Radio Frequency Localisation of RFID Tags in a Raspberry-Pi Sensor Network

Aleksandar Krastev

Master of Science
Computer Science
School of Informatics
University of Edinburgh
2013

Abstract

To Do

Acknowledgements

I thank my supervisor Dr Michael Rovatsos and Mr Michael Anslow for their support and patience.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Aleksandar Krastev*)

Table of Contents

Chapter 1

Introduction

Copied from IRP

Radio Frequency Identification (RFID) is an identification technology that also enables tracking of people and objects. RFID functions by remotely obtaining data stored on RFID tags. This information is mainly used for identification purposes. Systems relying on such data can only provide course-grained location information [Bouet and dos Santos, 2008]. Their RFID readers are positioned at strategic control points in order to recognise tags that enter their read range. However, if an object's identity is combined with its location, then the benefits of RFID could be greater. For example, patient care and hospital operations could be improved using remote identification and tracking of patients [Cangialosi et al., 2007].

RFID localisation principles are similar to the ones used for indoor wireless networks [Bouet and dos Santos, 2008]. There are certain differences between both technologies, which results in tracking methods that are altered to reflect the characteristics of RFID. This project uses some of these indoor localisation schemes to detect and track a tag using three reader nodes in a controlled indoor environment.

RFID systems mainly consist of tags and readers. While tags are simple devices, readers are more complex and usually require a connection to a host computer or network [Landt, 2005]. The high costs of tags and readers are a major factor that constrains the penetration of this technology [Want, 2006]. Nowadays, these devices are becoming affordable to users . In addition, the recent emergence of cheap and compact single-board computers, such as the Raspberry Pi[1], creates an exiting opportunity to build a cost-effective RFID sensor network capable of localising tags. This can be realised by connecting readers to single-board computers through USB or wired using

[1] About the Raspberry Pi - `http://www.raspberrypi.org/about`

a breadboard[2] and general purpose input/output (GPIO) pins on a chip.

On the one hand, the RFID technology has unprecedented advantages and it has gained the attention of big industries that have identified its potential. On the other hand, the high costs of RFID tracking systems and components prevent most people from using and developing the technology. The hardware combination of affordable readers, tags, and single-board computers has the potential to benefit a vast range of businesses but also do-it-yourself hobbyists and enthusiasts. This might result into improved automated handling and tracking of goods in a warehouse, for instance. It can also result in a fast-paced, community-based, and open-source development of the RFID technology applied in a wide range of scenarios. This project is interesting and exciting because it will try to apply RFID localisation algorithms on affordable hardware in order to create a tracking system. This will show that there can be cost-effective alternatives to commercial solutions, thus making the technology more accessible to a wider audience.

1.1 Hypothesis

Copied from IRP.

The hypothesis of this project is that existing algorithms for localisation and tracking of active tags can be applied on a cost-effective Raspberry-Pi-based sensor network to achieve a similar performance. More specifically, the purpose of the project is to construct and programme three reader nodes, each consisting of a reader connected to a single-board computer, that cooperate in an indoor environment to estimate the position of a stationary or moving active tag based on the Received Signal Strength Indicator (RSSI) using a localisation method called trilateration.

1.2 Contribution

1.3 Thesis Outline

1.4 Summary

[2] Breadboard is a solderless (plug-in) construction base used for experimenting with circuit design.

Chapter 2

Background

This chapter presents background information of the technologies and hardware used in this project. First, an introduction of RFID is provided in Section 2.1. This includes a description of the technology, typical components of such systems, and potential applications in the context of this work. Then, Section 2.2 discusses the Received Signal Strength Indicator (RSSI) as the metric for measuring distance between readers and tags. Next, the hardware components of the project are presented in Section 2.3. These are RFID readers, an active tag, and single-board computers (Raspberry Pis). Section 2.4 includes a discussion of location sensing techniques and evaluation criteria used throughout this project. This chapter is concluded by a survey of previous work using active RFID components to localise objects in indoor environments.

