by ensuring that their identities remain protected during communication within the network cluster. By using PeerHandles instead of MAC addresses, our system prioritizes user privacy while maintaining effective communication between smartphones.

The android WiFi aware APIs let apps perform the following operations:

Discover other devices

The Android Aware API provides a mechanism for discovering nearby devices within a network cluster. The process begins with a device publishing one or more discoverable services. When another device subscribes to one or more of these services and enters the Wi-Fi range of the publisher, the subscriber receives a notification indicating the discovery of a matching publisher. Once the subscriber identifies a publisher, it has two options: it can either send a short message to the publisher or establish a network connection with the discovered device. This bidirectional communication allows devices to concurrently act as both publishers and subscribers within the network cluster. By utilizing this API, our system enables devices to efficiently discover and interact with each other based on their published services, facilitating seamless communication and collaboration within the indoor localization network.

Create a network connection

Once two devices have discovered each other using the Android Aware API, they can establish a bi-directional Wi-Fi Aware network connection without the need for an access point. This means that the devices can directly communicate with each other using Wi-Fi technology, forming a network between themselves. The establishment of this network connection allows for efficient and direct data exchange between the two devices, enabling seamless communication and collaboration. This capability enhances the functionality of our indoor localization system, as it enables devices within the network cluster to exchange important information for localization purposes without relying on external infrastructure or intermediaries.[35]

5.2 Android WiFi Fine Time Measurement (FTM) based on Round trip time computation

The introduction of the IEEE $802.11mc$ fine time measurement (FTM) feature in the WiFi standard has revolutionized indoor localization capabilities. By leveraging round-trip time (RTT) measurements, FTM enables precise location determination in indoor environments. It achieves this by utilizing time stamps in Wi-Fi frames to estimate the time of flight of signals between mobile devices and Wi-Fi access points. The IEEE $802.11mc$ amendment encompasses various other features besides FTM. For instance, enhanced beacon frames provide more accurate and frequent updates of access point locations. Additionally, a standardized interface facilitates the exchange of location information between devices and access points. These advancements have greatly enhanced the accuracy and efficiency of indoor localization, opening doors to applications like asset tracking, indoor navigation, and location-based services.

Android has integrated the IEEE $802.11mc$ FTM technique into its operating system, making it readily accessible to developers through the *LocationManager* API. When an Android application requests location updates, the *LocationManager* triggers a signal to nearby Wi-Fi access points. These access points respond with Wi-Fi frames that include time stamps. Android utilizes these timestamps to calculate the RTT of the signal and, subsequently, the distance between the mobile device and the access point, taking the speed of light into account. Notably, this communication can occur directly between Android smartphone devices in ad-hoc mode, eliminating the need for access points. The Android *LocationManager* API provides access to both raw FTM measurements and location estimates derived from these measurements. This integration empowers developers to leverage FTM for indoor localization within their applications, enhancing user experiences and enabling a wide range of location-based functionalities.

The Android WiFi Fine Time Measurement (FTM) technique, available in the Android 9.0 operating system, allows for distance measurement between a smartphone device and a WiFi Round Trip Time (RTT) capable access point (AP), or between peer WiFi Aware devices. This is achieved by computing the RTT through the packet travel between devices, as illustrated in Figure 5.1.

When the communication is between a smartphone and an access point, only the smartphone is capable of measuring the distance. This approach maintains the privacy of smartphone devices. However, in our indoor localization technique, we leverage the WiFi Aware technique discussed earlier in Section 5.1 for discovery and connectivity among peer smartphone devices. This introduces trust within the network cluster, enabling any smartphone device within the cluster to compute the distance using the WiFi RTT technique [36]. To address the time clock synchronicity issue, we cannot solely rely on one-way time differences to compute the distance between smartphones. Instead, we employ the Round-Trip-Time (RTT) measurement, as depicted in Figure 5.1. Synchronizing time clocks on smartphone devices would require significant processing power and drain battery life extensively. In RTT estimation, the clock offset is opposite when the signal travels back from the smartphone or access point compared to the initial request, allowing for accurate distance calculations.

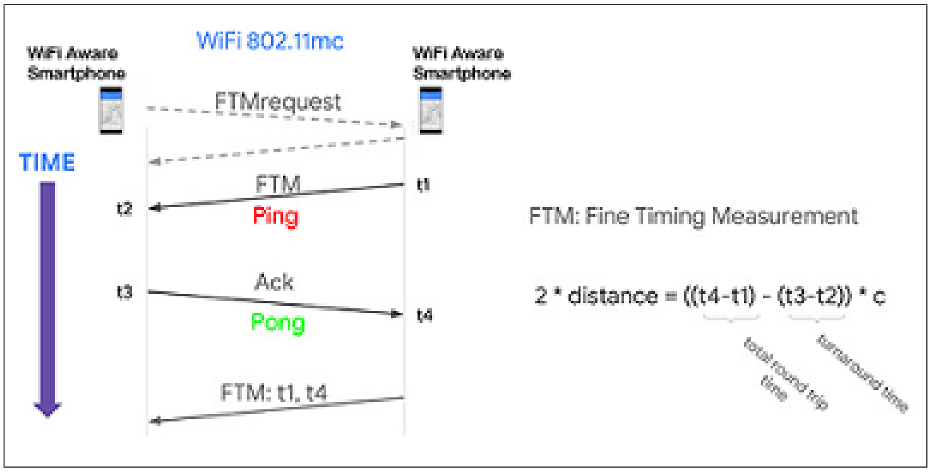

Figure 5.1: WiFi Fine Time Measurement (FTM) using Round Trip Time (RTT) estimation

The RTT measurement technique is a fundamental component of the IEEE $802.11mc$ WiFi protocol's Fine Time Measurement (FTM) technique [37]. It overcomes the challenges of time clock synchronization and enables precise distance measurements, contributing to the overall accuracy and reliability of our indoor localization system.

$$2 * d = ((t_4 - t_1) - (t_3 - t_2)) * c \tag{5.1}$$

The fine time measurement (FTM) is computed according to Equation 5.1. In this equation, d represents the distance between two WiFi Aware smartphone devices, and $t1, t2, t3$, and $t4$ represent the timestamps captured at each interval of the FTM technique, as illustrated in Figure 5.1. The speed of light is denoted by c.

In our indoor localization design, once the Android WiFi Aware connection is established and the publishers are part of the cluster, the target node (subscriber) is designated as the master. The master initiates the WiFi RTT communication to estimate the distance by collecting the RTT data from the reference nodes (publishers), which act as slaves in this context.

5.3 Location Estimation Algorithm based on GPS Trilateration

Indoor GPS navigation is known to be unreliable and performs poorly due to the limited penetration of satellite signals inside buildings [38]. While GPS cannot be used effectively indoors, our indoor navigation technique employs a trilateration method similar to GPS. By utilizing Android WiFi Aware and Android WiFi RTT techniques, we can calculate the distances between the target node (master) and the reference nodes (slaves). Once we have obtained these distances using the WiFi RTT technique from three or more neighboring devices within the WiFi Aware networking cluster, we apply the multilateration technique, illustrated in Figure 5.2, which is akin to the GPS positioning technique. This allows us to determine the location of the target node (master). In Figure 5.2, $R1$, $R2$, and $R3$ represent the reference nodes with locations (x_1, y_1), (x_2, y_2), and (x_3, y_3), respectively, while T represents the target node with location (x_T, y_T). Assuming that the distances between the reference nodes and the target node are denoted as d_1, d_2, and d_3 for $R1$, $R2$, and $R3$ respectively, the location of the target node T can be computed using Equation 5.2.

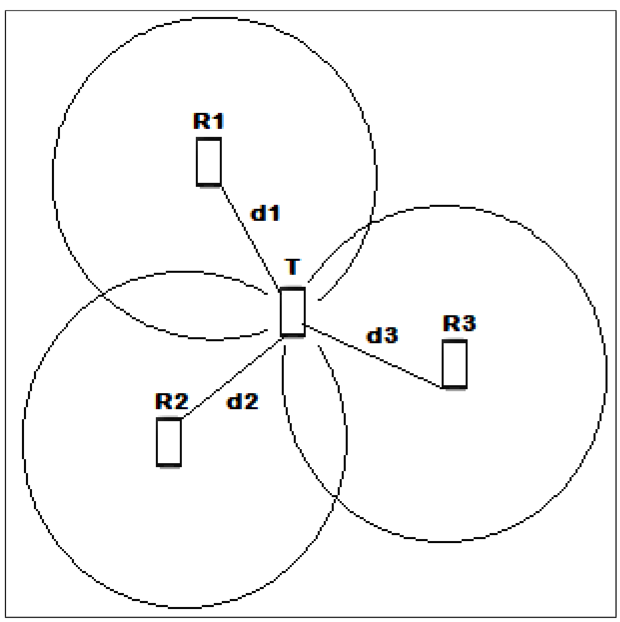

Figure 5.2: GPS Trilateration technique

$$
\begin{aligned}
(x_T - x_1)^2 + (y_T - y_1)^2 &= d_1^2 \\
(x_T - x_2)^2 + (y_T - y_2)^2 &= d_2^2 \\
(x_T - x_3)^2 + (y_T - y_3)^2 &= d_3^2
\end{aligned} \tag{5.2}
$$

Once the distances between the target node and the reference nodes are determined using Equation 5.1, our indoor localization technique proceeds to compute the location of the target node using a location estimation algorithm. This algorithm solves an optimization problem that arises from the intersection of circles formed by taking the distance between the target node and reference nodes as the radii, and the reference node locations as the centers. Ideally, if all these circles intersect at a single point, it would represent the estimated location of the target node. However, in real-world scenarios, such a perfect intersection is rare due to various factors.

To solve this optimization problem, we employ the concept of the center of gravity. This technique helps to reduce the error in the location estimation by considering the center of gravity as the result. The center of gravity (COG) is akin to the centroid of a cluster of points, assuming each point has unit mass. It should be noted that the center of gravity and center of mass for any arbitrary body are equivalent. In our technique, we utilize the COG method to calculate the center of a cluster of target node positions generated after each iteration, utilizing Equation 5.3.

$$
C = \frac{\sum_{i=1}^{N} P_i}{N} \tag{5.3}
$$

In Equation 5.2, C represents the Center of gravity of the cluster of estimated target node positions, N is the total number of iterations/Grid Points and P_i is the target node position coordinate of the i_{th} iteration.

5.4 Sophisticated Grid Method Algorithm

We have developed a technique called the grid method to determine the position of the target node (subscriber smartphone) using the coordinates of the reference peer nodes (publisher smartphones) and the distances obtained from smartphone-based WiFi Fine Time Measurement (FTM).

This method utilizes multilateration, where we calculate the intersection area of concentric circles (donut circles) formed by using the coordinates of the reference peer nodes as centers and the WiFi FTM distances as radii. To define these concentric circles or donut circles, we introduce the concept of ”donut widths”. Starting with the WiFi FTM distance as the radius, we subtract 20% of the radius to obtain the lower donut boundary and add 20% of the radius to obtain the upper donut boundary. These boundaries delineate a ”donut-shaped” region of space around the target node (subscriber smartphone).

These concentric circles are then overlaid onto a grid composed of $1m$-by-$1m$ grid points. By considering only the intersection area of all the circle donuts, we focus on the region with the highest probability of containing the position of the target node. We extract all the grid points within this intersection area. Subsequently, we employ the Center-of-Gravity (COG) technique to calculate the center of this cluster of grid points. This center represents the estimated position of the target node obtained through the Grid method. Figure 5.3 illustrates the Grid method with grid points and reference nodes using multilateration in a Matlab figure.

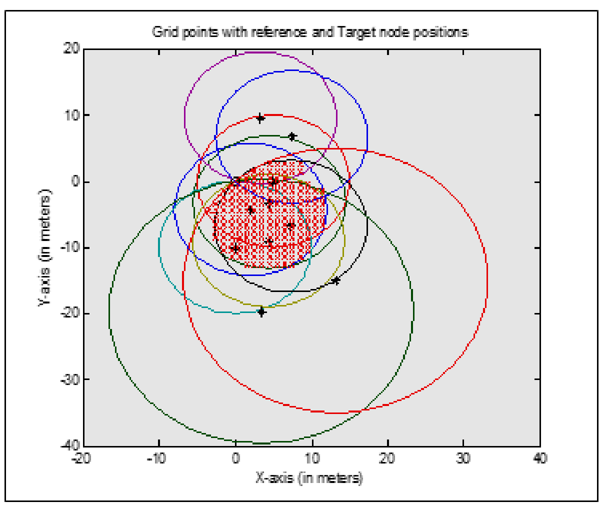

Figure 5.3: Plot of all the grid points in the intersection area including the reference node and real target node positions

5.5 Modes of Operation based on Smartphone WiFi FTM technology

There are two modes of operation and the details are discussed in the following sections.

5.5.1 Ad-Hoc Mode

The Android-aware technology enables a mode where smartphones can establish communication and compute round-trip time (RTT) data without relying on an Access Point (AP). In this mode, smartphones can operate as either publishers or subscribers within the network. The decision to become a publisher is based on whether the smartphone's location is known. If the location is known, the smartphone can function as a publisher; otherwise, it will operate as a subscriber. Publishers continuously broadcast their presence in the network, allowing subscribers to locate and establish communication for RTT ranging. Subscribers can identify publishers using their MAC addresses or PeerHandles, which are broadcasted by the publishers. Once a subscriber successfully establishes a connection with at least three publishers, it can determine its own location using multilateration and the Grid method. After completing this process, the subscriber smartphone can transition to becoming a publisher, offering its services to other subscribers within the network. This mode enables smartphones to seamlessly switch between being publishers and subscribers, facilitating efficient communication and location determination within the Android-aware network.

5.5.2 AP Mode

In this mode of communication, the process is akin to Ad-Hoc mode, but with a distinction: instead of a Publisher, it is the Access Point (AP) that broadcasts messages in the network. The purpose of this broadcast is to allow subscriber smartphones to discover the AP and initiate WiFi RTT ranging. APs are WiFi routers equipped with WiFi RTT-capable hardware, while subscribers are standard smartphones. Just like in Ad-Hoc mode, subscribers have the ability to become publishers once they compute their own location. The network can consist of a combination of APs and publishers, providing a variety of options for subscribers to discover and range in order to compute their respective locations. This mode facilitates efficient communication and location determination by leveraging both APs and publishers within the network.

5.6 Modes of Operations based on the FireFighter Localization Application

The determination of these modes of operation was based on an extensive research process that involved conducting over a hundred in-person interviews with firefighters. The primary objective of these interviews was to gain a deep understanding of the firefighters' work environment, the specific rules and guidelines they follow during fire scenes, and their tasks as part of the Rapid Intervention Team (RIT). By gathering insights directly from firefighters, we were able to identify the most relevant modes of operation for the smartphone application. These modes were designed to align with the specific needs and requirements of firefighters in their operational contexts. Additionally, the performance of the smartphone application and the level of accuracy required for the position information were also taken into consideration during this research process. Overall, the insights gathered from the interviews played a crucial role in determining the modes of operation and ensuring that the smartphone application meets the specific needs of firefighters in terms of performance and accuracy.

5.6.1 MayDay Mode

In the MayDay mode of operation, the smartphone application operates at its maximum performance level to provide the most accurate real-time location information for all firefighters working in the fire scene environment. This mode is specifically designed for critical scenarios where firefighters may find themselves trapped inside without a clear escape route. When a MayDay signal is issued by the Fire Chief, indicating that a firefighter is in distress, the Rapid Intervention Team (RIT) is deployed to locate and rescue the firefighter. The MayDay mode of the indoor localization system is optimized to prioritize performance and accuracy over preserving the smartphone device's battery life. In this mode, the Android application works diligently to determine the most precise position for the target node, ensuring that the RIT can quickly and accurately locate and rescue the firefighter in need.

5.6.2 Battery Saving Mode

In this mode of operation, the emphasis is not on achieving the highest level of accuracy in position determination, as it is in the MayDay scenario. Instead, the focus is on balancing the accuracy of the position information with the conservation of battery energy. Firefighters are actively engaged in firefighting and rescue operations, including civilian rescue missions. Therefore, the system operates in a mode where the collection of WiFi RTT data for position estimation is performed at regular intervals rather than in real time. This approach allows for an acceptable level of accuracy while optimizing the energy consumption of the smartphone's battery. By collecting RTT data periodically, the application strikes a balance between providing useful position information and conserving the device's battery power for extended operational duration.

Chapter 6

Experimental Setup

6.1 Implementation

In this section, we will delve into the implementation details of our indoor localization technique. The technique can be divided into two main parts: the RTT distance estimation and the location estimation using sophisticated algorithms. Let's explore each part in more depth.

The first part of our indoor localization technique leverages two key technologies: Android WiFi Aware and Android WiFi RTT. These technologies, discussed in detail in sections 5.1 and 5.2 respectively, form the foundation of our approach. Android WiFi Aware enables devices to discover and communicate with each other without relying on an internet connection or traditional access points. This functionality allows our system to establish communication and exchange data between the target device and reference nodes. Android WiFi RTT, on the other hand, provides precise distance measurement capabilities by utilizing the Round-Trip-Time (RTT) technique. It allows us to calculate the time taken for signals to travel between the target device and reference nodes, enabling accurate distance estimation.

By integrating these technologies, we are able to obtain reliable RTT distance estimates, which serve as crucial input for the subsequent location estimation process. In the following sections, we will delve deeper into the algorithms and methodologies employed for location estimation, enabling us to determine the precise position of the target device based on the RTT distance data.

To evaluate the accuracy and performance of our indoor localization technique, we conducted extensive testing at the Auburn University Shelby Center. The testing area was carefully divided into three distinct zones, each designed to simulate different indoor scenarios and assess various aspects of our technique.

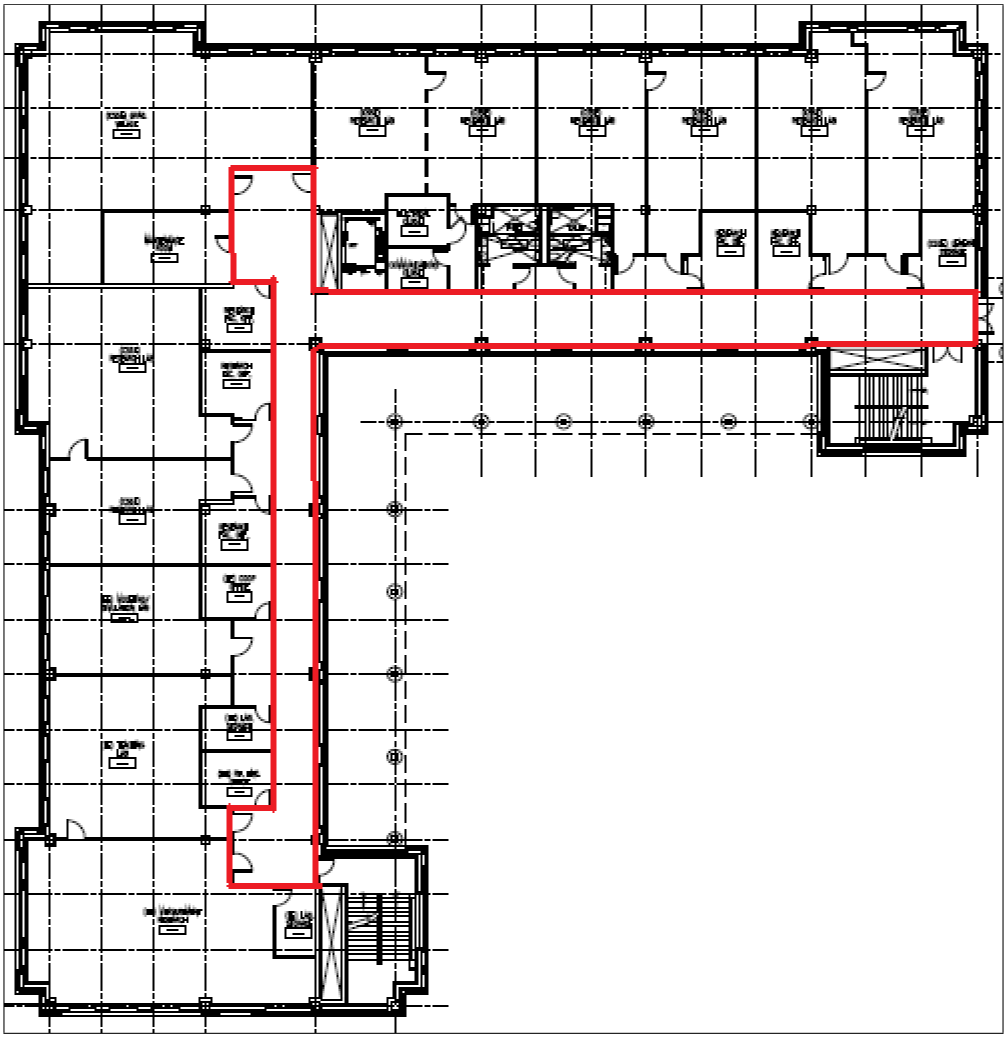

Figure 6.1: Auburn University Shelby Center Hallway Blueprint showing the testing location for Line-of-Sight (LOS) Scenario

The first zone, depicted in Figure 6.1, focused on testing indoor localization in a hallway environment. This zone allowed us to assess the accuracy and effectiveness of our technique in a straightforward Line-of-Sight (LOS) scenario. By conducting tests in this controlled setting, we could gather valuable data to validate our approach. Expanding beyond the hallway environment, we proceeded to the next testing zone, as illustrated in Figure 6.2. This zone encompassed both the multiroom and hallway areas of the Shelby Center. By incorporating multiple rooms and a hallway, we aimed to evaluate the performance of our technique in more complex indoor environments. This

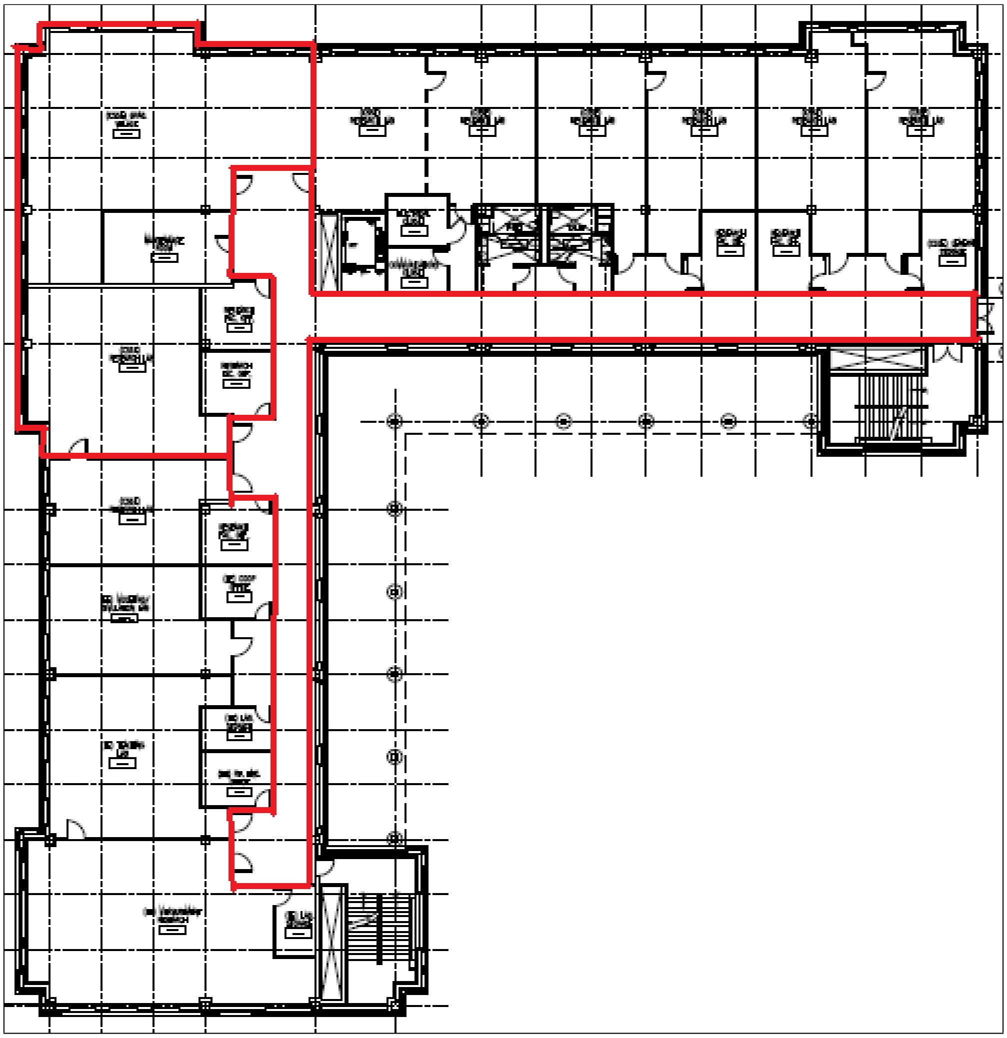

Figure 6.2: Auburn University Shelby Center Hallway and room Blueprint showing the testing location for non-Line-of-Sight (NLOS) Scenario

zone facilitated testing in both Line-of-Sight (LOS) and non-Line-of-Sight (non-LOS) scenarios, simulating real-world conditions where obstacles and signal obstructions are present.

Finally, we designated a separate testing zone solely for multiroom scenarios, as depicted in Figure 6.2. This zone allowed us to assess the performance of our technique in a setting where multiple rooms are involved, each posing unique challenges for indoor localization. By examining the results in this zone, we gained insights into the scalability and adaptability of our technique when applied to larger indoor spaces. By carefully designing these testing zones and conducting experiments within them, we were able to comprehensively evaluate the capabilities of our indoor localization technique across different indoor scenarios. The collected data and observations served as a crucial basis for assessing the accuracy, robustness, and suitability of our technique for real-world applications.

Table 6.1: Android Smartphone Specification

Specification	**Value**
Smartphone Name	Google Pixel 2
Processor	Qualcomm MSM8998 Snapdragon 835
CPU Frequency	2.35 GHz
Communication Frequency Wireless Protocol	2.4 GHz IEEE 802.11 mc
OS Kernel	Android 9.0 (Pie)

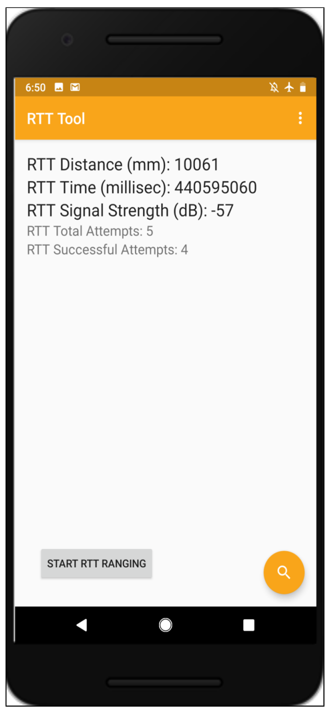

Figure 6.3: Android Smartphone RTT distance estimate result

Our indoor localization technique utilizes the capabilities of Android WiFi Aware and WiFi RTT, implemented on Google Pixel 2 smartphones running the Android 9.0 operating system, also known as Android Pie. To develop the indoor localization application, we followed the guidelines provided by the Android standard development kit (SDK) using the Android Studio software platform.

The application interface captures the distance between smartphones and displays it on the screen, as shown in Figure 6.3. This interface allows users to visualize the real-time distance measurements obtained through the WiFi RTT technique.

The accuracy of the estimated location and the distance measurements using WiFi RTT are thoroughly examined and analyzed in the Performance Analysis chapter, as referenced in Chapter 7. This analysis provides detailed insights into the performance of our technique in terms of location accuracy and RTT distance measurement accuracy.

For real-world testing, we selected smartphones that meet the necessary specifications. The specifications of the smartphones used in our experiments are listed in Table 6.1, providing a comprehensive overview of the hardware details that contributed to the implementation and evaluation of our indoor localization technique.

6.2 Case Study: Mobility Scenario Strategy

This section focuses on various indoor positioning and navigation scenarios that can potentially result in poor accuracy or uncertain outcomes. We will explore the strategies employed by our technique to address these challenges. By examining these scenarios, we aim to provide a clearer understanding of the behavior of our smartphone-based indoor localization technique utilizing fine time measurement (FTM).

6.2.1 What if all the reference smartphone devices are aligned in the same direction?

The GPS technique faces a particular issue called Geometric Dilution of Precision (GDOP), as illustrated in Figure 6.4, where all the satellites align in a single direction. This alignment prevents the trilateration technique from finding a unique intersection point to accurately determine the location of the GPS receiver, resulting in diminished position accuracy. Consequently, the precision of the location becomes diluted. Similarly, our localization technique may encounter similar challenges, leading to reduced accuracy in estimating the position.

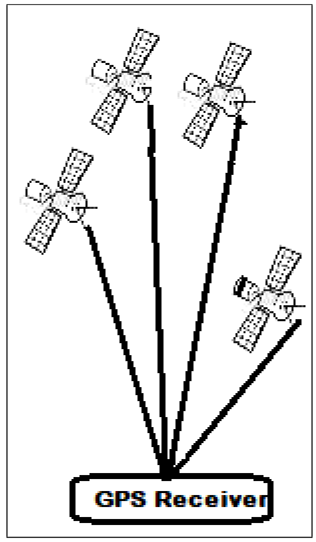

Figure 6.4: Android Smartphone RTT distance estimate result

6.2.2 How do smartphone devices obtain their initial location information?

As mentioned previously, our technique leverages the WiFi Aware feature offered by the Android operating system to discover and establish connections with nearby smartphones. To acquire the initial location information, our technique relies on the GPS functionality present in every smartphone device. If any device within the cluster can establish communication with a device capable of obtaining GPS coordinates (typically located outside the building), the initial location information is acquired through this approach. This strategy is particularly useful in firefighter scenarios where the fire chief, who often coordinates the task from outside, can provide GPS coordinates.

6.2.3 What if there is no reference smartphone device in range?

In certain scenarios, it is plausible for the target smartphone devices to be out of range from other smartphones that can communicate using WiFi RTT technology. In such situations, the system needs to resort to alternative techniques, such as the Pedestrian Dead Reckoning (Inertial sensor units) system available in smartphones, to generate location information until a nearby smartphone comes within range of the target device. This scenario is often encountered in firefighter environments, and it is advantageous to develop a system that incorporates multiple localization technologies to ensure a more robust and reliable solution, rather than relying solely on a single system.

6.2.4 How many reference smartphone devices are required to obtain acceptable location accuracy?

Through a series of extensive indoor localization experiments conducted at Auburn University Shelby Center, we have determined that in order to achieve satisfactory location accuracy of less than $2m$, it is necessary to have at least three reference nodes within the proximity of the target node. These reference nodes play a crucial role in providing the necessary signal measurements and data for precise localization calculations. By ensuring the presence of a sufficient number of reference nodes, we can enhance the accuracy and reliability of our indoor localization system.

6.3 Experimentation Setup to measure the error propagation in Smartphone WiFi RTT localization

Error propagation refers to the phenomenon where errors that occur in one part of a system can propagate or spread throughout the entire system, potentially causing more errors or disruptions. In a WiFi network, error propagation can occur due to various factors such as interference from other electronic devices, physical obstructions like walls or buildings, and distance between the access point and the device. When errors occur in the transmission of data between the access point and a device, they can propagate and cause further errors in subsequent transmissions, leading to degraded network performance, dropped connections, and reduced data throughput. In a multi-hop wireless network, error propagation can occur when a transmission error in one node can affect the entire network. To measure error propagation in such a network, researchers typically use simulation-based approaches that involve modeling the network and evaluating its performance under various conditions. This involves developing simulation models that consider the network topology, channel characteristics, and the type of error-correction scheme used in the network.

Our error propagation system is specifically designed to assess the extent to which errors propagate in the estimated location when employing smartphone-based WiFi FTM and a sophisticated grid method. To evaluate error propagation, we introduce an error at the first hop subscriber and measure the resulting error at each subsequent multi-hop neighbor. This enables us to determine the propagation behavior of errors within our system. For instance, if there is a 5% estimated error in the location of the first hop subscriber smartphone, we intentionally introduce this error into the system when estimating the location of the second hop subscriber smartphone. We then analyze and compare the error values obtained with and without introducing the initial error, allowing us to calculate the percentage of error propagation throughout the system.

To conduct our experiment, we performed indoor tests involving multiple smartphones utilizing WiFi FTM-based localization. Additionally, we conducted error propagation measurements in a scenario where the first publisher smartphone obtained its location information from GPS while situated outside the building, assisting the localization process of the subscriber smartphone located inside. We carried out these experiments up to the fourth hop neighbor, systematically evaluating the error propagation at each stage.

Chapter 7

Performance Evaluation

This chapter presents the findings of a real-world experiment focused on indoor localization, specifically utilizing Google Pixel 2 smartphones. The experiment involved measuring the Round-Trip Time (RTT) distance using the Android WiFi RTT technique. The tests were conducted in three distinct testing zones within the Auburn University Shelby Center, as discussed in detail in section 6.1.

To provide a visual reference, the specific testing zones are highlighted in Figures 6.1 and 6.2, showcasing their locations on the Shelby Center blueprint map. These figures offer a clear depiction of the areas where the experiments took place, aiding in understanding the spatial context of the findings.

7.1 Evaluating the Effectiveness of Smartphone WiFi RTT for Precise Indoor Distance Measurement

The initial real-world testing phase focused on evaluating the distance measurement capabilities of publisher and subscriber smartphones in the Auburn University Shelby Center Hallway. For each test, the smartphones were positioned precisely 5 meters apart, and the WiFi Round-Trip Time (RTT) distance measurement technique was employed. During each experiment, a set of 10 RTT distance measurements were collected and recorded for analysis.

Figure 7.1 illustrates the plot of Average RTT measured distances against the corresponding experiment numbers. These experiments were conducted on different days and involved smartphones positioned 5 meters apart in various sections of the hallway. Distance measurements were quantified in millimeters.

Analysis of the plot reveals a standard deviation of 0.56 meters for the average RTT distances, indicating a moderate level of variation around the mean distance. Furthermore, the 95^{th} percentile confidence interval of 0.49 meters reflects a high level of confidence in the reliability of the RTT measurements. This narrow range suggests that the actual average RTT distance is likely to fall within this interval with a high degree of predictability.

However, it is important to note that experiments $1, 2$, and 5 exhibit a relatively higher error rate in the Average RTT distance measurements. These observations can be attributed to WiFi multipath issues, as these experiments were conducted in the corners of the hallway where the smartphones were surrounded by walls. The presence of walls introduces reflections and interference, leading to a less accurate estimation of the RTT distance in these specific scenarios.

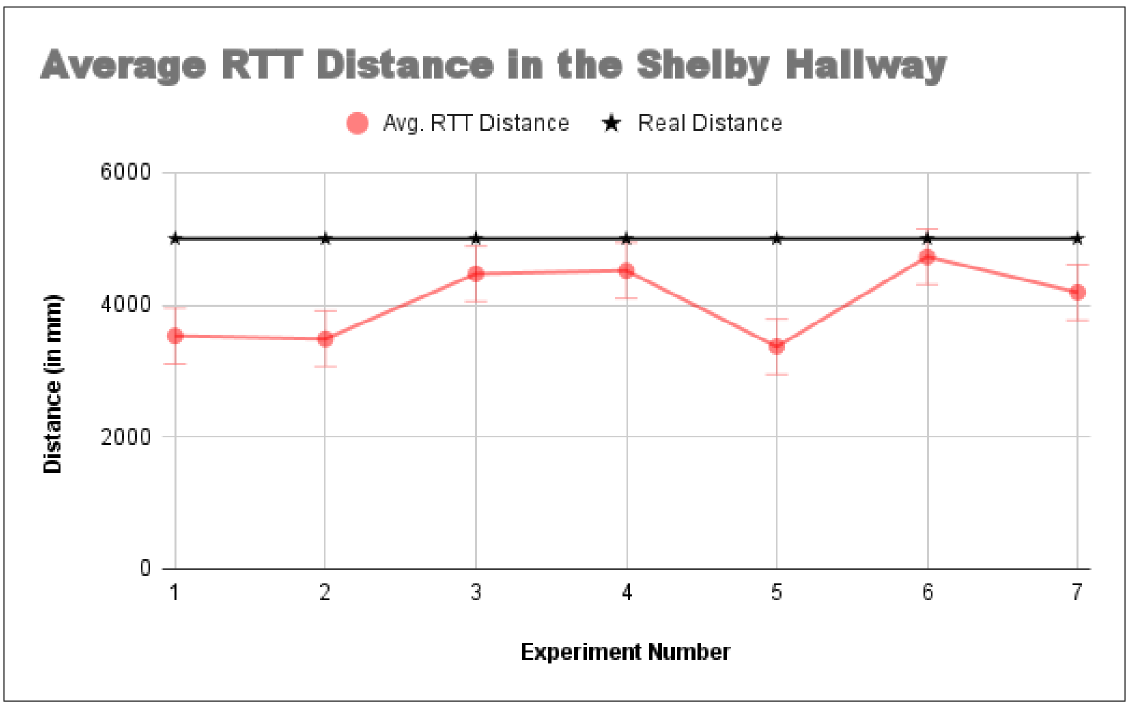

Figure 7.1: Auburn University Shelby Center Hallway Line-of-Sight (LOS) RTT measurement of smartphones 5 meters apart with 95% confidence interval

The findings from Figure 7.1 are further supported by the results depicted in Figure 7.2, which presents a similar plot with smartphones positioned 10 meters apart in the Auburn University Shelby Center Hallway.

In this case, the standard deviation of the average RTT distances, as shown in the plot, is calculated to be 0.54 meters. This indicates a comparable level of variability to the measurements taken at 5 meters distance. Similarly, the 95^{th} percentile confidence interval is determined to be 0.40 meters, reflecting a high level of confidence and predictability in the RTT measurements, aligning closely with the 5-meter RTT distance measurements.

These consistent statistical measures across the two distances, 5 meters and 10 meters, provide further assurance of the reliability and accuracy of the RTT measurements for distance estimation. The confidence level in the RTT measurements remains consistently high regardless of the increased distance between the smartphones, supporting the robustness of the WiFi RTT technique in indoor distance measurement scenarios.

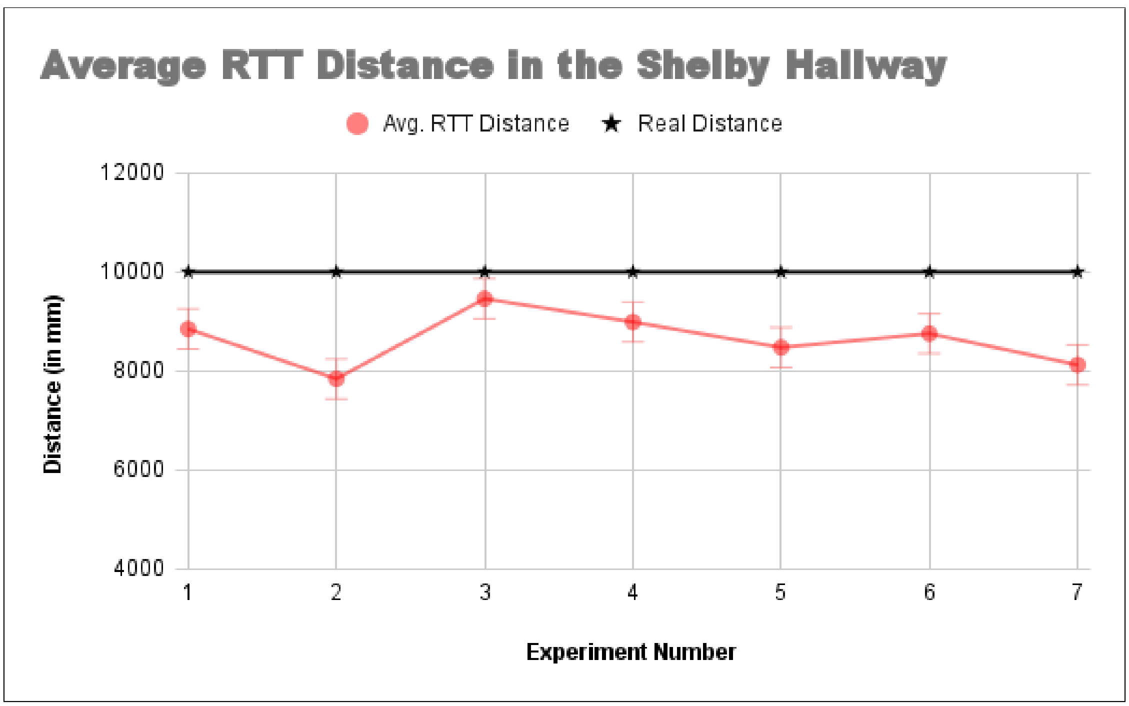

Figure 7.2: Auburn University Shelby Center Hallway Line-of-Sight (LOS) RTT measurement of smartphones 10 meters apart with 95% confidence interval

Figure 7.3 and Figure 7.4 depict plots similar to the previously discussed figures, but with measurements taken at distances of 15 meters and 20 meters, respectively.

In the plot representing the 15-meter distance measurement (Figure 7.3), the standard deviation of the average RTT distances is calculated to be 0.62 meters. This value indicates a slightly higher level of variation compared to the previous distances. Furthermore, the 95^{th} percentile confidence interval for this plot is determined to be 0.46 meters, emphasizing the predictability and reliability of the RTT measurements within this range.