2.1 Radio Frequency Identification (RFID)

Radio Frequency Identification (RFID) is a wireless technology that communicates electronically stored data between radio devices. This information is used to remotely identify objects marked with tags [Hunt et al., 2007, p. 5]. RFID uses electromagnetic waves to carry information. Such systems differ from each other by their radio frequency of operation, physical coupling method, and transmission range [Finkenzeller, 2010, p. 21]. Radio frequencies used in RFID range from 100kHz to 10GHz [Landt, 2005]. The physical coupling methods in RFID classify such systems into three main categories. Close-coupling systems have a small interrogation range of up to 1cm. Remote coupled systems are capable of sensing information of up to 1m. All systems that can wirelessly read data from a marked object positioned over 1m away are called long-range systems [Finkenzeller, 2010, p. 22]. An RFID system has three compulsory

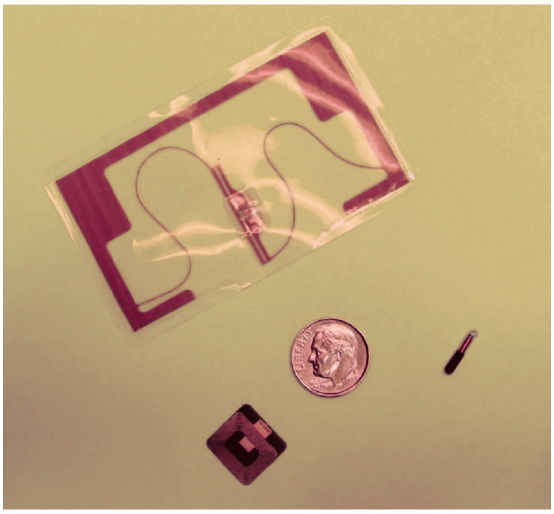


Figure 2.1: Three different variants of RFID tags. Figure from [Want, 2006].

hardware components, tags (also known as transpoders), readers (also called interrogators), and controllers.

2.1.1 RFID tags

A tag is a data-carrying device that transmits identification information in response to a received signal from a reader. RFID tags usually consist of an antenna attached to a microchip [Want, 2006, p. 2]. The hardware can be encapsulated in different types of enclosures. Tags come in different shapes and sizes depending on their operational environment (Figure 2.1). RFID tags also have memory where the identification and sometimes additional information is stored. Additional data might include a delivery date of a parcel, for example. The information stored on a tag is usually only for reading. However, there are implementations of the RFID technology that benefit from writing data to a tag. For instance, a pallet might have a tag attached to it that can store the content of the pallet as it changes over time [Hunt et al., 2007, p. 8].

2.1.1.1 Passive and active tags

Tags can be classified into two main categories, passive and active. Passive tags do not require a power source. They communicate with readers by reflecting part of received radio waves, a term referred to as backscatter modulation [Bolic et al., 2010]. They have a number of advantages, which include their small size, very long operational life, and low price. Nevertheless, passive tags need to be in the readers' range in order to operate. This is because passive tags obtain the power they need to supply their circuitry from an electromagnetic signal received from an RFID reader. A charge builds up into a capacitor that can power the passive tag and transmit the identification information it is storing [Weinstein, 2005].

In contrast, active tags require a power source in the form of a battery or are directly connected to the electrical grid [Want, 2006]. Although their lifetime might be limited by the available energy, active tags can be read from greater distances compared to passive tags. This is because they have their own power source which enables them to emit strong signals to the readers [Weinstein, 2005]. Compared to passive tags, active ones are larger in size and have a higher price.

2.1.2 RFID readers

A reader is a radio device that is capable of transmitting interrogation signals and capturing information send back by tags. The reader's transmission frequency specifies the operational frequency of the RFID system, which also defines their practical reading range [Finkenzeller, 2010]. These devices usually consist of a radio frequency (RF) module that is capable of sending and receiving signals, an antenna, and a control unit in the form of a microprocessor. RFID readers have the following main functions:

- read/write data from/to an RFID tag,
- power a passive tag,
- relay the obtained information to a host computer [Hunt et al., 2007, p. 9].

Readers are responsible for bringing additional functionality to an RFID system. This includes support for simultaneous sensing of multiple tags, authentication of tags to prevent unauthorised access to a system, and data encryption of the stored data to ensure integrity [Hunt et al., 2007, p. 10]. There is a wide range of RFID readers that differ in their operational radio frequency, range, and coupling method. These properties are formed by factors such as the specifications of the system, its budget, and security requirements [Finkenzeller, 2010, p. 25].