Moving to the 20-meter distance measurement (Figure 7.4), the standard deviation of the average RTT distances is found to be 0.47 meters. This value suggests a relatively lower level of variation compared to the 15-meter distance. Similarly, the 95^{th} percentile confidence interval is calculated to be 0.35 meters, indicating a high level of confidence in the accuracy and reliability of the RTT measurements within this interval.

These results demonstrate that as the distance between smartphones increases, there is a slight increase in the variability of the average RTT distances. Nevertheless, the confidence intervals remain relatively narrow, signifying the consistency and reliability of the RTT measurements even at larger distances.

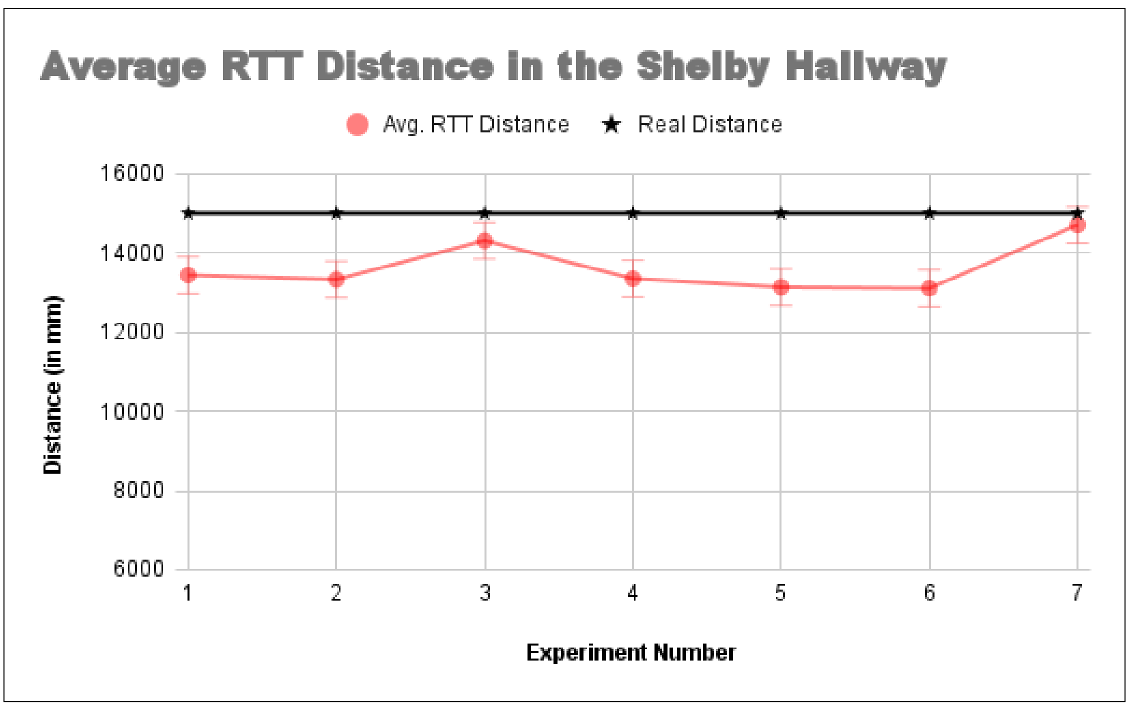

Figure 7.3: Auburn University Shelby Center Hallway Line-of-Sight (LOS) RTT measurement of smartphones 15 meters apart with 95% confidence interval

Figure 7.5 presents a comprehensive comparison between the Average Round-Trip Time (RTT) distance measurements and the actual distances of the smartphones at 5, 10, 15, and 20

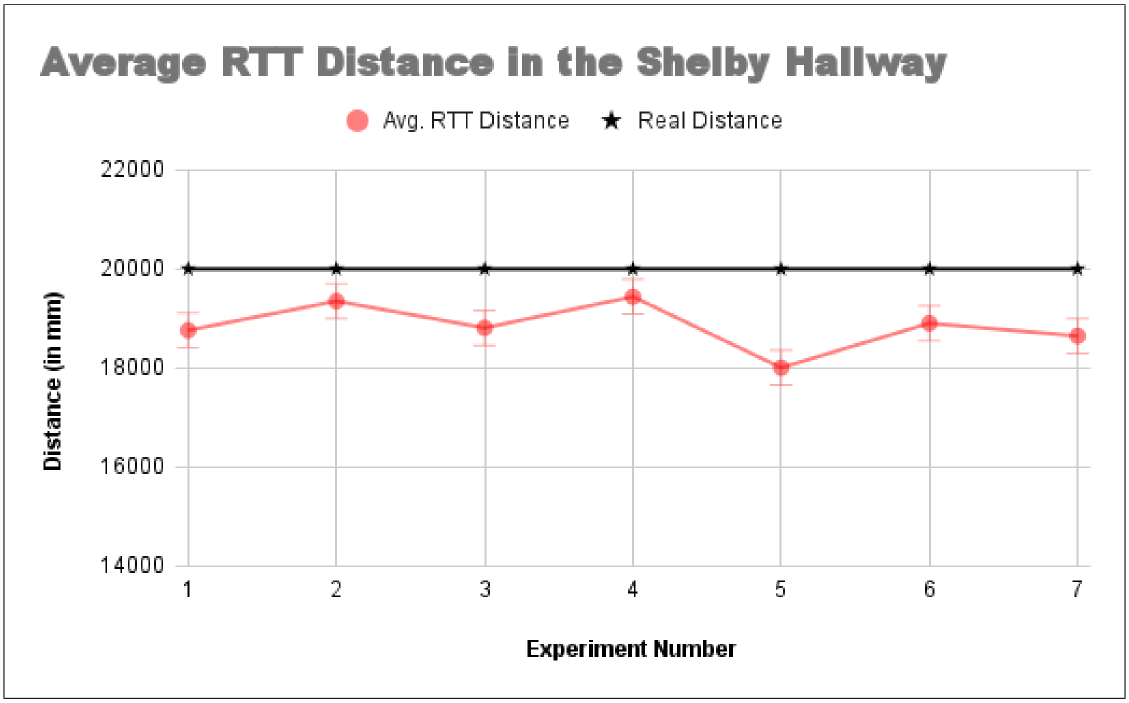

Figure 7.4: Auburn University Shelby Center Hallway Line-of-Sight (LOS) RTT measurement of smartphones 20 meters apart with 95% confidence interval

meters. This plot enables a direct evaluation of the consistency and accuracy of the RTT distance measurements.

Upon careful analysis, it becomes evident that the average RTT distance measurements closely align with the real distances, as depicted by the linear line in the plot. This linear relationship indicates that the average RTT distance measurements provide a reliable estimation of the actual distances between the smartphones.

Furthermore, the standard deviations for the average RTT distances at each of the four distances are similar. Specifically, the standard deviations are calculated to be 0.41, 0.40, 0.46, and 0.35 meters for the 5, 10, 15, and 20-meter distances, respectively. These consistent standard deviations across different distances affirm the reliability and precision of the RTT distance measurements.

An additional observation from the data is that the error rate in the RTT distances is relatively higher at lower distances between the smartphones compared to the error rate at greater distances. This discrepancy can be attributed to the impact of multipath errors. At shorter distances, the influence of multipath errors is more pronounced, leading to a higher error rate in the RTT distance measurements. In contrast, at greater distances, the multipath errors become less significant, resulting in a lower error rate in the RTT distance measurements.

In summary, the findings from this analysis highlight the consistency, accuracy, and robustness of the average RTT distance measurements, as they closely align with the actual distances. The similar standard deviations across different distances further reinforce the reliability of the RTT measurements. Additionally, the observations regarding the error rates shed light on the impact of multipath errors on RTT distance estimation, emphasizing the need for careful consideration of distance and environmental factors in such measurements.

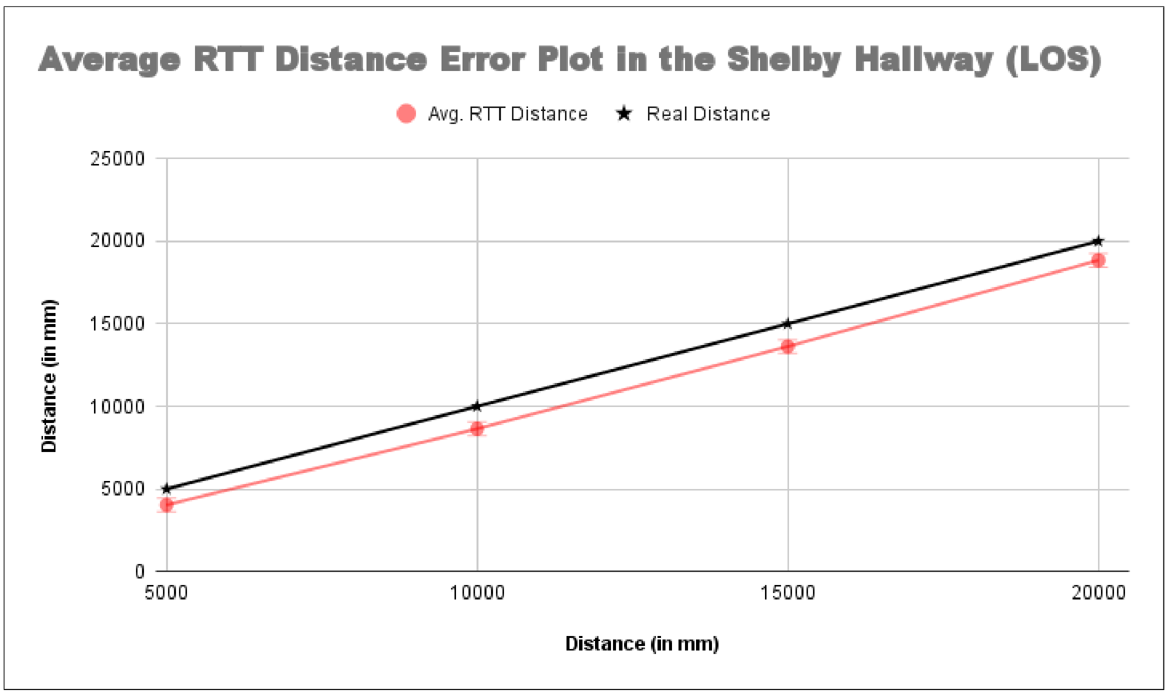

Figure 7.5: Auburn University Shelby Center Hallway Line-of-Sight (LOS) RTT measurement of all the distances with 95% confidence interval

The second phase of real-world testing took place in the room 2323 of the Auburn University Shelby Center, focusing on measuring the distance between the publisher and subscriber smartphones. To ensure consistency, the smartphones were handheld precisely 5 meters apart, and the WiFi Round-Trip Time (RTT) distance measurement technique was employed.

The data collection process followed the same methodology as the initial testing conducted in the hallway. This approach ensured a standardized approach to gathering the necessary data for analysis and comparison. By maintaining uniformity in the data collection procedure, it became possible to assess the performance of the WiFi RTT distance measurement technique in a different indoor environment.

Overall, the second testing phase aimed to provide additional insights into the accuracy and reliability of the WiFi RTT distance measurements. By conducting the experiments in room 2323, the study sought to explore any variations or similarities in the results obtained compared to the previous hallway testing scenario.

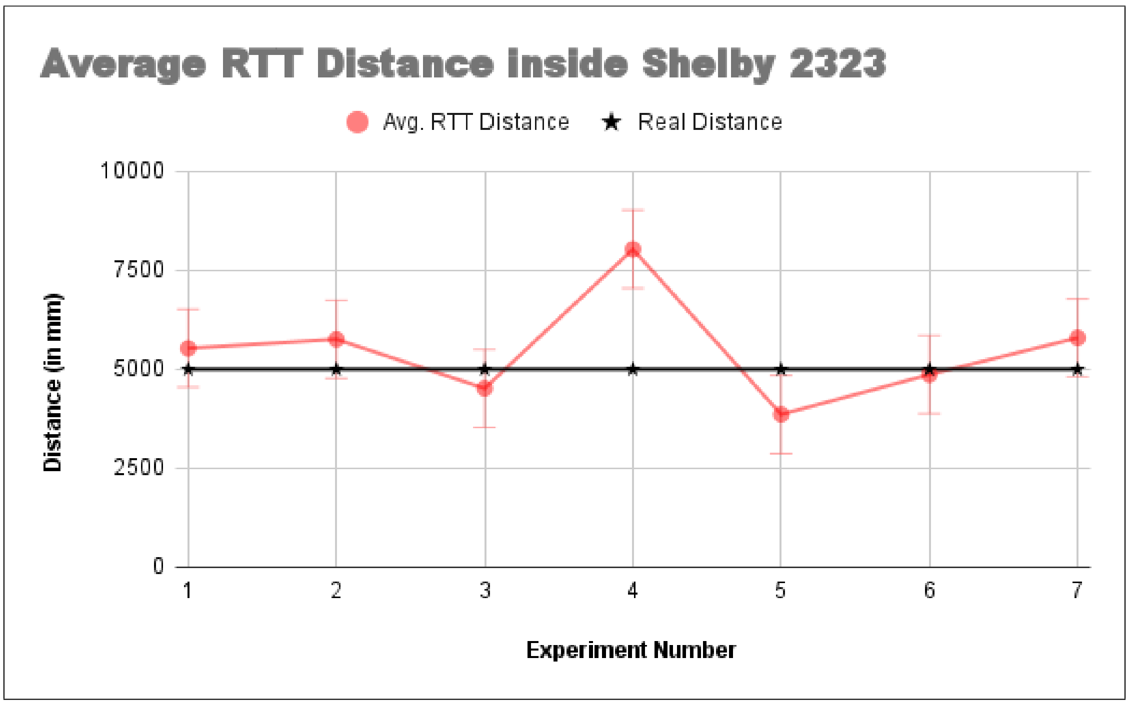

Figure 7.6: Auburn University Shelby Center inside room Line-of-Sight (LOS) RTT measurement of smartphones 5 meters apart with 95% confidence interval

Figure 7.6 provides a detailed visualization of the Average Round-Trip Time (RTT) measured distance plot against different experiment numbers. The plot showcases the measurements obtained from smartphones positioned 5 meters apart in various sections of the room and conducted on different days. The distances, accurately recorded in millimeters, serve as crucial indicators of the signal propagation characteristics in the environment.

Analyzing the plot, we observe that the average RTT distances display a certain level of variation, as indicated by the standard deviation of 1.33 meters. This statistical measure suggests that the average RTT values tend to deviate from the mean distance by approximately 1.33 meters. However, despite this variability, the 95^{th} percentile confidence interval of 0.98 meters instills confidence in the reliability of the RTT measurements. This means that there is a high level of certainty that the actual average RTT distance falls within this narrow range.

It is worth noting that experiment 4 stands out with a relatively higher error rate in the RTT distance measurements. This occurrence can be attributed to the presence of WiFi multipath issues. In this specific experiment, the smartphones were positioned in the corner of the room, surrounded by walls. The walls could have caused reflections and interference, leading to an increased error rate in the RTT measurements. This finding highlights the impact of environmental factors on wireless signal propagation and emphasizes the importance of considering such factors in the analysis and interpretation of RTT data.

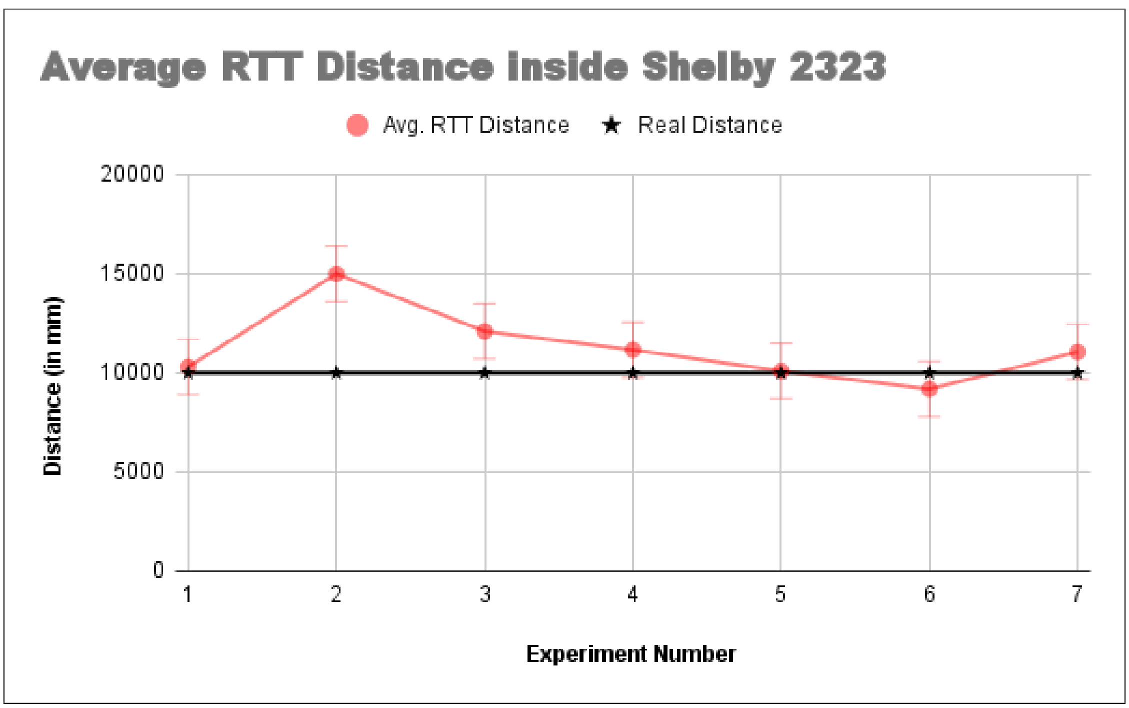

Figure 7.7: Auburn University Shelby Center inside room Line-of-Sight (LOS) RTT measurement of smartphones 10 meters apart with 95% confidence interval

Figure 7.7 provides a plot similar to Figure 7.6, but with the smartphones positioned 10 meters apart within the room. This plot enables a comparison of the Average Round-Trip Time (RTT) measured distances at different distances, shedding light on the reliability and consistency of the RTT measurements in the specific environment.

Analyzing the plot, we find that the standard deviation of the average RTT distances is calculated to be 1.88 meters. This value indicates a moderate level of variability in the RTT measurements, similar to that observed at the 5-meter distance. Additionally, the 95^{th} percentile confidence interval is determined to be 1.39 meters, suggesting a high level of confidence in the accuracy and reliability of the RTT measurements within this interval.

However, it is important to note that experiment 2 exhibits a relatively higher error rate in the Average RTT distance measurements. This can be attributed to the WiFi multipath issue encountered during this specific experiment. The smartphones were positioned diagonally across the room, near the corner where both devices were surrounded by walls. The presence of walls and the resulting multipath interference likely contributed to the higher error rate observed in this scenario.

Overall, the findings from this analysis support the predictability and reliability of the RTT measurements, even at a distance of 10 meters. The standard deviation and confidence interval values align with those observed at the 5-meter distance, suggesting the consistent performance of the WiFi RTT technique in the indoor environment. However, it is crucial to consider environmental factors, such as multipath interference, as they can impact the accuracy of the RTT distance measurements in specific scenarios.

Figure 7.8 presents a plot similar to Figure 7.6, but with the smartphones positioned 5 meters apart in a different room within the Shelby Center. This comparison allows for an assessment of the Average Round-Trip Time (RTT) measured distances in a distinct indoor environment.

Analyzing the plot, we find that the standard deviation of the average RTT distances is calculated to be 0.85 meters. This value indicates a slightly higher level of variability compared to the RTT distance measurements in room 2323. However, the 95^{th} percentile confidence interval of

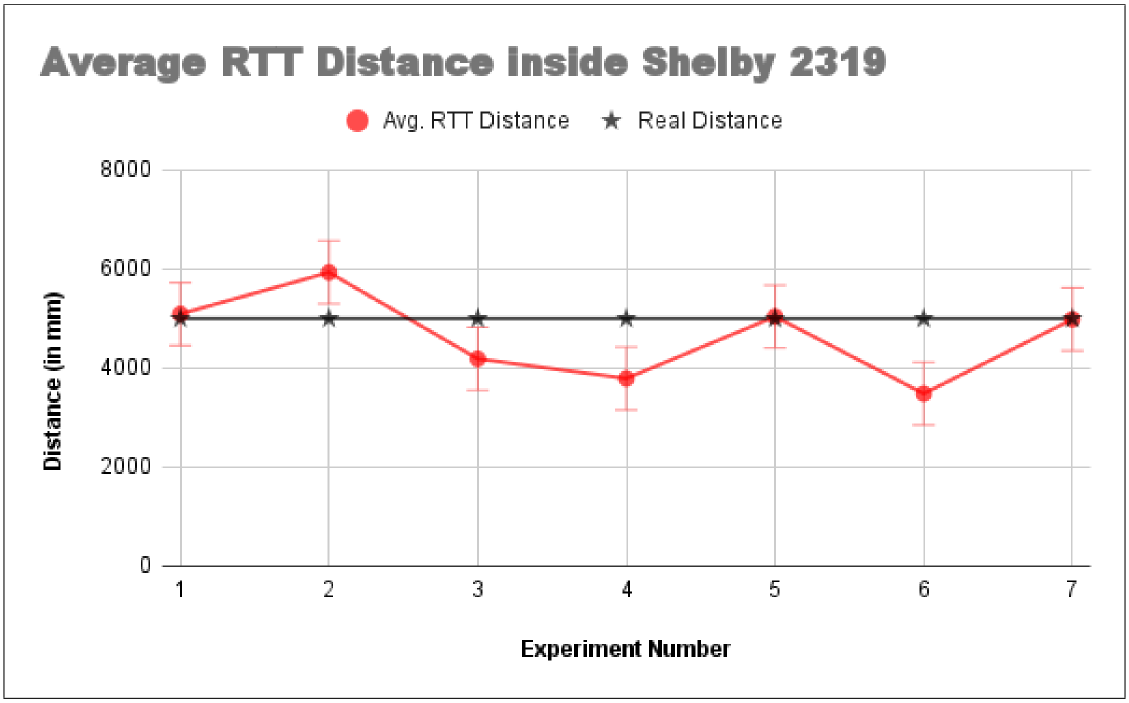

Figure 7.8: Auburn University Shelby Center inside room Line-of-Sight (LOS) RTT measurement of smartphones 5 meters apart with 95% confidence interval

0.63 meters reflects a high level of confidence in the accuracy and reliability of the RTT measurements within this interval.

Remarkably, the RTT measurements in room 2319 exhibit better performance compared to the measurements conducted in room 2323. This improvement may be attributed to the reduced clutter and obstacles present in room 2319. The lesser number of obstructions in the environment likely contributes to a more reliable and consistent signal propagation, resulting in improved RTT distance measurements.

These findings highlight the predictability and reliability of the RTT measurements conducted in room 2319, with a lower standard deviation and narrower confidence interval compared to the measurements in room 2323. The influence of environmental factors, such as clutter and obstacles, is underscored, emphasizing the significance of the physical environment in determining the accuracy and performance of the RTT distance measurements.

The third phase of real-world testing was carried out in the transitional area between the hallway and room 2323 of the Auburn University Shelby Center. The primary objective was to measure the distance between the publisher and subscriber smartphones in this specific setup. To ensure accuracy and consistency, the smartphones were handheld precisely 6 meters apart, with one smartphone positioned in the hallway and the other inside room 2323. The WiFi Round-Trip Time (RTT) distance measurement technique was employed to collect the necessary data.

The data collection process for this testing phase followed the same methodology as the previous experiments, ensuring uniformity and comparability across the different scenarios. By maintaining consistency in the data collection approach, the study aimed to evaluate the performance and reliability of the WiFi RTT distance measurements in this particular transitional environment.

Overall, this phase of testing provided valuable insights into the accuracy and effectiveness of the WiFi RTT distance measurements when the smartphones were located in the transitional space between the hallway and room 2323. The collected data served as a basis for analysis and comparison with the previous testing scenarios, allowing us to assess any variations or similarities in the performance of the RTT measurements in this unique setting.

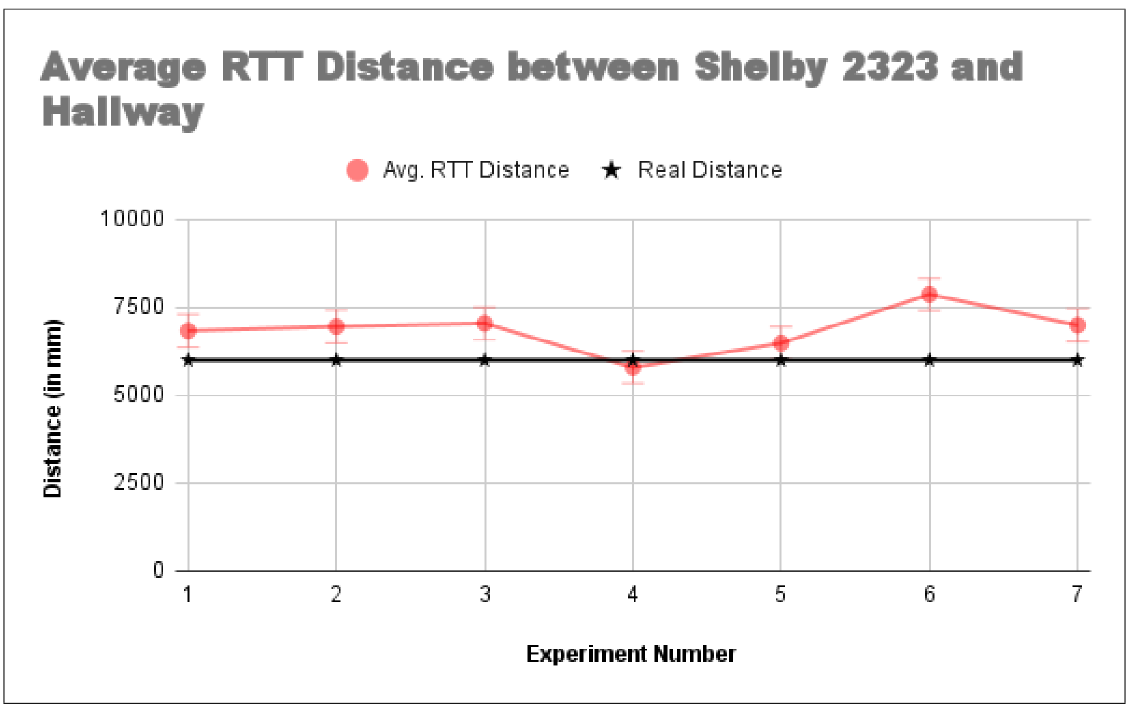

Figure 7.9: Auburn University Shelby Center between the hallway and room Line-of-Sight (LOS) RTT measurement of smartphones 6 meters apart with 95% confidence interval

Figure 7.9 presents an Average Round-Trip Time (RTT) measured distance plot against the experiment number, depicting the results obtained from smartphones positioned 6 meters apart in different areas and on different days. In this setup, the smartphones were in Line-of-Sight (LOS) with each other, with the line of sight passing through the door.

The plot provides valuable insights into the RTT measurements performed in this scenario. The measured distances, recorded in millimeters, serve as crucial indicators of the signal propagation characteristics within the environment. Analyzing the plot, we find that the standard deviation of the average RTT distances is calculated to be 0.62 meters. This value suggests a moderate level of variability in the RTT measurements.

Furthermore, the 95^{th} percentile confidence interval, determined to be 0.46 meters, indicates a high level of confidence in the accuracy and reliability of the RTT measurements within this range. These findings emphasize that the RTT measurements are predictably reliable, bolstering confidence in the accuracy of the distance measurements obtained using the WiFi RTT technique in this LOS setup. The combination of a relatively low standard deviation and a narrow confidence interval affirms the consistent and dependable nature of the RTT measurements.

Figure 7.10 illustrates a comparable plot to Figure 7.9, with smartphones positioned 6 meters apart in non-line-of-sight (NLOS) areas. Unlike the LOS setup, in this scenario, the smartphones were required to communicate through walls to measure the Round-Trip Time (RTT) distances accurately. Analyzing the plot, we observe that the standard deviation of the average RTT distance is calculated to be 1.46 meters. This value indicates a higher level of variability in the RTT measurements compared to the LOS setup. Additionally, the wider 95^{th} percentile confidence interval of 1.17 meters suggests a slightly lower level of confidence in the accuracy and reliability of the RTT measurements within this range, in comparison to the LOS measurements.

These findings highlight the impact of obstacles, such as walls, on the reliability and accuracy of the RTT measurements. The requirement for smartphones to communicate through walls, as opposed to thin and clear doors, introduces additional signal attenuation and multipath effects,

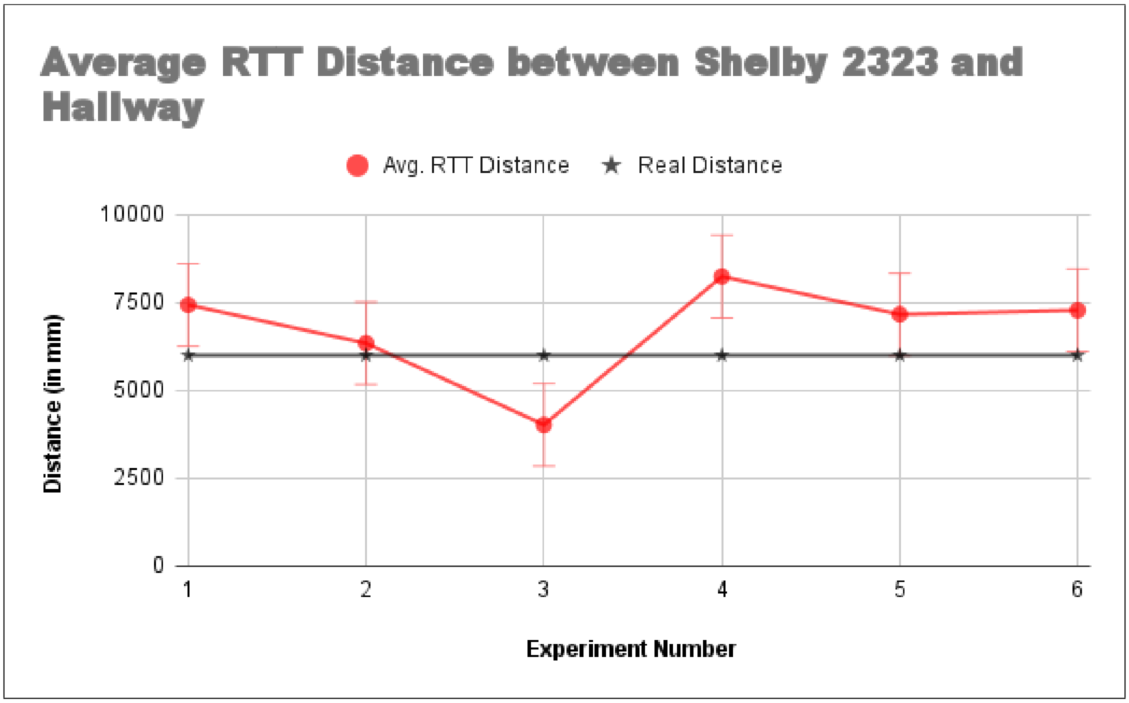

Figure 7.10: Auburn University Shelby Center between the hallway and room Non Line-of-Sight (NLOS) RTT measurement of smartphones 6 meters apart with 95% confidence interval

which contribute to the increased variability in the RTT measurements. Consequently, the reliability and accuracy of the RTT distance measurements experience a downgrade in the NLOS scenario compared to the LOS setup. Nevertheless, despite the challenges posed by NLOS conditions, the RTT measurements remain predictably reliable, and there is still a high level of confidence in their accuracy.

In the fourth phase of real-world testing, the distance between the publisher and subscriber smartphones was measured within room 2323 of the Auburn University Shelby Center. For this particular experiment, the smartphones were handheld precisely 6 meters apart while both individuals were seated on chairs. The WiFi Round-Trip Time (RTT) distance measurement technique was employed to capture the necessary data. The primary objective of this experiment was to evaluate the performance and error rate of the RTT distance measurement when the smartphones were positioned closer to the ground. By conducting the measurements while seated, we aimed to simulate a scenario where users interact with their smartphones in a seated position, potentially affecting the accuracy of the distance measurements.

The data collection process followed the same methodology as the previous testing, ensuring consistency and comparability across the different experimental setups. By conducting the measurements in room 2323 and maintaining consistency in the data collection approach, we could assess any variations or similarities in the performance of the RTT measurements in this specific setting.

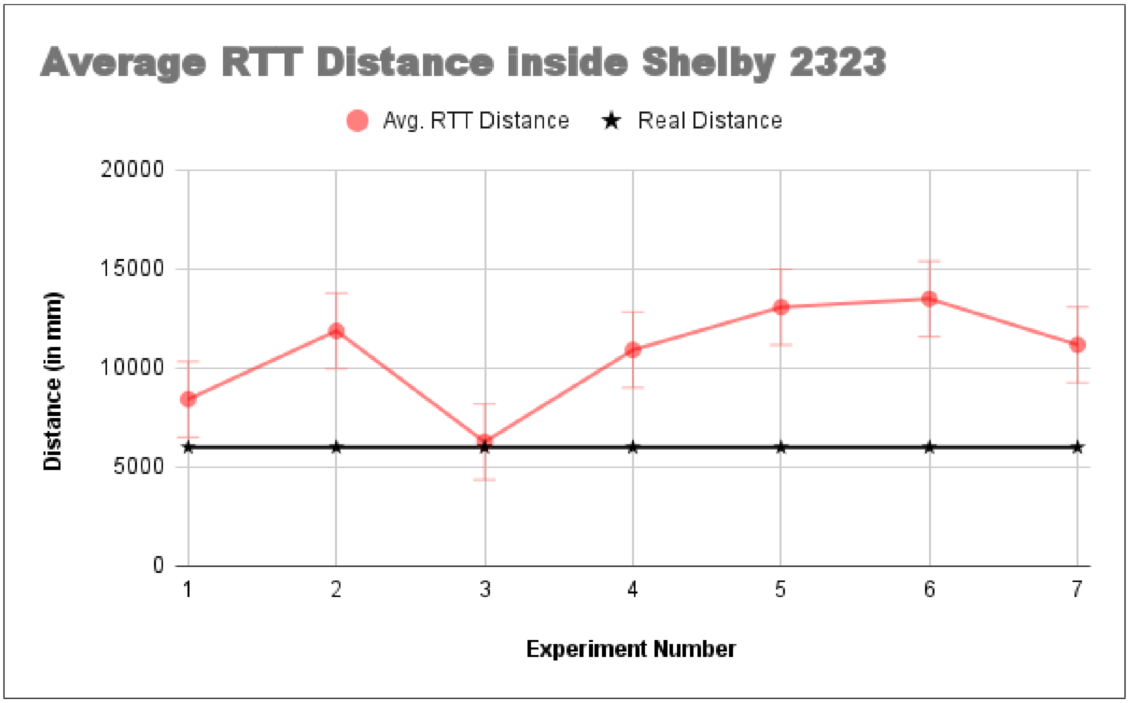

Figure 7.11: Auburn University Shelby Center inside room Line-of-Sight (LOS) with both phones held while sitting on a chair RTT measurement of smartphones 6 meters apart with 95% confidence interval

Figure 7.11 presents an Average Round-Trip Time (RTT) measured distance plot against the experiment number. Each experiment represents the measurement of smartphones placed 6 meters apart in various sections of the room on different days. It's important to note that the smartphones were positioned in a Line-of-Sight (LOS) configuration, meaning there were no significant obstructions between them. The distances were accurately measured in millimeters, serving as crucial data points to assess the signal propagation characteristics within the environment.

Upon analyzing the plot, we observe that the standard deviation of the average RTT distance is calculated to be 2.57 meters. This value indicates a substantial level of variability in the RTT measurements. Furthermore, the wider 95^{th} percentile confidence interval of 1.9 meters suggests a relatively lower level of confidence in the accuracy and reliability of the RTT measurements within this range.

These findings indicate that the RTT measurements in this setup are predictably unreliable, and there is a relatively lower level of confidence in the accuracy of the distance measurements obtained using the WiFi RTT technique. The significant variability in the measurements and the wide confidence interval highlight the challenges and limitations encountered in accurately capturing the distance between smartphones placed 6 meters apart when the smartphones are close to the ground indicating a higher level of interference from the multipath error.

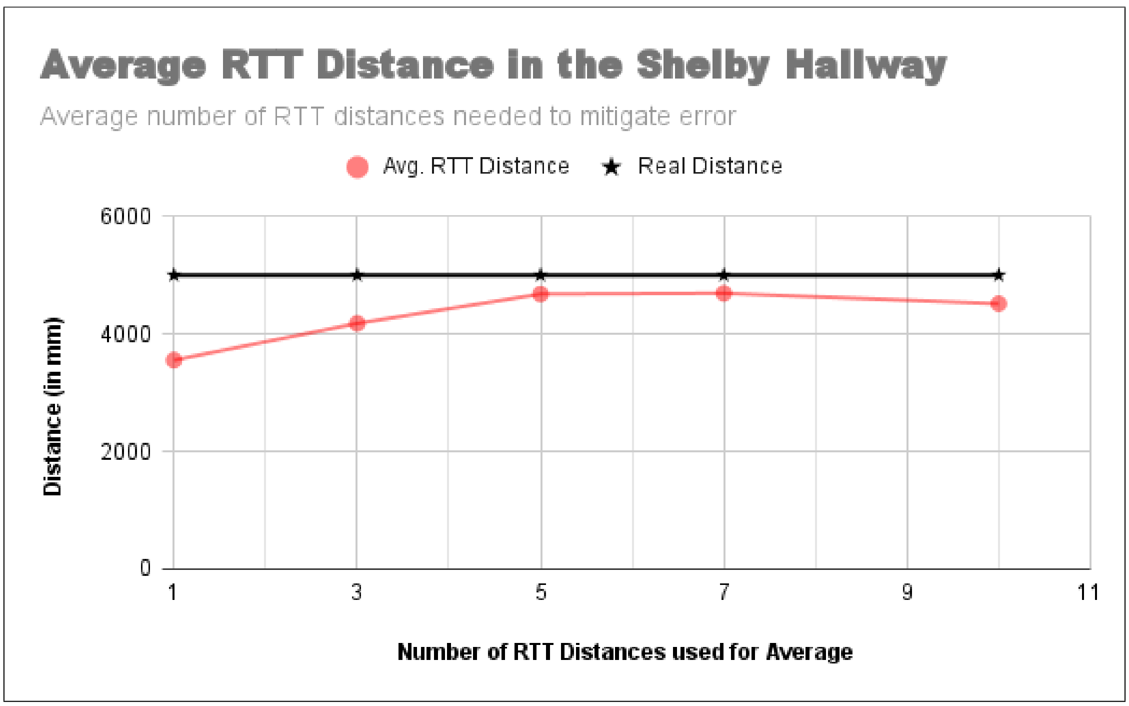

Figure 7.12: Average RTT measurement of smartphones 5 meters apart in Auburn University Shelby center hallway with varying number of RTT distances used for Averaging result

Figure 7.12 illustrates a graph depicting the impact of different numbers of raw Round-Trip Time (RTT) distances on the average RTT distance between smartphones. The purpose of this experiment was to determine the optimal number of raw RTT distance measurements required to achieve an acceptable level of accuracy while considering the system's performance.

The data collection process followed the same methodology as the previous testing, ensuring consistency and comparability. Figure 7.12 demonstrates that the average RTT distance between smartphones exhibits improved accuracy as the number of raw RTT distance measurements increases. By increasing the number of measurements, the system can achieve a more reliable and accurate estimation of the distance between the smartphones. This finding is significant as it contributes to enhancing the precision of location computation for smartphones using the grid method algorithm.

Through analysis, it was determined that collecting 10 raw RTT distances between smartphones and subsequently computing the average RTT distance provides a solid foundation for accurate distance computation. This finding aids in refining the accuracy of location estimation for smartphones, enabling more reliable and precise tracking using the grid method algorithm.

Overall, this experiment highlights the importance of collecting an optimal number of raw RTT distance measurements to improve the accuracy of distance computation between smartphones. By identifying the ideal number of measurements, we can enhance the performance and reliability of the system, enabling more accurate localization and facilitating the effective implementation of location-based services.

The fifth real-world testing took place in the Auburn University Shelby Center hallway to evaluate the distance between the publisher and subscriber smartphones. In this experiment, the smartphones were held exactly 20 meters apart, and three smartphones were positioned in the hallway in a Line-of-Sight (LOS) configuration. The WiFi Round-Trip Time (RTT) distance measurement technique was utilized to capture the distances accurately.

Within this setup, two publisher smartphones were placed on opposite sides of the hallway, while the subscriber smartphone was located in the middle, as depicted in Figure 6.1. The objective of this experiment was to assess the performance and error rate of the RTT distance measurement when the subscriber smartphone simultaneously measured the RTT distances to both publisher smartphones.

The data collection process for this experiment followed the same methodology as the previous testing, ensuring consistency and comparability in the results obtained. By employing the WiFi RTT technique, the experiment aimed to gather accurate distance measurements and analyze the performance of the system in this specific scenario. The experiment yielded valuable insights into the performance and error rate of the RTT distance measurement when the subscriber smartphone was tasked with measuring the distances to both publisher smartphones simultaneously.