2.1.3 RFID controllers

The third component of an RFID system is the controller or server. It is a computer that is responsible for connecting and communicating with multiple readers, aggregating any incoming data, and processing it. Readers can be connected to the server using a network or serial connection. Identification information is usually stored in a database and is used by an application software [Hunt et al., 2007, p. 11]. Figure 2.2 shows the components of a typical RFID system.

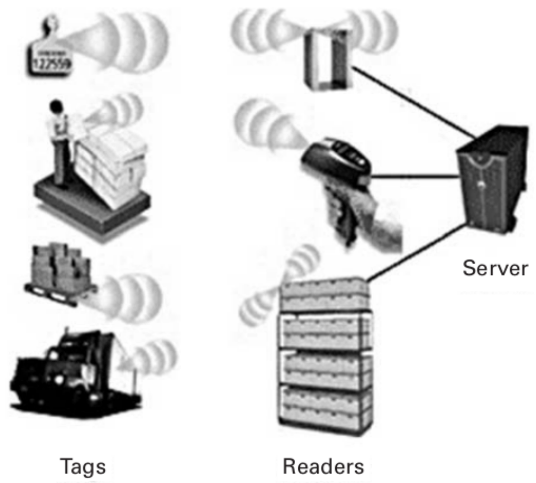


Figure 2.2: Components and applications of RFID. Figure from [Rida et al., 2010, p. 20].

2.1.4 RFID applications

RFID hardware is becoming more inexpensive, which creates a wide range of possible scenarios where this technology can be applied [Nath et al., 2006]. The most widespread applications are in tracking of objects or people, in supply chain and asset management, and in health services [Weinstein, 2005].

Passive RFID systems can be used as an alternative and improvement of the current standard for identification of products, the barcodes. Reading a barcode attached to an object requires a direct line of sight between a reader and a tag. In addition, barcodes can get obscured by other objects or substances, which hinders the identification process. RFID solves these disadvantages. A line of sight is not required when reading data from tags attached to objects. RFID tags also support a larger set of unique IDs compared to bar codes, can be reprogrammed, and can store additional data depending on the application requirements [Weinstein, 2005].

In the context of this project, which is concerned with location sensing, RFID has applications in tracking important objects or personnel and trying to pinpoint their position. For example, active RFID systems can be used in hospitals to monitor the location and life cycle of patients in an indoor environment [Cangialosi et al., 2007]. Expensive hospital equipment could be tracked so that it would be at the right place and time. Another possible scenario is to track school kinds while on school grounds in order to find lost children and monitor their attendance [Swartz, 2004].

2.2 Received Signal Strength Indicator (RSSI)

Some RFID readers provide an indication of the strength of radio signals received from tags. This metric is called Received Signal Strength Indicator (RSSI). Its value is often output along with the identity information stored in a tag. It is estimated at the reader side before amplifying the received input. RSSI is a unitless measurement of the power of the received signal represented as a positive value with certain resolution range. A resolution of three bits gives a precision of eight possible values for RSSI. This means that there are eight different steps of estimating how far a tag is. A resolution of eight bits, supported by the project hardware, lets readers output values between 0 and 255 giving a better approximation of the distance between a reader and a tag.

2.2.1 RSSI and RSS

RSSI is not to be confused with the Received Signal Strength (RSS), on which RSSI is based. RSS is usually measured in dBm[1]. It represents the attenuation of a received signal and is a function of the distance between a receiver and transmitter [Bouet and dos Santos, 2008]. In WiFi, the 802.11 standard does not define the relationship between RSSI values and reported signal power levels. It is up to the manufacturers to provide a conversion function or table that specifies range and accuracy of the RSS values and how these translate into a RSSI range between zero and a maximum value [Lui et al., 2011]. The above applies to RFID, which is also a radio technology.

2.2.2 RSSI and distance

A third relationship is the one between RSSI and distance. In other words, it is the problem of using RSSI reader measurements to estimate the distance separating a receiver and transmitter. More importantly, one might ask whether RSSI is a reliable

[1]dBm - Power ratio in decibels of power referenced to one milliwatt (mW) - http://en.wikipedia.org/wiki/DBm

parameter for localisation algorithms in wireless networks. This is not the main question that this work is concerned with. However, the reliability of this measure is of prime importance because here position estimation relies solely on this parameter.

On the one hand, there are studies that test the reliability of both RSS and RSSI for location sensing [Elnahrawy et al., 2004, Parameswaran et al., 2009]. These concluded that the limitations of determining inter nodal distances are fundamental. On the other hand, signal strength is readily available in devices today, which creates attractive opportunities for estimating position without any additional hardware. Indeed, there are a number of WiFi-based systems that rely on received signal strength including the Horus WLAN location system [Youssef and Agrawala, 2005], the EZ localisation system [Chintalapudi et al., 2010], an indoor location system using trilateration [Cook et al., 2005].

2.2.3 How RSSI fits in this project

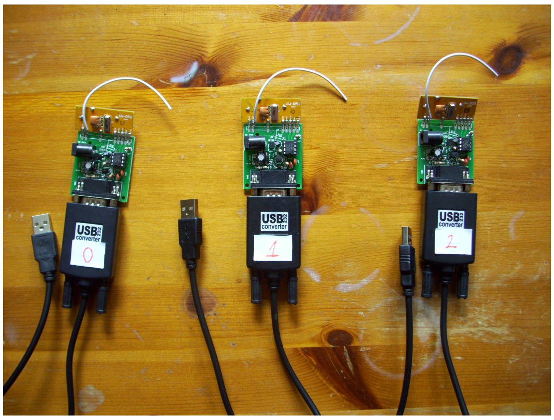
Although RSSI is not considered a reliable measure for distance due to the physical properties of radio waves and due to cluttered and dynamic indoor environments, the aforementioned systems show that researchers and engineers try to get the best of what already is provided. In the same manner, this work uses RSSI reader measurements as a basis for location sensing. This is done using a translation table that converts RSSI into a distance metric. The methods used and the challenges faced are discussed in Section 3.4.

2.3 Project Hardware

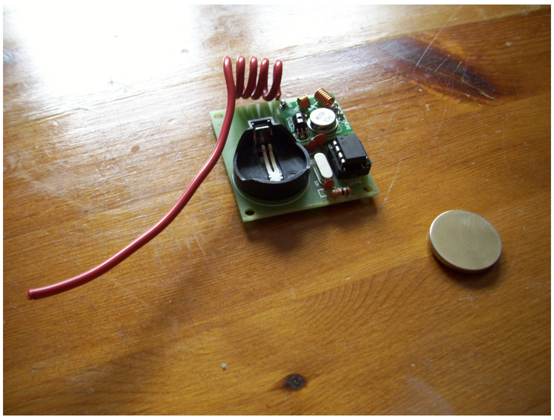
This project combines three types of hardware. The first is active RFID devices, the second - single-board computers, and the last is a commodity network infrastructure. The main hardware components can be seen on Figures 2.3 and 2.6. The following subsections present these devices and their specifications. Details on how the components are combined in forming the resulting system are given section 3.1.

2.3.1 RF9315R-u Active RFID 8 Meters Receiver with RSSI Module

The project uses three active RFID readers containing RSSI modules. Figure 2.3a shows the three readers, where each is connected to a serial to USB converter that



(a) Three active RFID readers



(b) One active RFID tag

Figure 2.3: RFID hardware used in this project.

offers a convenient interface to any computer with USB ports. For a discussion of the problems encountered while using the converters refer to 4.3.2.

The readers are superheterodyne receivers meaning they convert received signals to an intermediate frequency that is more convenient to process and gives a more stable design. The readers operate at 315 MHz, which lies in the lower band of Ultra High Frequency (UHF). Such waves propagate mainly by line-of-sight. They are blocked by large objects such as buildings, but can penetrate through a few building walls, which is enough for indoor location sensing. UHF are also sensitive to antenna orientations [Hunt et al., 2007, p. 15].

The readers get their power supply from a serial or USB connection (when serial to USB converter is used). The readers have a DC 9V socket that can be used to power the devices in case the above connections are not able to supply sufficient power. The receiver devices have a built-in watchdog timer of 2.3 seconds. The watchdog timer is used to detect hardware malfunction. The readers reset the time before it elapses to confirm that they are operating correctly. These receivers can read up to 80 tags simultaneously. There is not any anti-collision protocol. The readers rely on the tags to transmit their identification data every 2.5 to 3.0 seconds.