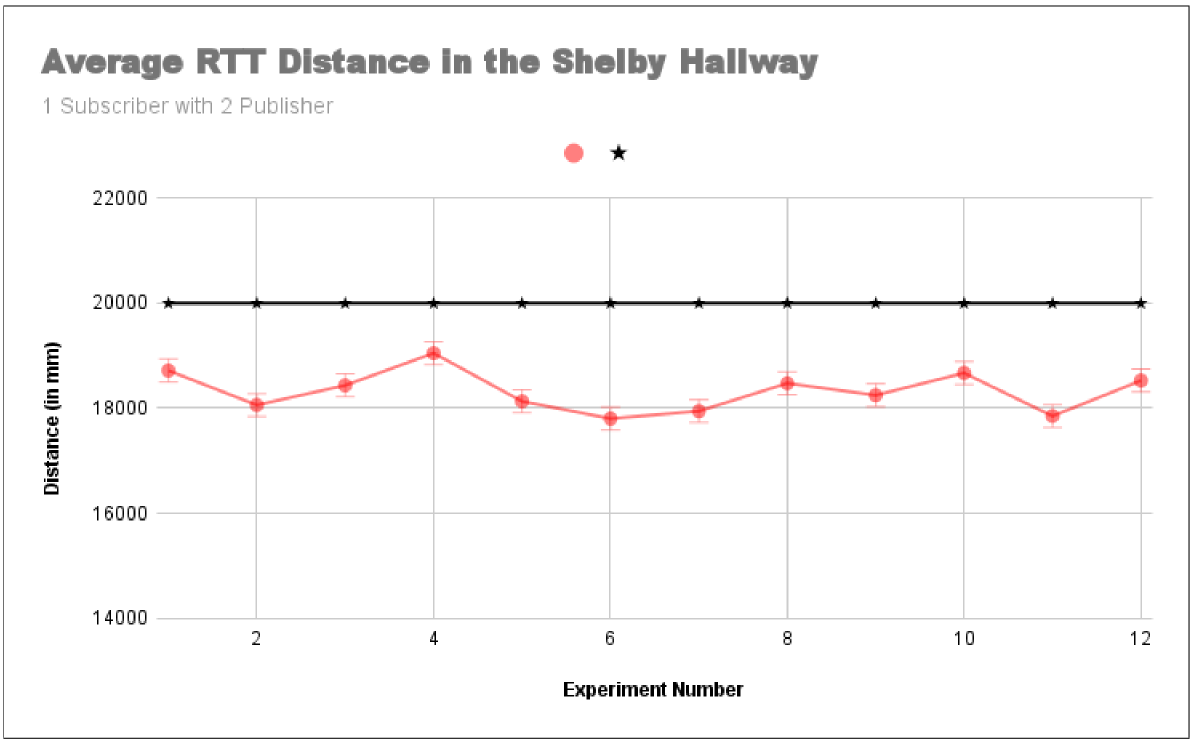

Figure 7.13: Average RTT measurement of smartphones 20 meters apart in Auburn University Shelby center hallway with simultaneous RTT communication with two publisher smartphones

Figure 7.13 provides a graph depicting the Average Round-Trip Time (RTT) distances measured simultaneously by the subscriber smartphone from two publisher smartphones in the Auburn University Shelby Center hallway. The data collection process for this graph followed the same methodology as the previous testing, which resulted in the collection of RTT distances showcased in Figure 7.4.

The distances in the graph are measured in millimeters. The standard deviation of the average RTT distances shown in the plot is 0.38 meters, and the 95^{th} percentile confidence interval is

0.21 meters. Figure 7.13 presents the average RTT distances obtained from a total of 12 experiments. The first six experiments correspond to measurements from publisher number 1, while the remaining six experiments correspond to measurements from publisher number 2.

Notably, Figure 7.13 reveals that the average RTT distances achieved in this simultaneous measurement scenario demonstrate a similar level of performance and accuracy when compared to the experiment involving a single publisher and subscriber smartphone measurement, as depicted in Figure 7.4. These findings suggest that conducting simultaneous RTT distance measurements from multiple publisher smartphones does not significantly impact the performance or accuracy of the average RTT distances obtained. The comparable results obtained from both scenarios indicate that the system is capable of maintaining consistent and reliable measurements, even when measuring distances from multiple sources simultaneously.

This approach ensures the reliability and validity of the obtained results and allows for meaningful comparisons and analysis of the system's performance in multi-publisher scenarios.

7.2 Evaluating the effectiveness of Smartphone WiFi FTM based on RTT Location Measurement

In this section, we will thoroughly examine the accuracy of smartphone WiFi RTT-based location computation. Our approach involves utilizing the average WiFi RTT distances between Publisher and Subscriber smartphones, which serve as crucial input for an efficient grid method localization algorithm. By leveraging these components in conjunction, we aim to achieve precise and reliable location estimation. By carefully collecting and averaging these RTT distances, we can derive a more accurate representation of the actual physical distance between the devices.

To further enhance the accuracy of our location estimation, we employ an efficient grid method localization algorithm. This algorithm takes into account the measured RTT distances and employs advanced mathematical techniques to determine the precise location of the devices within a defined grid or coordinate system.

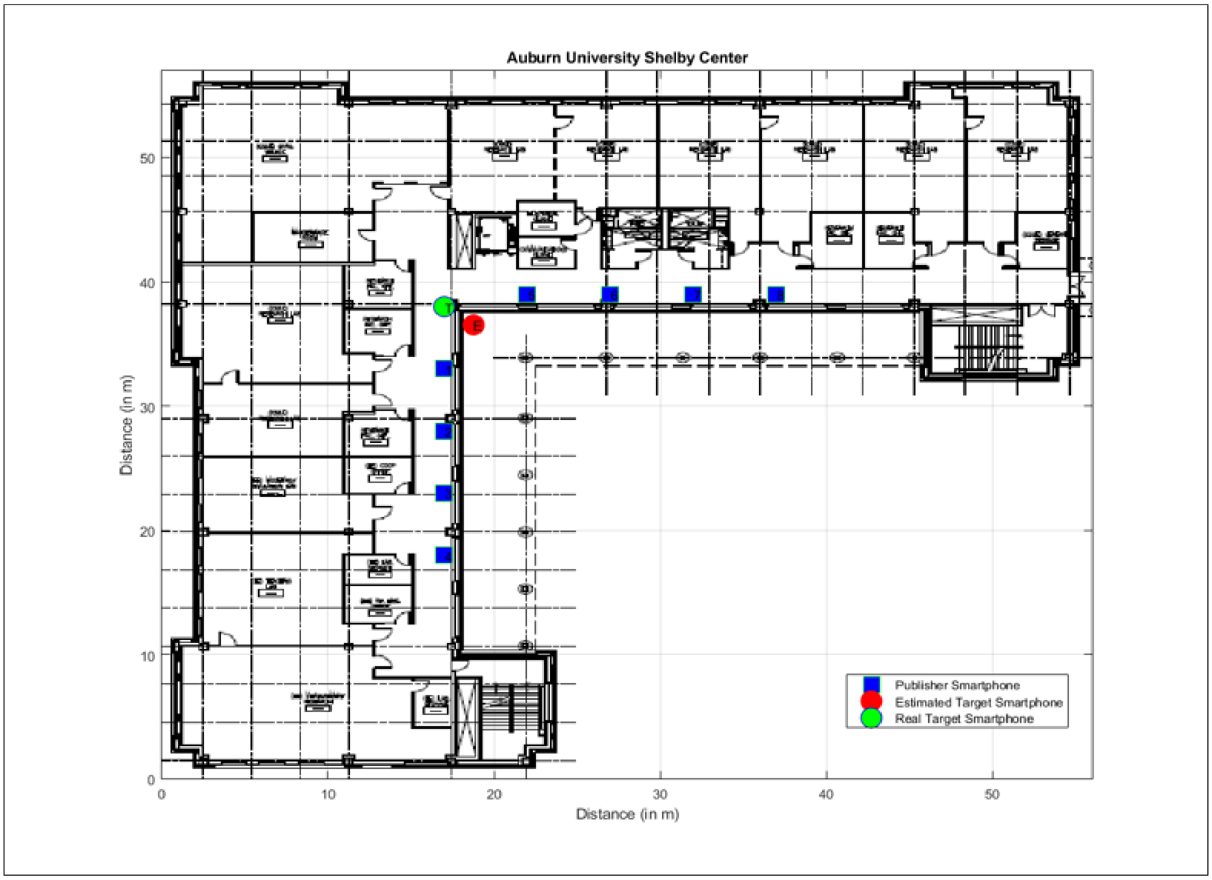

Figure 7.14: Auburn University Shelby Center Line-of-Sight location estimation using eight Reference nodes

The initial real-world experiment took place in the Auburn University Shelby Center Hallway, involving the deployment of publisher and subscriber smartphones at distances of 5, 10, 15, and 20 meters apart. The RTT distances between all publishers and the subscriber were captured, enabling the subscriber to estimate its own location using the sophisticated grid method discussed in Section 5.4.

Figure 7.14 presents a blueprint of the hallway, showcasing the positions of each publisher smartphone denoted by blue squares. Additionally, it illustrates both the actual and estimated location of the subscriber smartphone (referred to as the target smartphone). The accuracy of the estimated location, achieved through the utilization of smartphone WiFi RTT technology and the involvement of *eight* publisher smartphones, is measured at 2.28 meters. However, it should be noted that the accuracy of the estimated location was compromised in this scenario due to the specific arrangement of the publishers. As illustrated in Figure 7.14, all publisher smartphones were aligned in a singular direction, leading to a case of poor dilution of precision (PDoP), as explained in detail in Section 6.2. This alignment adversely affected the accuracy of the estimated location.

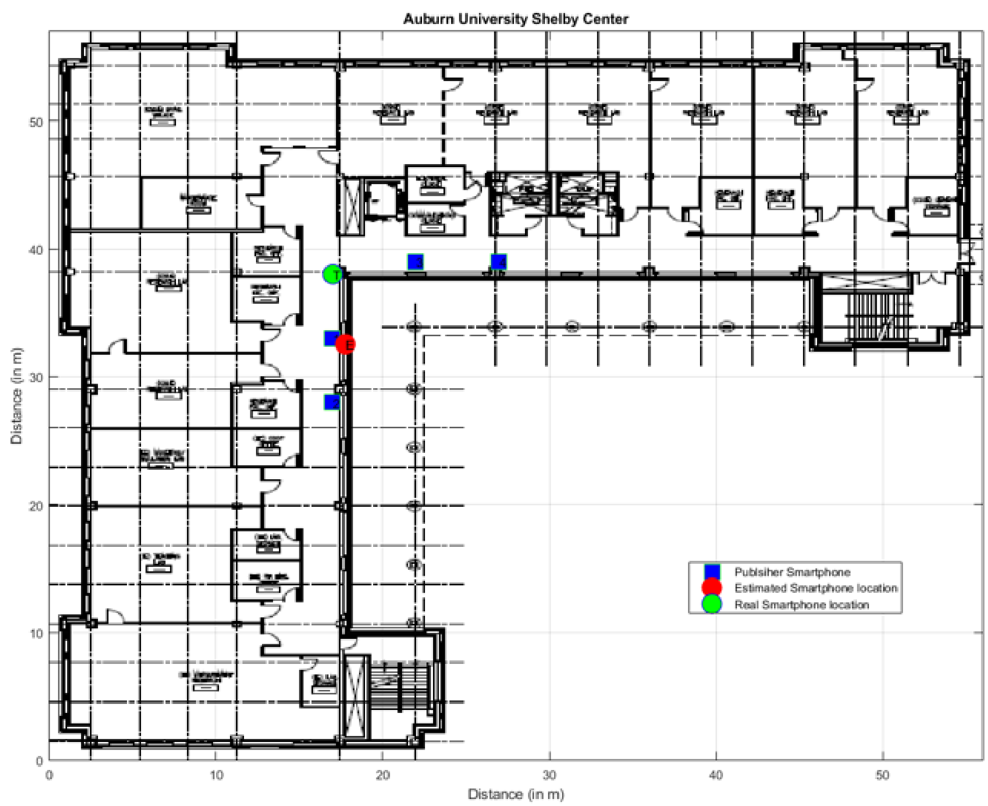

Figure 7.15: Auburn University Shelby Center Line-of-Sight location estimation using four Reference nodes

Figure 7.15 depicts the layout of an experiment where the subscriber smartphone device leverages the assistance of four publisher smartphones for localization using WiFi RTT technology alongside the grid method algorithm. Notably, the reduced number of publisher smartphones in this experiment leads to a significant degradation in location accuracy compared to the previous experiment shown in Figure 7.14.

In this setup, the accuracy of the estimated location is measured at 5.5 meters, a noticeable increase compared to the previous experiment's accuracy of 2.28 meters. The decline in accuracy can be attributed to the limited number of publisher smartphones available to assist in the localization process. With fewer reference points, the ability to precisely determine the subscriber smartphone's location is compromised.

The accuracy of the subscriber smartphone's location, determined through the utilization of WiFi RTT technology with the grid method algorithm, is influenced by two main factors: the accuracy of the RTT distances and the spatial distribution of the publisher smartphones within the area of interest. Despite the presence of an error rate of approximately 10% in the average RTT distance measurements between the publisher and subscriber, the resulting location accuracy remains within the range of under 3 meters.

However, it is worth noting that in the specific experiment conducted within the Auburn University Shelby Center hallway, the accuracy of the location estimation was adversely affected by the Dilution of Precision (DOP) error. This error arises from the spatial distribution of the publisher smartphones, which can lead to a reduction in the accuracy of the overall localization process.

The second real-world testing aimed to assess the performance of the WiFi RTT technology in different areas of the Auburn University Shelby Center, including Room 2323, Room 2319, and the hallway. For this testing, the publisher and subscriber smartphones were positioned at distances of $5, 10,$, and 15 meters from each other. The experiment area and setup are visualized in Figure 6.2, providing a clear overview of the testing environment.

Figure 7.16 showcases the results of the second real-world testing conducted in different areas of Auburn University Shelby Center, including Room 2323, Room 2319, and the hallway. The figure illustrates the locations of the publisher smartphones represented by blue squares, along with the real location of the subscriber smartphone (target smartphone) and its estimated location using WiFi RTT technology.

Notably, the accuracy of the estimated smartphone location with the presence of *six* publisher smartphones is remarkably high, measuring only 0.37 meters. This level of accuracy is achieved

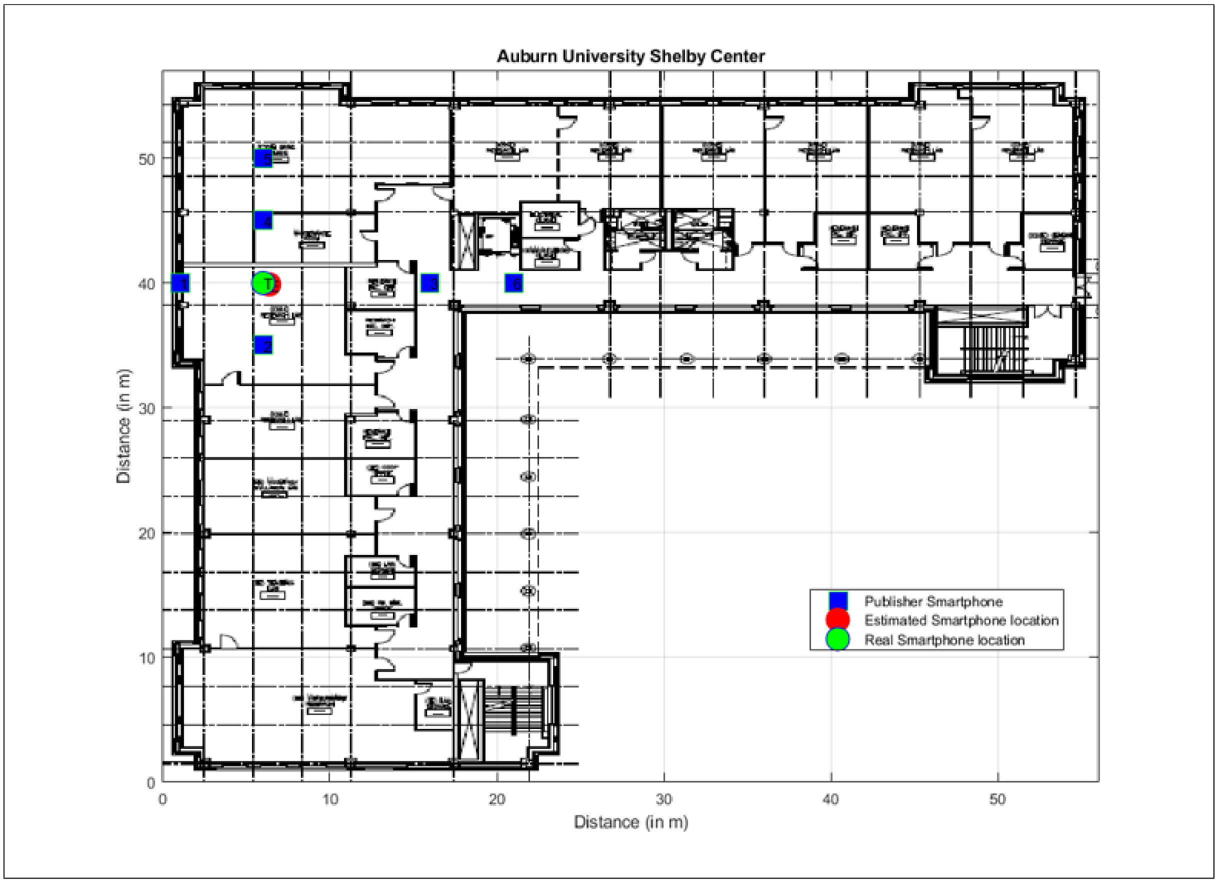

Figure 7.16: Auburn University Shelby Center location estimation using six Reference nodes with publisher and subscriber being in different non-Line-of-Sight (NLOS) areas

even in a Non-line-of-sight (NLOS) situation, where obstacles may hinder direct signal propagation. The figure highlights the effectiveness of the grid method algorithm aided by the distributed placement of the publisher smartphones, which mitigates the dilution of precision (DOP) error and contributes to the accurate location estimation.

Figure 7.17 presents the results of another experiment conducted in various areas of Auburn University Shelby Center, including Room 2323, Room 2319, and the hallway. The figure illustrates the locations of the publisher smartphones aiding the subscriber smartphone in estimating its location using WiFi RTT technology.

Despite having only *four* publisher smartphones in this experiment, the accuracy of the estimated smartphone location is still notable, measuring 3.89 meters. Compared to the previous experiment depicted in Figure 7.15, the accuracy is considerably improved. This improvement

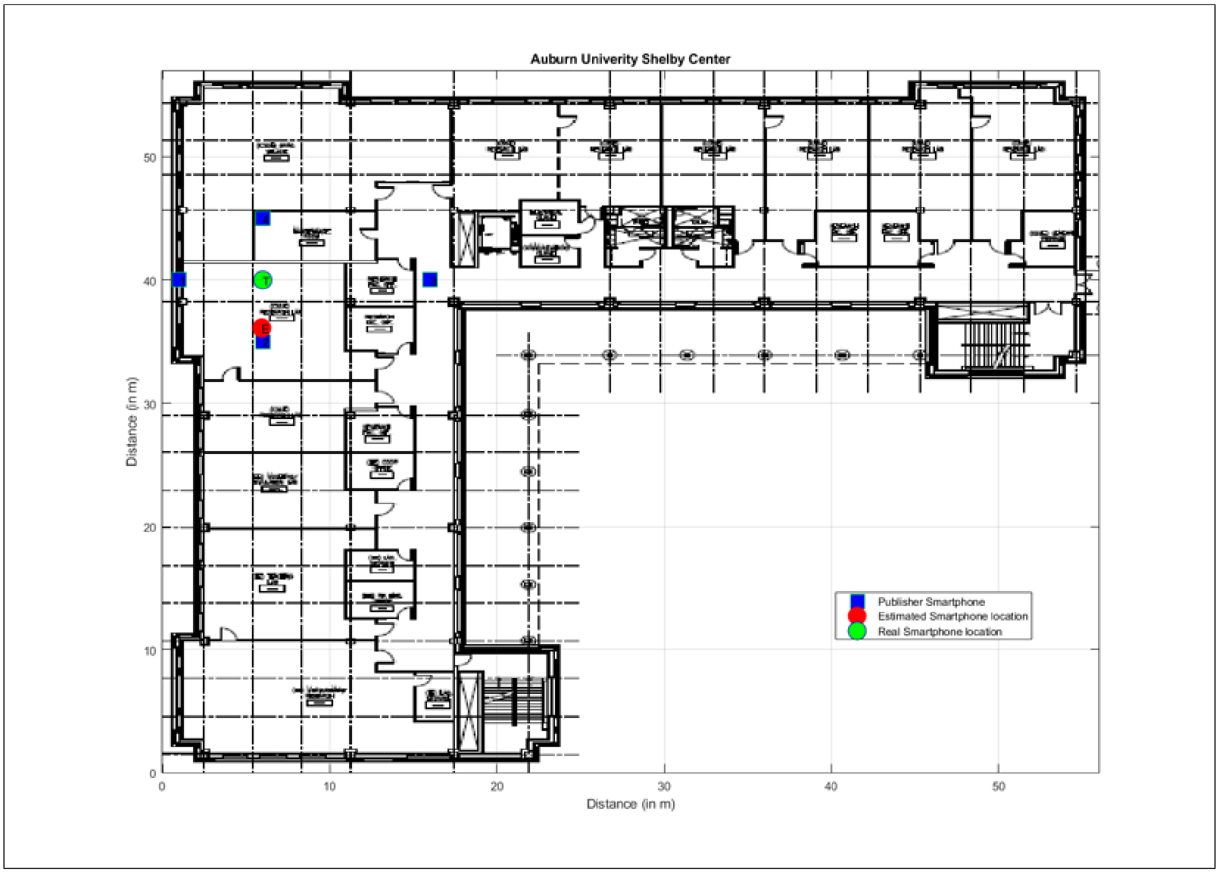

Figure 7.17: Auburn University Shelby Center location estimation using four Reference nodes with publisher and subscriber being in different non-Line-of-Sight (NLOS) areas

can be attributed to the effective distribution of the publisher smartphones around the subscriber smartphone, which helps minimize the dilution of precision (DOP) error. As a result, the location accuracy is enhanced even with a reduced number of publisher smartphones.