The RFID readers employ a simple communication protocol. The serial port settings for these devices can be seen in Table 2.1. Data are send in a raw character format without data encryption. The ID of a tag, consisting of four characters, is concatenated with a RSSI measurement, which could range from 0 to 255. For a discussion on the actual RSSI ranges observed during experiments refer to **REF**. Each new reading is separated by a space character. A sample input from the RFID readers is illustrated on

Figure 2.4.

Parameter	Value
Baud rate	9600 bits per second
Data bits	8 bits
Stop bits	1 bit
Parity	None
Flow control	None

Table 2.1: Serial port parameter settings to communicate with the readers.

Figure 2.4: RFID reader input on a communication port. The first four characters are the ID of a tag. The number concatenated to the ID is the RSSI value. Individual readings are separated by a space character.

The integrated RSSI module measures the received radio frequency signal over a range of 60 dBm. The manufacturer specifies that RSSI values vary between units[2]. Figure 2.5 shows how the radio frequency signal level on the x axis changes against the RSSI voltage on the y axis. In can be observed that the signal levels can be effectively measured between -55 and -115 dBm giving a range of 60 dBm. The readers' specifications described above are summarised in Table 2.2. These reader devices are a good fit for this project because of their low price, RSSI modules with good resolution, active RFID type, and USB connectivity.

[2] Ananiah Electronics active RFID reader - `http://www.ananiahelectronics.com/RF9315R-u.htm`.

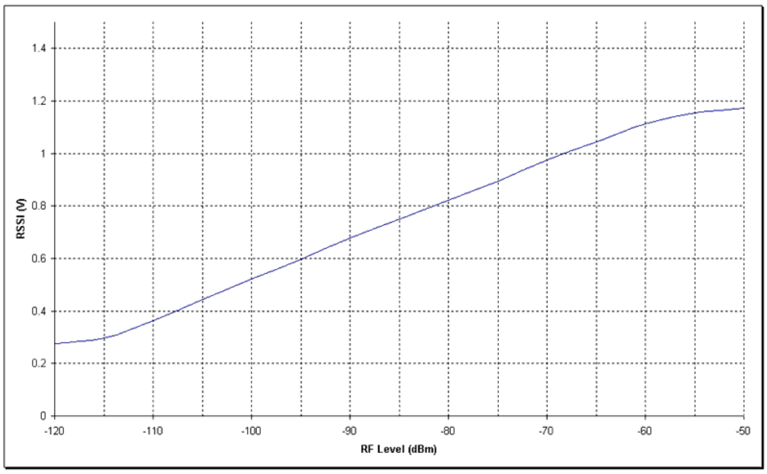


Figure 2.5: A plot of radio frequency signal levels against RSSI voltage. Figure from product website[2].

2.3.2 RF8315T Active RFID 8 Meters Transmitting Module

The active tag is a radio transmitting device that sends out its unique four character identification every 2.5 $\pm$ 0.5 seconds. Figure 2.3b shows the active RFID tag used in this work. Its transmission time is around 11ms giving a small probability of tags' signals colliding. The tag can use CR2025 and CR2032 batteries as a power supply. It can operate between 5,000 and 7,000 hours with a single battery. The tag consumes most of its power (4mA) while transmitting. During the rest of the time, the tag stays into hibernation mode using only 18uA.

The effective transmission range of the tag is estimated at eight meters by the manufacturer. For range measurements conducted during this project refer to **REF**. The tag arrived without an antenna. According to the specifications, to achieve its effective range, the antenna should have a 8mm coil diameter and 2cm coil length. The construction of the antenna is discussed in section 4.3.1. The tag specifications described above are summarised in Table 2.3. This RFID tag was chosen for this project because of its low price, transmission range, portable size, low power consumption, and matching operating frequency to the readers.

[3]Ananiah Electronics active RFID tag - `http://www.ananiahelectronics.com/RF8315T.htm`.

Specification	Value
Dimensions	4cm x 6cm x 1.8cm
Operating Temperature	0 - 50° C
Operating Frequency	315 MHz
Incoming signal range	60 dBm
Power source	Serial / USB port, DC 9V socket
Communication	RS-232 serial port
Watchdog timer	2.3 seconds
Simultaneous reads	80 tags
Reader control	No control protocol
Data representation	Raw character data, No data encryption
Data output	ID: 4 characters + RSSI: 0-255
Price	US $49.95

Table 2.2: Specifications of RF9315R-u active RFID reader. Table from product website[2].

2.3.3 The Raspberry Pi

A single-board computer is a computer that is built on a single circuit board. It features most of the components of a personal computer. It has a processor, memory, storage, different microprocessors, and input/output interfaces. The Raspberry Pi is a particular implementation of a single-board computer. This project uses three such devices in order to construct a location sensing system. Figure 2.6a shows these computers.

The Raspberry Pi has compact dimensions and is low on weight. It consumes little power, but has enough processing power, memory, and storage to run a standard operating system, such as the Raspbian Linux. The Raspberry Pi has two USB 2.0 ports as well as an Ethernet network port. In addition, the Pi has some characteristics of a development board employing a General Purpose Input/Output (GPIO) interface, which is could be used for connecting low-level peripherals such as RFID readers. The characteristics of the Raspberry Pi make it a great candidate for this project. Its specifications are summarised in Table 2.4.

[4]The Raspberry Pi website - `http://www.raspberrypi.org/faqs`.

Specification	Value
Dimensions	4cm x 5cm x 1.8cm
Operating Temperature	0 - 50° C
Operating Frequency	315 MHz
Power source	CR2025 / CR2032 battery
Battery life	5,000 / 7,000 hours
Power consumption	4mA when transmitting, 19uA when idle
RF output power	$<$ 2mW
Effective range	8 meters with 8mm coil diameter, 2cm long antenna
Data output	ID: 4 characters
Price	US $19.95

Table 2.3: Specifications of RF8315T active RFID tag. Table from product website[3].

2.4 Location sensing

Estimating the position of an RFID transmitter using three receivers is the main goal of this project. This section describes a localisation technique that is suitable for the available hardware. It also defines criteria for evaluating estimated positions.

2.4.1 Trilateration

Trilateration is a localisation technique that uses the geometric properties of triangles in order to compute locations [Hightower and Borriello, 2001a]. This technique has practical applications in surveying and global positioning systems. Trilateration requires the known locations of two or more reference nodes as well as the distance measurements between a reference node and the unknown object [Zhang et al., 2009, p. 280]. In this project, the distance from a reader to a tag is the radius of a circle that could be drawn around the reader. The intersection of circles of three readers can be used to determine the approximate location of a tag relative to the readers. In order to compute the position of an unknown object in two dimensions, trilateration requires three reference points that are non-collinear [Zhang et al., 2009]. In a three dimensions case, four non-coplanar nodes and their distance measurements are needed [Hightower and Borriello, 2001a]. Figure 2.7 shows a graphical representation of the

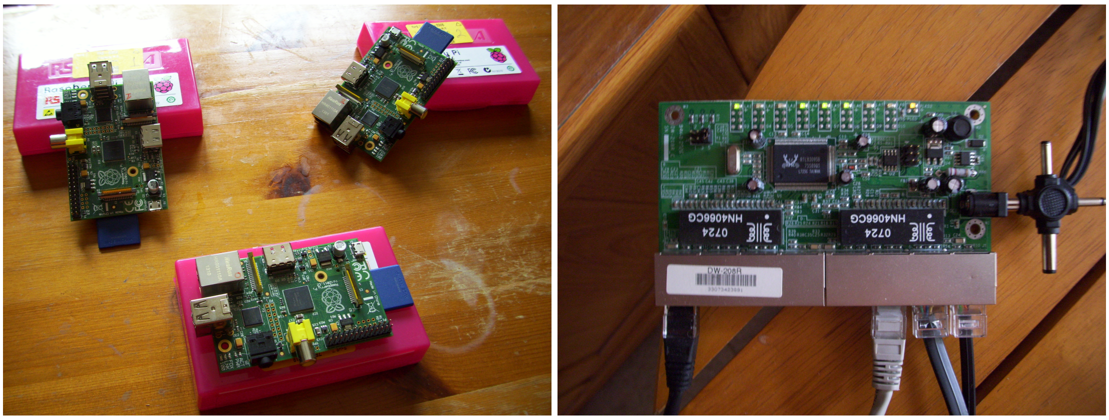
Specification	**Value**
Dimensions	86mm x 54mm
Weight	45g
Power source	5V MircroUSB or GPIO
Power rating	700mA (3.5W)
System on a chip	Broadcom BCM2835
CPU	700MHz ARM1176JZF-S
GPU	Broadcom VideoCore IV 250MHz
Memory	512MB
Storage:	SD card slot
USB 2.0 ports	2
Networking	10/100 Ethernet
Low-level peripherals	8 x GPIO, UART, I^2C bus, SPI bus
Operating system	Raspbian Linux
Price	US $35

Table 2.4: Specifications of the Raspberry Pi Model B revision 2 single-board computer. Table from product website[4].

concept of trilateration. Section 3.5 presents the mathematical method for estimating the position of unknown object in three dimensions.