7.3 Evaluating the error propagation of Smartphone WiFi FTM based on RTT Location Measurement

In this section, we focus on analyzing the error propagation characteristics of our smartphone-based indoor localization system. Initially, we conduct a localization experiment involving a subscriber smartphone and five publisher smartphones that assist in the localization process. It's important to note that all five publisher smartphones serve as first-hop neighbors, meaning that their self-location information is error-free. Figure 7.18 visually represents the results, showcasing the high accuracy of our indoor localization technology with an estimated error of only 1.1 meters in the subscriber's location. Since there are no errors in the location information provided by the publisher smartphones, the resulting error in the subscriber smartphone's location is minimal.

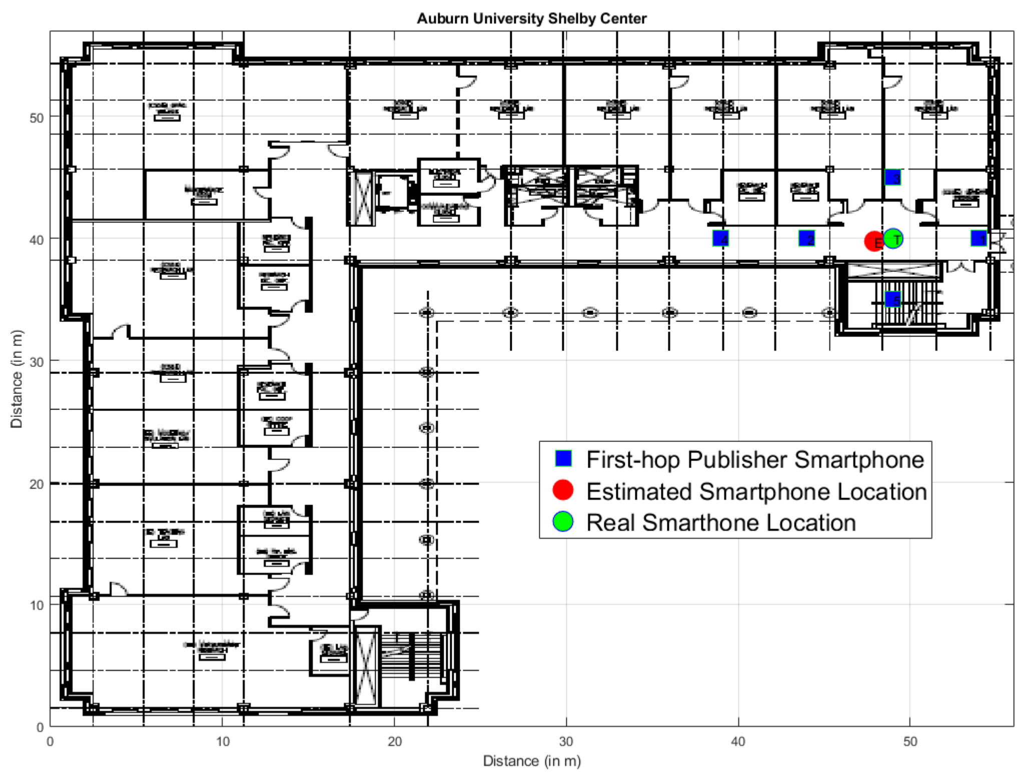

Figure 7.18: Auburn University Shelby Center location estimation using five publisher smartphones and no error in the publisher smartphone locations

To further investigate the error propagation in our indoor localization system, we conducted an additional experiment where we induced the error observed in the previous experiment. In the previous experiment, all the publisher smartphones served as first-hop neighbors and had error-free locations. In this new experiment, we examined the location estimation of the subscriber smartphone when aided by second-hop publisher neighbors. Each publisher's location was induced with an error of 1.1 meters, which was determined in the previous first-hop neighbor experiment.

The results are illustrated in Figure 7.19. It can be observed that the error in the estimated location of the subscriber smartphone has doubled, resulting in an error of 2.13 meters. This finding suggests that if the subscriber smartphone communicates with a second-hop publisher smartphone, which obtained its location from the first-hop publishers earlier in time, the error in the estimated location is doubled. This highlights the significance of error propagation when multiple hops are involved in the localization process.

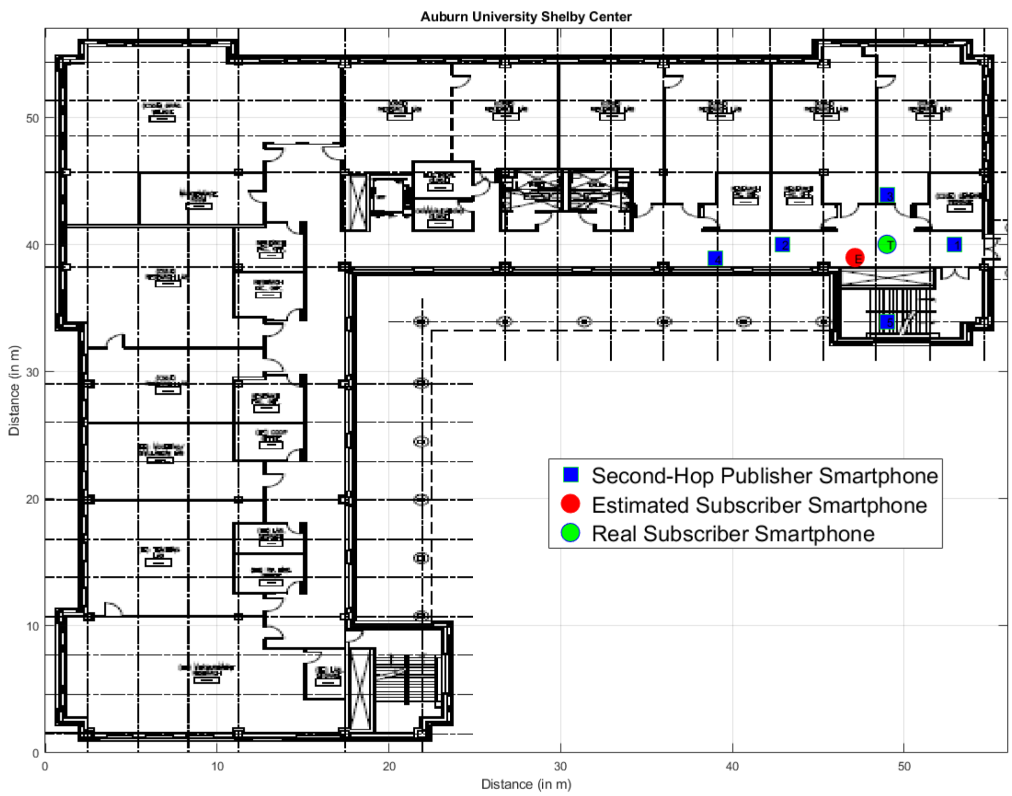

Figure 7.19: Auburn University Shelby Center location estimation using five publisher smartphones and one hop error in the publisher smartphone location as obtained from the first hop error propagation experiment

Furthermore, we extended our experiments to investigate the error propagation in the estimated smartphone location when the publisher smartphones acted as third and fourth-hop neighbors, respectively. The results are presented in Figure 7.20 and Figure 7.21, displaying the error in the estimated location of the subscriber smartphone.

From the figures, it is evident that the error propagation follows a consistent pattern of doubling with each subsequent hop count of the publisher smartphone aiding the localization process.

In the case of the third-hop neighbor, the error in the estimated location reaches 24.98 meters, while for the fourth-hop neighbor, it increases to 48.38 meters. These findings highlight the cumulative effect of error propagation as the number of hops increases, emphasizing the importance of mitigating errors at each stage of the localization process.

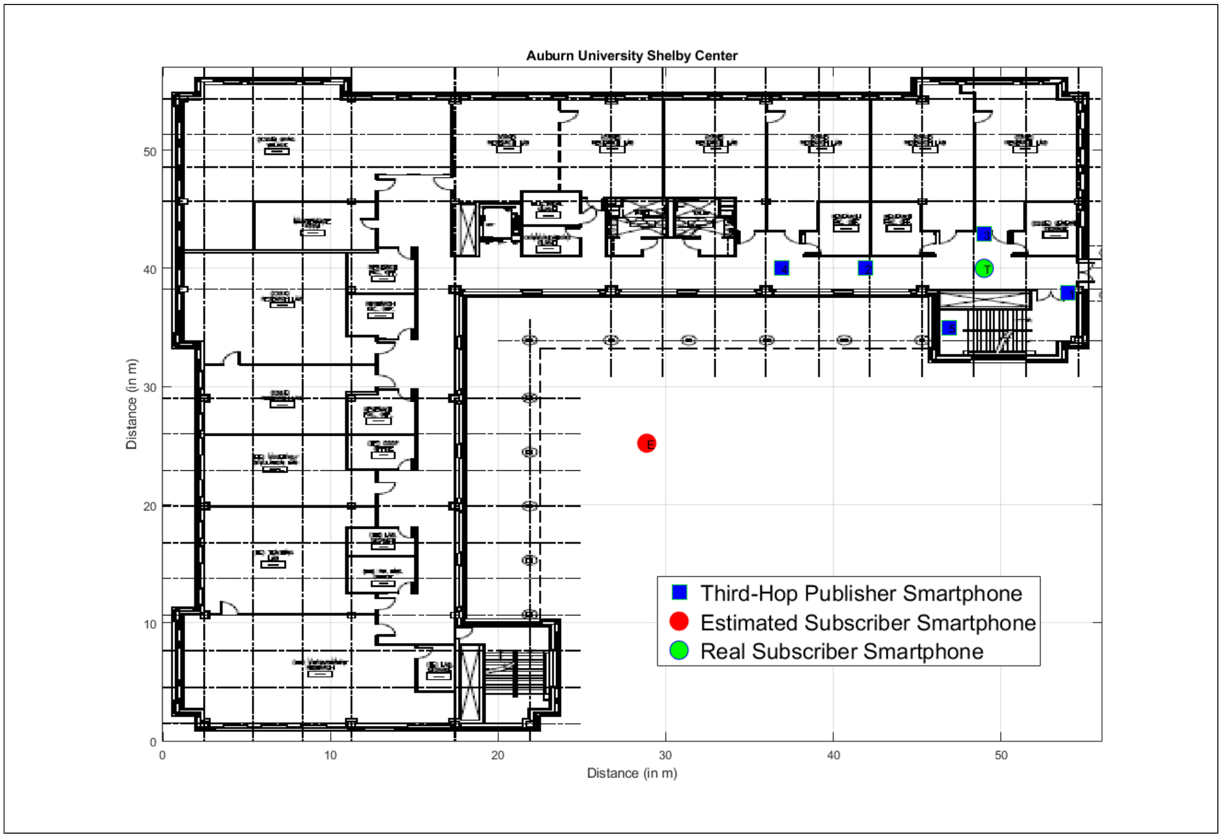

Figure 7.20: Auburn University Shelby Center location estimation using five publisher smartphones and second hop error in the publisher smartphone location as obtained from the second hop error propagation experiment

Based on the error propagation experiments conducted in our smartphone-based indoor localization system, we can draw several conclusions.

Firstly, the experiments clearly demonstrate that errors in the location estimation of publisher smartphones can propagate and affect the accuracy of the subscriber smartphone's location estimation. As the publisher smartphones act as aids in the localization process, any error introduced in their location information can lead to a corresponding increase in the error of the subscriber smartphone's location estimation. Secondly, the results show a consistent pattern of doubling in

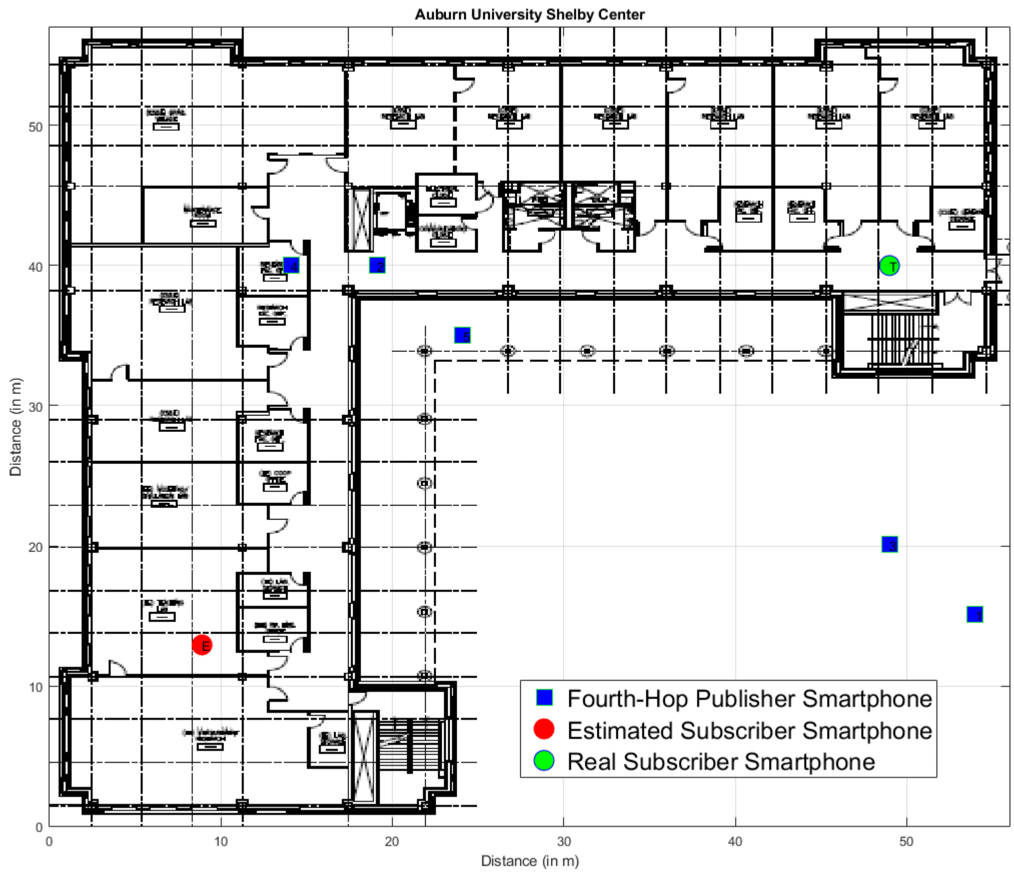

Figure 7.21: Auburn University Shelby Center location estimation using five publisher smartphones and third hop error in the publisher smartphone location as obtained from the third hop error propagation experiment

the error magnitude with each subsequent hop count of the publisher smartphone. This indicates that the error propagation follows a cumulative trend, with the error increasing exponentially as the number of hops between publisher and subscriber smartphones increases.

These findings emphasize the importance of minimizing errors at each stage of the localization process and highlight the need for accurate localization of the publisher smartphones to ensure reliable and precise location estimation for the subscriber smartphone. Mitigating errors in the initial localization of the publisher smartphones can significantly reduce the overall error propagation and improve the accuracy of the subscriber smartphone's location estimation. Overall, these error propagation experiments provide valuable insights into the potential sources of error and their impact on the accuracy of smartphone-based indoor localization systems. By understanding the characteristics of error propagation, researchers and system designers can develop strategies to mitigate errors and enhance the overall performance of indoor localization systems.

Chapter 8

Conclusions

Indoor localization and tracking have become indispensable for various applications, ranging from assisting users in navigating complex indoor environments to ensuring efficient asset tracking and robust security systems. In this research, we introduce a novel smartphone-based indoor localization technique that harnesses the power of IEEE $802.11mc$ fine time measurement (FTM), which represents the next generation of indoor navigation and tracking systems. By leveraging this advanced technology, we aim to address the challenges associated with indoor localization and provide accurate and reliable positioning information.

Our FTM-based technique is designed with scalability and adaptability in mind. A notable advantage is that it does not rely on additional hardware or infrastructure, as it utilizes existing WiFi access points. This makes it highly cost-effective and easy to deploy compared to other indoor localization methods that often require specialized equipment or infrastructure investments. By leveraging the ubiquitous presence of WiFi access points, our technique can be readily applied to a variety of indoor environments, making it a versatile solution for different applications and scenarios.

The results of our research demonstrate the superior performance of the FTM-based technique in terms of accuracy. Our approach leverages the precise estimation of the round-trip time (RTT) distances through FTM, enabling highly accurate position estimation. Extensive real-world experiments conducted in typical university campus environments reveal that our technique achieves an impressive localization error of less than 1.5 meters in 95% of cases. This level of accuracy is crucial for many indoor applications, ensuring users can rely on precise location information to navigate indoor spaces or track assets with confidence.

One key advantage of our FTM-based technique is its superiority over other WiFi-based approaches. Unlike traditional methods that compute distance estimation at the MAC layer, which can introduce software delays and inaccuracies, our technique operates at the physical layer through dedicated hardware units. This hardware-based approach eliminates software-related delays and ensures faster and more accurate distance measurements. The improved efficiency and accuracy of our distance estimation contribute to the overall performance enhancement of the indoor localization system, offering users a reliable and precise positioning experience.

In conclusion, our FTM-based technique for indoor localization and tracking exhibits significant promise and presents several advantages. Its high accuracy, coupled with its scalability and adaptability, positions it as an attractive solution for diverse indoor applications. The fact that it does not require additional hardware or infrastructure simplifies its implementation and reduces costs. Furthermore, future work will focus on further refining the technique's accuracy and performance, exploring potential enhancements to overcome specific challenges in different indoor environments. By continuing to advance this innovative approach, we can unlock its potential for improving indoor navigation, asset tracking, security systems, and other domains requiring precise indoor localization capabilities.

Chapter 9

Future Work

Although our FTM-based indoor localization and tracking technique has yielded promising outcomes, there are still opportunities for refinement and additional research. In the following section, we explore several potential directions for future work:

Improving Accuracy and Robustness

Enhancing the accuracy and robustness of our FTM-based technique is a significant area for future exploration. While our technique has demonstrated impressive accuracy in typical university campus environments, it may face challenges when deployed in different settings with varying signal propagation characteristics or interference. To overcome these limitations and ensure the adaptability of our technique, further research is necessary. This research should focus on improving the robustness of our approach, optimizing it for diverse environments, and addressing potential challenges related to signal propagation and interference. By doing so, we can enhance the performance and applicability of our FTM-based technique across a wider range of indoor environments.

Multi-Modal Localization

Future research can also focus on investigating the potential benefits of integrating FTM-based localization with other modalities, such as Bluetooth Low Energy (BLE) beacons or Inertial Measurement Units (IMUs). By combining multiple localization techniques, we can leverage the strengths of each modality to enhance the accuracy and reliability of indoor localization systems. For instance, BLE beacons can provide additional reference points and help overcome challenges posed by signal propagation in complex environments. Similarly, IMUs can capture motion and orientation data, complementing the static measurements obtained from FTM-based localization. By integrating these modalities, we can create a comprehensive and robust indoor localization solution that offers improved accuracy, even in challenging signal environments. Further exploration in this area will unlock new possibilities for enhancing the performance and versatility of indoor localization systems.

Real-Time Tracking and Navigation

While our current FTM-based technique excels in accurate position estimation, it does not offer real-time tracking or navigation capabilities. To address this limitation, future research can focus on exploring the potential of utilizing our technique for real-time tracking and navigation purposes. This would involve not only refining the position estimation accuracy but also developing efficient algorithms for path planning and guidance. By incorporating real-time tracking and navigation features, our FTM-based technique can become a valuable tool for applications such as indoor navigation, asset tracking, and interactive location-based services. This direction of research would require considering factors such as real-time data processing, continuous updating of position estimates, and seamless integration with user interfaces for intuitive guidance. The successful integration of real-time tracking and navigation capabilities would significantly enhance the practicality and usability of our FTM-based technique in various indoor environments.