2.4.2 Evaluating an estimated position

An important requirement for every location system is to estimate positions consistently and accurately. Hightower [Hightower and Borriello, 2001a] proposed criteria for the classification of such systems based on properties including accuracy, precision, and distribution of erroneous positions to the true one. Such metrics can be used when evaluating a location system's performance in terms of how often it locates an object within some distance of its true location. For instance, if a GPS receiver can locate its position within five meters for 90 percent of the time, then its accuracy is five meters and has a precision of 90 percent for that accuracy [Hightower and Borriello, 2001b]. There is a trade-off between accuracy and precision. To achieve a higher precision, one might have to sacrifice accuracy. As a result, in order to arrive at a concise quantitative summary of these attributes, Hightower proposed to assess the error distribution



(a) Three Raspberry Pis (b) LAN switch

Figure 2.6: Computer and network hardware used in this project.

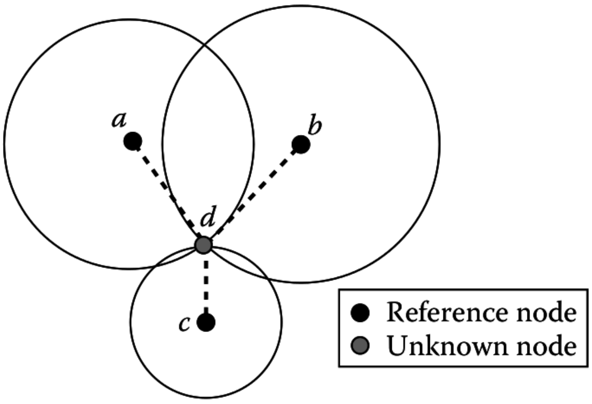


Figure 2.7: The trilateration technique for positioning an unknown node based on distance measurements from three reference nodes. Figure from [Zhang et al., 2009, p. 281].

accumulated during a system's operation [Hightower and Borriello, 2001b]. In addition, this should be combined with other parameters such as the number of nodes in the system, the density of the infrastructure, and the size of the indoor space. In this work, these metrics are used to evaluate the localisation performance of the resulting system.

2.5 Previous work

The RFID technology is of substantial interest to researchers. Location sensing using active RFID devices is a specific subcategory of this field. Nevertheless, there have been a number of important research efforts to construct localisation systems using the available RFID hardware at that time. This section presents some of the previous work that directly relates to this project.

2.5.1 SpotON

SpotON is a fine-grained tagging technology for 3D location sensing using radio signal strength analysis [Hightower et al., 2000]. The author's motivation was to develop a low cost system compared to commercial solutions available at their time. SpotON's operation involves a number of readers collecting signal strength information from active tags to determine their positions in the 3D space. It was the author's believe that the accuracy and efficiency of location sensing could be enhanced by sensor fusion, i.e. adding more sensors (accelerometers) and building proximity maps [Hightower et al., 2000].

2.5.1.1 Operation

The SpotON algorithm consists of two main parts. First, RSS measurements are converted into a distance estimation. This is done using a translation function relying on numerical variables that were identified based on observation. This function is hardware-specific and cannot be applied in this project. Second, distance measurements are used as an input to a localisation algorithm that tries to minimise RSS errors [Hightower et al., 2000]. It is based on the lateration geometrical process to estimate a position for an active RFID tag.

2.5.1.2 Limitations

At the time or their research, Hightower and his colleagues were using RFID hardware with 2-bit accuracy when measuring received signal strength [Hightower et al., 2000]. They identified that this accuracy is not enough to achieve the precision required for localisation in small indoor environments. The authors mentioned that 8-bit accuracy (supported by this project's hardware) could be used in the future for improved performance [Hightower et al., 2000]. Another limitation was the frequency of collecting measurements. This would take between 10 and 20 seconds, which is generally too slow for monitoring real-time position changes of objects. These drawbacks were solved by creating custom RFID hardware.