Privacy and Security

Lastly, it is essential to address the important concerns of privacy and security when working with any location-based technology. In order to ensure the widespread adoption and acceptance of our FTM-based indoor localization and tracking technique, future research should prioritize the development of privacy-preserving and secure methodologies.

One potential avenue for future work is the exploration of privacy-enhancing techniques that allow users to maintain control over their location information. This could involve implementing techniques such as data anonymization, encryption, and access control mechanisms to safeguard the privacy of individuals' location data. By incorporating privacy-preserving measures into our FTM-based technique, we can alleviate concerns regarding the potential misuse or unauthorized access to sensitive location information.

Additionally, it is crucial to address security concerns to protect the integrity and reliability of the indoor localization system. Future research efforts should focus on developing robust security measures, including authentication mechanisms, data integrity checks, and secure communication protocols. By implementing these security measures, we can ensure that the indoor localization and tracking system is resilient to potential attacks or unauthorized manipulations.

Overall, by dedicating attention to privacy and security considerations, we can enhance the trustworthiness and acceptance of our FTM-based indoor localization and tracking technique. Through ongoing research and development, we can create a framework that not only achieves high accuracy and performance but also prioritizes the privacy and security of individuals' location data.

In conclusion, our FTM-based technique for indoor localization and tracking has demonstrated promising results and holds significant potential for various applications. However, there is still ample room for further research and development in order to enhance its accuracy, robustness, and security.

Continued research efforts can focus on refining the algorithms and methodologies employed in our technique to achieve even higher levels of accuracy in indoor localization. This can involve exploring advanced signal processing techniques, machine learning algorithms, and statistical modeling approaches to improve the precision and reliability of the position estimation.

Moreover, the robustness of our technique can be further enhanced by investigating strategies to mitigate the impact of environmental factors, such as signal interference, multipath propagation, and dynamic obstacles. By developing adaptive algorithms that can adapt to different environmental conditions, our technique can maintain high performance across a wide range of indoor settings.

In addition, there is a need to address the security aspects of indoor localization and tracking systems. Further research can focus on developing secure protocols, encryption mechanisms, and access control mechanisms to safeguard the privacy and integrity of the collected location data. By integrating robust security measures, we can ensure the trustworthiness and confidentiality of the indoor localization system.

Overall, through ongoing research and development, we can continue to advance the field of indoor localization and tracking, resulting in more accurate, robust, and secure solutions. With the potential to revolutionize indoor navigation, asset tracking, and security, our FTM-based technique lays the foundation for a new era of indoor positioning technology.

Bibliography

[1] Mark Sullivan. *A brief history of GPS*. 2023. URL: http://www.pcworld.com/article/2000276/a-brief-history-of-gps.html (visited on 11/23/2022).

[2] Charles Little and Ralph Peters. "Simulated Mobile Self-Location Using 3D Range Sensing and an A-Priori Map". In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. Barcelona, Spain, 2005.

[3] Darius Burschka and Gregory D. Hager. "V-GPS(SLAM): Vision-Based Inertial System for Mobile Robots". In: *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*. New Orleans, LA, 2004.

[4] Daniel C. Asmar, John S. Zelek, and Samer M. Abdallah. "SmartSLAM: localization and mapping across multienvironment". In: *Proc. of the 2004 IEEE International Conference on Systems, Man and Cybernetics*. 2004.

[5] Fernáin Izquierdo et al. "Performance evaluation of a TOA-based trilateration method to locate terminals in WLAN". In: (2006).

[6] Tianyu Yang, Adil Cabani, and Hicham Chafouk. "A Survey of Recent Indoor Localization Scenarios and Methodologies". In: *Sensors* 21.23 (2021), p. 8086. DOI: 10.3390/s21238086. URL: https://doi.org/10.3390/s21238086.

[7] Laura Feeney. *Wireless Ad Hoc Network*. Accessed 23 November 2022. 2022. URL: https://en.wikipedia.org/wiki/Wireless_ad_hoc_network#Protocol_stack.

[8] G. Zanca et al. "Experimental comparison of RSSI-based localization algorithms for indoor wireless sensor networks". In: Glasgow, Scotland, 2008, pp. 1–5.

[9] R. Want et al. “The active badge location system”. In: *Proceedings of ACM Transactions on Information Systems*. 1992, pp. 91–102.

[10] Jonathan Dixon and Oliver Henlich. *Mobile Robot Navigation*. 1997. URL: `http://www.doc.ic.ac.uk/~nd/surprise_97/journal/vol4/jmd/`.

[11] Pengfei Du et al. “Experimental demonstration of 3D visible light positioning using received signal strength with low-complexity trilateration assisted by deep learning technique”. In: *IEEE Access* 7 (2019), pp. 93986–93997.

[12] Sebastian Sadowski, Petros Spachos, and Konstantinos N Plataniotis. “Memoryless techniques and wireless technologies for indoor localization with the internet of things”. In: *IEEE Internet of Things Journal* 7.11 (2020), pp. 10996–11005.

[13] Abdul Syafiq Abdull Sukor et al. “RSSI-based for device-free localization using deep learning technique”. In: *Smart Cities* 3.2 (2020), pp. 444–455.

[14] Abebe Belay Adege et al. “An indoor and outdoor positioning using a hybrid of support vector machine and deep neural network algorithms”. In: *Journal of sensors* 2018 (2018), pp. 1–12.

[15] Eduardo Carvalho et al. “Exploiting the use of machine learning in two different sensor network architectures for indoor localization”. In: *2016 IEEE International Conference on Industrial Technology (ICIT)*. IEEE. 2016, pp. 652–657.

[16] Yong Shi et al. “An RSSI classification and tracing algorithm to improve trilateration-based positioning”. In: *Sensors* 20.15 (2020), p. 4244.

[17] Mohamad M. Saad et al. “High-Accuracy Reference-Free Ultrasonic Location Estimation”. In: *IEEE Transactions on Instrumentation and Measurement* 61.6 (2012), pp. 1561–1570.

[18] Chris J. Bleakley et al. “Robust High-Accuracy Ultrasonic Range Measurement System”. In: *IEEE Transactions on Instrumentation and Measurement* 60.10 (2011), pp. 3334–3341.

[19] C. D. McGillem and T. S. Rappaport. “Infra-red location system for navigation of autonomous vehicles”. In: *Proceedings. 1988 IEEE International Conference on Robotics and Automation*. IEEE. 2002.

[20] M. M. Saad et al. “High-Accuracy location estimation for a Mobile Tag using one-way UWB signaling”. In: *Ubiquitous Positioning, Indoor Navigation and Location Based Service*. 2012, pp. 1–8.

[21] Andre Gunther and Christian Hoene. “Measuring Round Trip Times to Determine the Distance Between WLAN Nodes”. In: *Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*. 2005, pp. 768–779.

[22] Jonas Hallberg et al. “Positioning with Bluetooth”. In: *10th International Conference on Telecommunications, ICT*. Vol. 2. 2003, pp. 954–958.

[23] Ahmed Makki et al. “Survey of WiFi Positioning Using Time-Based Techniques”. In: *The International Journal of Computer and Telecommunications Networking* (2015), pp. 218–233.

[24] Xuyu Wang, Lingjun Gao, and Shiwen Mao. “PhaseFi: Phase Fingerprinting for Indoor Localization with a Deep Learning Approach”. In: *2015 IEEE Global Communications Conference (GLOBECOM)*. 2015, pp. 1–6. DOI: 10.1109/GLOCOM.2015.7417517.

[25] P. Bahl and V.N. Padmanabhan. “RADAR: an in-building RF-based user location and tracking system”. In: *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*. Vol. 2. 2000, 775–784 vol.2. DOI: 10.1109/INFCOM.2000.832252.

[26] Moustafa Youssef and Ashok Agrawala. “The Horus WLAN Location Determination System”. In: *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*. MobiSys ’05. Association for Computing Machinery, 2005, 205–218. ISBN:

1931971315. DOI: 10.1145/1067170.1067193. URL: https://doi.org/10.1145/1067170.1067193.

[27] Zheng Yang, Zimu Zhou, and Yunhao Liu. "From RSSI to CSI: Indoor localization via channel response". In: *ACM Computing Surveys (CSUR)* 46.2 (2013), pp. 1–32.

[28] Ali Yassin et al. "Recent advances in indoor localization: A survey on theoretical approaches and applications". In: *IEEE Communications Surveys & Tutorials* 19.2 (2016), pp. 1327–1346.

[29] Zahid Farid, Rosdiadee Nordin, Mahamod Ismail, et al. "Recent advances in wireless indoor localization techniques and system". In: *Journal of Computer Networks and Communications* 2013 (2013).

[30] Wonho Kang and Youngnam Han. "SmartPDR: Smartphone-Based Pedestrian Dead Reckoning for Indoor Localization". In: *IEEE Sensors Journal* 15.5 (2015), pp. 2906–2916.

[31] Junjie Liu and R Jain. "Survey of wireless based indoor localization technologies". In: *Department of Science & Engineering, Washington University* (2014).

[32] Anahid Basiri et al. "Indoor location based services challenges, requirements and usability of current solutions". In: *Computer Science Review* 24 (2017), pp. 1–12.

[33] Yanying Gu, Anthony Lo, and Ignas Niemegeers. "A survey of indoor positioning systems for wireless personal networks". In: *IEEE Communications surveys & tutorials* 11.1 (2009), pp. 13–32.

[34] Google. *Android Operating System*. Accessed 23 November 2022. 2022. URL: https://source.android.com/.

[35] Google. *Wi-Fi Aware overview*. Accessed 23 November 2022. 2022. URL: https://developer.android.com/guide/topics/connectivity/wifi-aware.html.

[36] Google. *Wi-Fi location: ranging with RTT*. Accessed 23 November 2022. 2022. URL: https://developer.android.com/guide/topics/connectivity/wifi-rtt.

[37] Edward Au. “The latest Progress on IEEE 802.11mc and IEEE 802.11ai[Standards]”. In: 11 (2016), pp. 19 –21.

[38] Claudia Barberis et al. “Experiencing Indoor Navigation on Mobile Devices”. In: *IT Professional* 16.1 (2013), pp. 50–57.

Appendices

Appendix A

Indoor Localization Fine Time Measurement Android Application Source Code

```
/* Class to maintain the Android App to conduct the Android Aware Scan and
   Run the FTM technique.
   Author: Abhishek Kulkarni (aak0010)
*/
public class MainActivity extends AppCompatActivity {

    private class ScanWifiNetworkReceiver extends BroadcastReceiver {

        @Override
        public void onReceive(final Context context, final Intent intent) {
            final List<ScanResult> wifiNetworks = wifiManager.getScanResults();
            Timber.d("received scan result. %s, size: %d", intent.toString(), wifiNetworks.size());
            lblSearchHint.setVisibility(View.GONE);
            wifiNetworkAdapter.setWifiNetworks(wifiManager.getScanResults());
        }
    }

    private static final int REQUEST_ENABLE_LOCATION = 8956;
```

```java

    public static boolean isLocationEnabled(@NonNull final Context context) {
        LocationManager lm = (LocationManager) context.getSystemService(Context.LOCATION_SERVICE);
        return lm.isLocationEnabled();
    }

    @BindView(R.id.coordinator)
    CoordinatorLayout coordinatorLayout;
    @BindView(R.id.fab)
    FloatingActionButton fab;
    @BindView(R.id.lblSearchHint)
    TextView lblSearchHint;
    @BindView(R.id.listWifiNetworks)
    RecyclerView listWifiNetworks;
    @BindView(R.id.toolbar)
    Toolbar toolbar;
    @BindView(R.id.txtCapabilities)
    TextView txtCapabilities;
    @BindView(R.id.txtDeviceToAPSupported)
    TextView txtDeviceToApSupported;
    @BindView(R.id.aware_button)
    Button button;
    @BindView(R.id.rttResultView)
    TextView rttResultView;
    private LocationPermissionController permissionController;
    private WifiRttManager rttManager;
    private WifiManager wifiManager;
```

```java
    private WifiNetworkAdapter wifiNetworkAdapter;
    private ScanWifiNetworkReceiver wifiNetworkReceiver;
    private Context mContext;
    private Handler mHandler;

    private static final String TAG = "MainActivity";
    public static final String AWARE_SERVICE_NAME = "LIMLAB-AWARE-SERVICE";

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        menu.add(getString(R.string.version_info_app, BuildConfig.VERSION_NAME, BuildConfig.VERSION_CODE));
        return true;
    }

    @Override
    public void onRequestPermissionsResult(final int requestCode, @NonNull final String[] permissions,
                                           @NonNull final int[] grantResults) {
        if (permissionController.onRequestPermissionsResult(requestCode, permissions, grantResults)) {
            startWifiScan();
        } else {
            super.onRequestPermissionsResult(requestCode, permissions, grantResults);
        }
    }
```

```java

    @Override
    protected void onActivityResult(final int requestCode, final int
resultCode, final Intent data) {
        if (requestCode == REQUEST_ENABLE_LOCATION) {
            if (resultCode == RESULT_OK) {
                startWifiScan();
            } else {
                Snackbar.make(coordinatorLayout, R.string.
location_service_disabled, Snackbar.LENGTH_SHORT).setAction
                        (android.R.string.ok, view ->
startEnableLocationServicesActivity()).show();
            }
            return;
        }
        super.onActivityResult(requestCode, resultCode, data);
    }

    @SuppressLint("WrongConstant")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ButterKnife.bind(this);
        setSupportActionBar(toolbar);
        permissionController = new LocationPermissionController();
        fab.setOnClickListener(view -> startWifiScan());
        wifiManager = (WifiManager) getApplicationContext().
getSystemService(Context.WIFI_SERVICE);
```

```java
        wifiNetworkReceiver = new ScanWifiNetworkReceiver();
        rttManager = (WifiRttManager) getSystemService(Context.WIFI_RTT_RANGING_SERVICE);
        initUI();
        mContext = this; //context initialization to setup the android aware Broadcast receiver

        button.setOnClickListener(v -> invokeAndroidAware());
    }

    private void showShortMessage(String message) {
        Toast.makeText(mContext, message, Toast.LENGTH_SHORT).show();
    }
    private void invokeAndroidAware() {
        /*Android Aware Core Method */
        WifiAwareManager wifiAwareManager =
                (WifiAwareManager)this.getSystemService(Context.WIFI_AWARE_SERVICE);
        IntentFilter filter =
                new IntentFilter(WifiAwareManager.ACTION_WIFI_AWARE_STATE_CHANGED);
        BroadcastReceiver myReceiver = new BroadcastReceiver() {
            @Override
            public void onReceive(Context context, Intent intent) {
                if (wifiAwareManager.isAvailable()) {
                    showShortMessage("Wifi is On");
                }
            }
        };
```

```
        if (wifiAwareManager.isAvailable()) {
            Toast.makeText(this, "WiFi Aware is available", Toast.LENGTH_SHORT).show();
            wifiAwareManager.attach(new PublisherAttachCallback(this), mHandler); // initialization of Publisher
            wifiAwareManager.attach(new SubscriberAttachCallback(this), mHandler); // initialization of Subscriber
        }
        mContext.registerReceiver(myReceiver, filter);
    }

    @Override
    protected void onStop() {
        super.onStop();
        try {
            unregisterReceiver(wifiNetworkReceiver);
        } catch (IllegalArgumentException e) {
        }
    }

    private void handleLocationServiceDisabled() {
        Snackbar.make(coordinatorLayout, R.string.location_service_disabled, Snackbar.LENGTH_INDEFINITE)
                .setAction(android.R.string.ok, view -> startEnableLocationServicesActivity())
                .show();
    }

```

```
    private void initUI() {
        txtDeviceToApSupported.setText(String.valueOf(wifiManager.isDeviceToApRttSupported()));
        txtCapabilities.setText(getString(R.string.rtt_available, rttManager.isAvailable()));
        rttResultView.setText(getString(R.string.rtt_results));
        listWifiNetworks.setLayoutManager(new LinearLayoutManager(this));
        listWifiNetworks.setItemAnimator(new DefaultItemAnimator());
        listWifiNetworks.setHasFixedSize(true);
        listWifiNetworks.setVisibility(View.GONE);
        wifiNetworkAdapter = new WifiNetworkAdapter(getApplicationContext());
        listWifiNetworks.setAdapter(wifiNetworkAdapter);
        wifiNetworkAdapter.setClickListener(wifiNetwork -> {
            startActivity(SelectedActivity.builtIntent(wifiNetwork, getApplicationContext()));
        });
        lblSearchHint.setVisibility(View.VISIBLE);
    }

    private void startEnableLocationServicesActivity() {
        Intent enableLocationIntent = new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
        startActivityForResult(enableLocationIntent, REQUEST_ENABLE_LOCATION);
    }

    private void startWifiScan() {
```

```
        if (!permissionController.checkLocationPermissions(getApplicationContext())) {
            permissionController.requestLocationPermission(this, coordinatorLayout);
            return;
        }
        if (!isLocationEnabled(getApplicationContext())) {
            handleLocationServiceDisabled();
            return;
        }
        if (!wifiManager.isWifiEnabled()) {
            Snackbar.make(coordinatorLayout, R.string.enable_wifi, Snackbar.LENGTH_LONG).show();
            return;
        }
        listWifiNetworks.setVisibility(View.VISIBLE);
        IntentFilter filter = new IntentFilter(WifiManager.SCAN_RESULTS_AVAILABLE_ACTION);
        registerReceiver(wifiNetworkReceiver, filter);
        final boolean successful = wifiManager.startScan();
        Timber.d("Started scan successful: %b", successful);
    }
}
```

Source Code A.1: Central Control Module for Indoor Localization App Activities

```
/* Class to maintain Publisher activities.
   Author: Abhishek Kulkarni (aak0010)
*/
public class PublisherAttachCallback extends AttachCallback {
```

```java

    private static final String TAG = "PublisherAttachCallback";
    private static final boolean DBG = true;
    private static final String PublisherID = "Publisher #1";

    private PublishDiscoverySession mPublisherSession;
    private final Context mContext;

    public PublisherAttachCallback(Context context) {
        this.mContext = context;
    }

    private void showShortMessage(String message) {
        Toast.makeText(mContext, message, Toast.LENGTH_SHORT).show();
    }

    @Override
    public void onAttachFailed() {
        showShortMessage(TAG + "onAttachFailed");
    }

    @Override
    public void onAttached(WifiAwareSession session) {

        showShortMessage("onAttach");
        PublishConfig config = new PublishConfig.Builder()
                .setServiceName(MainActivity.AWARE_SERVICE_NAME)
                .setRangingEnabled(true)
                .build();
```

```

    session.publish(config, new DiscoverySessionCallback() {

        @Override
        public void onPublishStarted(@NonNull PublishDiscoverySession session) {
            showShortMessage(TAG + " onPublishStarted");
            DataCollectionFile.dataFile(TAG + " PublisherStarted " + " Time: " + System.nanoTime(), mContext);
        }

        @Override
        public void onServiceDiscovered(PeerHandle peerHandle,
                                        byte[] serviceSpecificInfo, List<byte[]> matchFilter){
            int messageId = 1234;
            showShortMessage("Location: " + locationInfo + " PeerHandle: " + peerHandle);
            DataCollectionFile.dataFile(TAG + " The Publisher Location: " + locationInfo, mContext);
            mPublisherSession.sendMessage(peerHandle, messageId, locationInfo.getBytes());
        }

        @Override
        public void onMessageReceived(PeerHandle peerHandle, byte[] message) {
            String str = new String(message, StandardCharsets.UTF_8);
```

```
                String[] msgSubscriberSplit = str.split("-");
                String msgTransmissionTime = String.valueOf((System.
nanoTime() - Long.parseLong(msgSubscriberSplit[1]))/1_000_000_000
.0);
                showShortMessage(TAG + " onMessageReceived : " + str)
;
                DataCollectionFile.dataFile(TAG + " PublisherID: " +
 PublisherID + " SubscriberID: " + msgSubscriberSplit[0] + "
TransmissionTime: " + msgTransmissionTime, mContext);
            }
        }, null);
    }
}
```

Source Code A.2: Initiation and Management of Publisher Role in Indoor Localization App

```
/* Class to maintain Subscriber tasks
   Author: Abhishek Kulkarni (aak0010)
*/
public class SubscriberAttachCallback extends AttachCallback {

    private static final String TAG = "SubscriberAttachCallback";
    private static final boolean DBG = true;
    private static final String SubscriberID = "Subscriber #1";
    private final Context mContext;
    private SubscribeDiscoverySession mSubscribeDiscoverySession;
    private final RttRangingManager rangingManager;
    private PeerHandle publisherPeerHandle;
    private static Disposable rangingDisposable;

```

```

    public SubscriberAttachCallback(Context context) {
        this.mContext = context;
        this.rangingManager = new RttRangingManager(context);
    }

    private void showShortMessage(String message) {
        Toast.makeText(mContext, message, Toast.LENGTH_SHORT).show();
    }
    private void beginRttRanging() {
        rangingDisposable = rangingManager.startRanging(
publisherPeerHandle)
                .repeat(10)
                .subscribeOn(Schedulers.io())
                .observeOn(AndroidSchedulers.mainThread())
                .subscribe();
    }

    public static void stopRttRanging() {
        if (rangingDisposable == null) {
            return;
        }
        rangingDisposable.dispose();
    }

    @Override
    public void onAttachFailed() {
        showShortMessage("onAttachFailed");
    }
```

```
    @Override
    public void onAttached(WifiAwareSession session) {

        SubscribeConfig config = new SubscribeConfig.Builder()
                .setServiceName(MainActivity.AWARE_SERVICE_NAME)
                .setMinDistanceMm(10)
                .build();

        session.subscribe(config, new DiscoverySessionCallback() {

            @Override
            public void onSubscribeStarted(@NonNull SubscribeDiscoverySession session) {
                mSubscribeDiscoverySession = session;
                DataCollectionFile.dataFile(TAG + " The Subscriber Session Started at: " + System.nanoTime(), mContext);
            }

            @Override
            public void onServiceDiscovered(PeerHandle peerHandle,
                                            byte[] serviceSpecificInfo, List<byte[]> matchFilter) {
                DataCollectionFile.dataFile("PeerHandle: " + peerHandle, mContext);
                publisherPeerHandle = peerHandle;
                beginRttRanging();
            }

```

```java
            @Override
            public void onMessageSendSucceeded(int messageId) {
                showShortMessage("The message has been successfully sent: " + messageId);
            }

            @Override
            public void onMessageSendFailed(int messageId) {
                showShortMessage("The message sending Failed: " + messageId);
            }
        }, null);
    }
}
```

Source Code A.3: Initiation and Management of Subscriber Role in Indoor Localization App

```java
/* Class to control all the Location permission needed to perform FTM.
   Author: Abhishek Kulkarni (aak0010)
*/
public class LocationPermissionController {

    private static final int REQUEST_LOCATION_PERMISSION = 8545;

    public boolean checkLocationPermissions(final Context context) {
        return ContextCompat.checkSelfPermission(context, Manifest.permission.ACCESS_COARSE_LOCATION) ==
                PackageManager.PERMISSION_GRANTED && ContextCompat.checkSelfPermission(context, Manifest.permission
```

```
                .ACCESS_FINE_LOCATION) == PackageManager.PERMISSION_GRANTED;
 }

 public boolean onRequestPermissionsResult(final int requestCode, @NonNull final String[] permissions,
         @NonNull final int[] grantResults) {
     if (requestCode == REQUEST_LOCATION_PERMISSION) {
         return verifyPermissions(grantResults);
     }
     return false;
 }

 public void requestLocationPermission(final Activity activity, final View snackbarContainer) {
     if (ActivityCompat.shouldShowRequestPermissionRationale(activity, Manifest.permission.ACCESS_COARSE_LOCATION) ||
             ActivityCompat.shouldShowRequestPermissionRationale(activity, Manifest.permission
                     .ACCESS_FINE_LOCATION)) {
         Snackbar.make(snackbarContainer, R.string.permission_location_description, Snackbar.LENGTH_INDEFINITE)
                 .setAction(android.R.string.ok, view -> requestPermissions(activity)).show();
     } else {
         requestPermissions(activity);
     }
 }

```

```
    private void requestPermissions(final Activity activity) {
        ActivityCompat.requestPermissions(activity,
                new String[]{Manifest.permission.
ACCESS_COARSE_LOCATION, Manifest.permission.ACCESS_FINE_LOCATION},
                REQUEST_LOCATION_PERMISSION);
    }

    private boolean verifyPermissions(int[] grantResults) {
        if (grantResults.length < 1) {
            return false;
        }

        for (int result : grantResults) {
            if (result != PackageManager.PERMISSION_GRANTED) {
                return false;
            }
        }
        return true;
    }
}
```

Source Code A.4: Location Permission Controller: Managing RTT Process Permissions in the Indoor Localization App

```
package com.tigermail.aak0010.rttmanager;
public class RttManagerApplication extends Application {
    @Override
    public void onCreate() {
        super.onCreate();
        Timber.plant(new Timber.DebugTree());
```

```
    }
}
```

Source Code A.5: RTT Manager: Controlling RTT Data Collection and Processing in the Indoor Localization App

```
/* Class to maintain and perfrom the RTT ranging using the list of
   PeerHandles
   or AP.
   Author: Abhishek Kulkarni (aak0010)
*/
public class RttRangingManager {

    private final Executor mainExecutor;
    private final WifiRttManager rttManager;
    private final Context mContext;
    private static final String TAG = "RttRangingManager";
    private int counter = 0;

    @SuppressLint("WrongConstant")
    public RttRangingManager(final Context context) {
        rttManager = (WifiRttManager) context.getSystemService(
   Context.WIFI_RTT_RANGING_SERVICE);
        mainExecutor = context.getMainExecutor();
        this.mContext = context;
    }

    private void showShortMessage(String message) {
        Toast.makeText(mContext, message, Toast.LENGTH_SHORT).show();
    }
```

```java
    @SuppressLint("MissingPermission")
    public Single<List<RangingResult>> startRanging(
            @NonNull final PeerHandle peerHandle) {
        return Single.create(emitter -> {
            final RangingRequest request = new RangingRequest.Builder()
                    .addWifiAwarePeer(peerHandle)
                    .build();
            final RangingResultCallback callback = new RangingResultCallback() {
                @Override
                public void onRangingFailure(final int i) {
                    emitter.onError(new RuntimeException("The WiFi-Ranging failed with error code: " + i));
                }

                @Override
                public void onRangingResults(final List<RangingResult> result) {
                    if (result.isEmpty()) {
                        counter++;
                        showShortMessage(" Failed PeerHandle: " + peerHandle);
                        DataCollectionFile.dataFile(TAG + " The ranging result is empty ", mContext);
                    }
                    DataCollectionFile.logRangingResult(result, mContext);
                    if(counter > 10) {
```

```
                        showShortMessage("Counter: " + counter);
                        SubscriberAttachCallback.stopRttRanging();
                    }
                    emitter.onSuccess(result);
                }
            };
            rttManager.startRanging(request, mainExecutor, callback);
        });
    }

}
```

Source Code A.6: RTT Ranging Manager: Initiating and Collecting Distance in the Indoor Localization App

```
/* Class to maintain and initialise the RTT ranging task.
   Author: Abhishek Kulkarni (aak0010)
*/
public class SelectedActivity extends AppCompatActivity {

    private static final String EXTRA_WIFI_NETWORK = "WIFI_NETWORK";

    public static Intent builtIntent(final ScanResult wifiNetwork,
    Context context) {
        Intent intent = new Intent(context, SelectedActivity.class);
        intent.putExtra(EXTRA_WIFI_NETWORK, wifiNetwork);
        return intent;
    }

    @BindView(R.id.rttResultView)
```

```
    TextView rttResultView;
    @BindView(R.id.startButton)
    Button startButton;
    @BindView(R.id.stopButton)
    Button stopButton;
    private Disposable rangingDisposable;
    private RttRangingManager rangingManager;
    private PeerHandle wifiNetwork;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_selected);
        ButterKnife.bind(this);
        rangingManager = new RttRangingManager(getApplicationContext
());
        readIntentExtras();
        initUI();
    }

    @Override
    protected void onStop() {
        super.onStop();
        stopRanging();
    }

    private String buildLogString(final RangingResult result) {
```

```
        String resultString = getString(R.string.log, result.
    getRangingTimestampMillis(), result.getRssi(), result
                .getDistanceMm(), rttResultView.getText()
                .toString());
        if (resultString.length() > 5000) {
            return resultString.substring(0, 5000);
        }
        return resultString;
    }


    private void initStartButtonListener() {
        startButton.setOnClickListener(view -> onStartButtonClicked()
    );

    }

    private void initStopButtonListener() {
        stopButton.setOnClickListener(view -> stopRanging());

    }

    private void initUI() {
        setTitle(getString(R.string.selected_activity_title,
    wifiNetwork));
        initStartButtonListener();
        initStopButtonListener();
    }

```

```java
    private void onStartButtonClicked() {
        rttResultView.setText("");

        rangingDisposable = rangingManager.startRanging(wifiNetwork)
                .repeat()
                .subscribeOn(Schedulers.io())
                .observeOn(AndroidSchedulers.mainThread())
                .subscribe(this::writeOutput,
                        throwable -> {
                            Timber.e(throwable, "An unexpected error occurred while start ranging.");
                            Snackbar.make(rttResultView, throwable.getMessage(), Snackbar.LENGTH_LONG).show();
                        });
    }

    private void readIntentExtras() {
        Bundle extras = getIntent().getExtras();
        wifiNetwork = (PeerHandle) extras.get(EXTRA_WIFI_NETWORK);
    }

    private void stopRanging() {
        if (rangingDisposable == null) {
            return;
        }
        rangingDisposable.dispose();
    }

```

```
    private void writeOutput(@NonNull final List<RangingResult>
   result) {
        if (result.isEmpty()) {
            Timber.d("EMPTY ranging result received.");
            return;
        }
        for (RangingResult res : result) {
            rttResultView.setText(buildLogString(res));
            Timber.d("Result: %d RSSI: %d Distance: %d mm", res.
   getRangingTimestampMillis(), res.getRssi(), res
                    .getDistanceMm());
        }
    }

}
```

Source Code A.7: Selected Activity Controller: Managing Activities in AP Mode for the Indoor Localization App

```
/* Class to maintain a file log to collect data for any application
   running
   in the App: Android Aware, RTT experiment data etc.
   Author: Abhishek Kulkarni (aak0010)
*/
public class DataCollectionFile {

    /* To save the File in the Android Storage */

    public static void dataFile(String data, Context context) {
```

```
    //    ****************************************** Directory Creation *************************
    // Data file is controlled and put into the correct directory in the Internal Data Folder of the App.
    String currentDate = new SimpleDateFormat("MM-dd-yyyy", Locale.getDefault()).format(new Date());
    File file = new File(context.getFilesDir(), "AndroidAware" + "-" + currentDate);
    final File fileName = new File(file, " AndroidAwareDataCollection.txt");
    if(!file.exists()){
        file.mkdir();
    }
    try {
        FileOutputStream fileOutputStream = new FileOutputStream(fileName, true);
        OutputStreamWriter outputStreamWriter = new OutputStreamWriter(fileOutputStream);
        outputStreamWriter.write(data + "\n");
        outputStreamWriter.close();
        fileOutputStream.flush();
        fileOutputStream.close();
    }
    catch (IOException e) {
        Timber.e("File write failed: " + e.toString());
    }
}
```

```java
    public static void logRangingResult(@NonNull List<RangingResult> result, Context context) {
        for (RangingResult res : result) {
            if(res.getStatus() == RangingResult.STATUS_SUCCESS) {
                Toast.makeText(context, "RTT_Result Log Success", Toast.LENGTH_SHORT).show();
                dataFile("PeerHandle: " + res.getPeerHandle() +
                                " RTT Attempt: " + res.getNumAttemptedMeasurements() +
                                " RTT Succ Attempt: " + res.getNumSuccessfulMeasurements() +
                                " Distance: " + res.getDistanceMm() + "mm " +
                                " Dist StdDev: " + res.getDistanceStdDevMm() + "mm ",
                        context);
            } else {
                Toast.makeText(context, "RTT_Result failed", Toast.LENGTH_SHORT).show();
            }
        }
    }
}
```

Source Code A.8: Indoor Localization App Data collection file to log all the distance data for processing

```matlab
% replace with an image of your choice
img = imread('ShelbyHall.png');

```

```
% set the range of the axes
% The image will be stretched to this.
min_x = 0:(56.38);
%min_x = 0:10;
%max_x = 100;
min_y = 0:(57.10);
%min_y = 0:10;
%max_y = 100;

%Reading the file for Node co-ordinates
Data = dlmread('Display_File.txt');
% Extract data to plot.
Array_Size = length(Data);
Ref_x = Data(1:((Array_Size-2)/2));
Ref_y = Data((((Array_Size-2)/2) +1): (Array_Size-2));
Estimated_Tx = Data(Array_Size-1);
Estimated_Ty = Data(Array_Size);
Tx = 49;
Ty = 40;
%plot(x,y,'b-*','linewidth',1.5);

% Flip the image upside down before showing it
imagesc(min_x,min_y,flip(img,1));

% NOTE: if your image is RGB, you should use flipdim(img, 1) instead
    of flipud.
hold on;
plot(Ref_x,Ref_y,'sg','MarkerSize',15, 'MarkerFaceColor','b');
a = [1:(Array_Size/2)-1]'; b = num2str(a); c = cellstr(b);
```

```
text(Ref_x, Ref_y, c);
plot(Estimated_Tx,Estimated_Ty,'or','MarkerSize',15, 'MarkerFaceColor
   ','r');
text(Estimated_Tx,Estimated_Ty, 'E');
plot(Tx,Ty,'ob','MarkerSize',15, 'MarkerFaceColor','g');
text(Tx,Ty, 'T');
% set the y-axis back to normal.
set(gca,'ydir','normal');
grid on;
hold off;
```

Source Code A.9: Location Image Generation: MATLAB File for Auburn University Shelby Center in Indoor Localization App

```
/* Sophisticated Grid Method to compute the location of subscriber
   Smartphone
   using the location information from the publisher smartphones and
   the RTT
   distance obtained from the FTM technique.
   Author: Abhishek Kulkarni (aak0010)
*/
//Writing the contents of the file
func Writing(X []float64, Y []float64, Tx float64, Ty float64) {
   f, err := os.Create("Display_File.txt")
  if err != nil {
    fmt.Println(err)
  }
  defer f.Close()
  if err == nil {

```

```
    for _, sx := range X{
      Data_StrX := strconv.FormatFloat(sx, 'g', -1, 64)
      f.WriteString(Data_StrX)
      f.WriteString("\n")
    }
    for _, sy := range Y{
      Data_StrY := strconv.FormatFloat(sy, 'g', -1, 64)
      f.WriteString(Data_StrY)
      f.WriteString("\n")
    }
  f.WriteString(strconv.FormatFloat(Tx, 'g', -1, 64))
  f.WriteString("\n")
  f.WriteString(strconv.FormatFloat(Ty, 'g', -1, 64))
  f.WriteString("\n")
  f.Sync()
  }
}

//GridMethod
func Grid(X []float64, Y []float64, R []float64) {
  //All the arrays are to store the intersection points of the reference donuts
  var (
    IntsecX1st, IntsecX2nd, IntsecX3rd, IntsecX4th, IntsecX5th []float64
    IntsecY1st, IntsecY2nd, IntsecY3rd, IntsecY4th, IntsecY5th []float64
    MidptX, MidptY                                             []float64
```

```
  R_in, R_out                                              []float64
    )
const ( // These intial value are the startying point of the grid point calculation
  INITIAL_X     = -100 // These need to be assigned using the reference node co-ordinates later
  INITIAL_Y     = -100
  DIVISION_SIZE = 200
      )
//Calculation of Grid points and storing it in MidptX and MidptY respectively
Xtemp := float64(INITIAL_X)
Ytemp := float64(INITIAL_Y)

for i := 1; i <= (DIVISION_SIZE - 1); i++ {
  MidptX = append(MidptX,((Xtemp + (Xtemp + float64(1))) / float64(2)))
  MidptY = append(MidptY, ((Ytemp + (Ytemp + float64(1))) / float64(2)))
  Xtemp++
  Ytemp++
}
// Initialisation of Seperate Intersection array counters for seperate intersecting donuts
var (
  intcnt1st = 0
  intcnt2nd = 0
  intcnt3rd = 0
```

```
  intcnt4th = 0
  intcnt5th = 0
    )
//Sum total of all intersection points to calculate the Center-of-Gravity
var (
  SumIntsecX1st float64 = 0
  SumIntsecX2nd float64 = 0
  SumIntsecX3rd float64 = 0
  SumIntsecX4th float64 = 0
  SumIntsecX5th float64 = 0
  SumIntsecY1st float64 = 0
  SumIntsecY2nd float64 = 0
  SumIntsecY3rd float64 = 0
  SumIntsecY4th float64 = 0
  SumIntsecY5th float64 = 0
    )

//Calculating the Donut Radius from the RTT distance
var N = len(R) // Number of reference nodes
for i := 0; i <= (N-1); i++ {
  R_in = append(R_in, (R[i] - (0.20 * R[i])))
  R_out = append(R_out, (R[i] + (0.20 * R[i])))
}

//Calculating the grid points in the intersection area of donuts
for x := 0; x <= (DIVISION_SIZE - 2); x++ {
  for y := 0; y <= (DIVISION_SIZE - 2); y++ {
    flag := 0
```

```
    for Refcnt := 0; Refcnt <= (N-1); Refcnt++ {
      var DSqr = ((X[Refcnt] - MidptX[x]) * (X[Refcnt] - MidptX[x])) + ((Y[Refcnt] - MidptY[y]) * (Y[Refcnt] - MidptY[y]))
      if DSqr < (R_out[Refcnt]*R_out[Refcnt]) && DSqr > (R_in[Refcnt]*R_in[Refcnt]) {
        flag += 1
      }
    }
    switch flag {
    case N:
      IntsecX1st = append(IntsecX1st, MidptX[x])
      IntsecY1st = append(IntsecY1st, MidptY[y])
      SumIntsecX1st += IntsecX1st[intcnt1st]
      SumIntsecY1st += IntsecY1st[intcnt1st]
      intcnt1st += 1
      break
    case N - 1:
      IntsecX2nd = append(IntsecX2nd, MidptX[x])
      IntsecY2nd = append(IntsecY2nd, MidptY[y])
      SumIntsecX2nd += IntsecX2nd[intcnt2nd]
      SumIntsecY2nd += IntsecY2nd[intcnt2nd]
      intcnt2nd += 1
      break
    case N - 2:
      IntsecX3rd = append(IntsecX3rd, MidptX[x])
      IntsecY3rd = append(IntsecY3rd, MidptY[y])
      SumIntsecX3rd += IntsecX3rd[intcnt3rd]
      SumIntsecY3rd += IntsecY3rd[intcnt3rd]
      intcnt3rd += 1
```

```
        break
      case N - 3:
        IntsecX4th = append(IntsecX4th, MidptX[x])
        IntsecY4th = append(IntsecY4th, MidptY[y])
        SumIntsecX4th += IntsecX4th[intcnt4th]
        SumIntsecY4th += IntsecY4th[intcnt4th]
        intcnt4th += 1
        break
      case N - 4:
        IntsecX5th = append(IntsecX5th, MidptX[x])
        IntsecY5th = append(IntsecY5th, MidptY[y])
        SumIntsecX5th += IntsecX5th[intcnt5th]
        SumIntsecY5th += IntsecY5th[intcnt5th]
        intcnt5th += 1
        break
      default:
        break
      }
    }

  }
  //To avoid the Divide by Zero error in the Node position
   calculation
  if len(IntsecX5th) == 0 {
    intcnt5th = 1
  }
  if len(IntsecX4th) == 0 {
    intcnt4th = 1
  }
```

```
if len(IntsecX3rd) == 0 {
  intcnt3rd = 1
}
if len(IntsecX2nd) == 0 {
  intcnt2nd = 1
}
if len(IntsecX1st) == 0 {
  intcnt1st = 1
}
//Weights for COG method  (Need to analyse the weight techniques
 from different papers)
const (
W1st float64 = float64(4)/float64(11)
W2nd float64 = float64(3)/float64(11)
W3rd float64 = float64(2)/float64(11)
W4th float64 = float64(1)/float64(11)
W5th float64 = float64(1)/float64(11)
)
IntsecX_COG := (((SumIntsecX1st)/float64(intcnt1st)) * W1st+((
 SumIntsecX2nd)/float64(intcnt2nd)) * W2nd+((SumIntsecX3rd)/float64
 (intcnt3rd)) * W3rd+((SumIntsecX4th)/float64(intcnt4th)) * W4th+((
 SumIntsecX5th)/float64(intcnt5th)) * W5th)
IntsecY_COG := (((SumIntsecY1st)/float64(intcnt1st)) * W1st + ((
 SumIntsecY2nd)/float64(intcnt2nd)) * W2nd + ((SumIntsecY3rd)/
 float64(intcnt3rd)) * W3rd + ((SumIntsecY4th)/float64(intcnt4th))
 * W4th + ((SumIntsecY5th)/float64(intcnt5th)) * W5th)
```

```
}
```

Source Code A.10: Location Computation: Go Language Source File for Computing Location with Sophisticated Grid Method