2.5.1.3 Results

When localising a tagged object, the SpotON system achieved accuracy of three meters using off-the-self hardware [Hightower et al., 2000]. Relying on their custom RFID devices, SpotON reported under one meter location sensing accuracy.

2.5.2 LANDMARC

LANDMARC is a 2D location sensing system that uses RFID for locating objects inside buildings. The major advantage is that it improves the overall accuracy of locating objects by using reference tags [Ni et al., 2004]. The authors believed that the choice of technology and techniques is of crucial importance for the granularity and accuracy of the location information. They identified that the range of an RFID system is determined by the power available at the tags, indoor topology, and environmental conditions. Ni and his colleagues found out that instead of using a lot of readers, they can arrange a number of tags in a 2D rectangular grid to use as reference tags [Ni et al., 2004]. The advantage is that tags are cheaper than readers. Also, reference tags are subject to the same environmental factors as the tags being tracked. The authors argue that the placement of readers and reference tags is very important for the accuracy of the system [Ni et al., 2004].

2.5.2.1 Operation

The core idea of LANDMARC is to select the k nearest reference tags that are closest to the unknown tag using differences in RSS measurements. Having identified the k nearest reference tags, their known positions are used to localise the unknown tag. Distances between tags are computed using Euclidean distance. The system also applies a weighing factor when computing coordinates, where small distances receive a bigger weight.

2.5.2.2 Limitations

The hardware problems of the current RFID technology were identified [Ni et al., 2004]. RFID hardware used in LANDMARC did not supply signal strength directly, which resulted in unnecessary processing and sacrificed accuracy. LANDMARC took a substantial time to estimate locations. Two factors were contributing to these problems. One being the scanning time of the readers in order to collect signal strengths. The second, the time interval of a tag emitting its identification information, which could not be controlled. Ni and his colleagues also measured different power levels from two tags placed at identical positions. This resulted in unstable system behaviour.

2.5.2.3 Results

The authors experimented with different number and placement of readers, reference and tracking tags. The best setup was consisting of four readers and one reference tag per square meter resulting in an average distance error of one meter [Ni et al., 2004].

2.5.2.4 Extension systems

VIRE extends the methods used in LANDMARC by defining virtual reference tags and a proximity map that every reader records [Zhao et al., 2007]. This proximity map consists of a 2D grid of reference tags where the centre of a cell is a tag. The difference in the RSS measurements between reference and unknown tag helps label cells in the proximity map so that it can be constructed. The union of individual proximity maps gives a global proximity map for the unknown tag. Experimental results showed an improvement of LANDMARC's precision between 17 and 73 percent for different scenarios and indoor environments [Zhao et al., 2007].

LANDMARC is a location sensing system that reports a two dimensional tag positions. The extended 3-D LANDMARC algorithm is a system that could localise tags in three dimensions [Khan and Antiwal, 2009]. A major difference is the use of passive tags instead of active ones. This system solves some of the original limitations of LANDMARC by using hardware providing received signal strength directly. The authors rely on similar methodology, but extend computations to three dimensions. The accuracy of the system was estimated at around 0.5 meters when employing three readers, two tracking tags, and 11 reference tags in an 11 cubic meter space [Khan and Antiwal, 2009].

2.6 Summary

This chapter presented background information of the technologies and hardware used in this project. First, an introduction of RFID is provided in Section 2.1. Then, Section 2.2 discusses the Received Signal Strength Indicator (RSSI). Next, the hardware components of the project are described in Section 2.3. Section 2.4 includes a discussion of location sensing techniques and evaluation criteria. This chapter is concluded by a survey of previous work relating to this project.

Chapter 3

Methodology

This chapter gives the reader a detailed overview of the system components and approach to the problem. Section 3.1 presents the hardware setup of the system. Section 3.2 describes the problem this project is trying to solve. Then, the overall software architecture is discussed in section 3.3. Section 3.4 details the methods for converting RSSI to distance measurements. Finally, section 3.5 describes trilateration, a mathematical technique that computes the relative position of an unknown object to three reference objects.

3.1 Hardware Setup

The devices used in this project were connected together to form the hardware setup of the system. These components and their specifications are described in detail in section 2.3. Figure 3.1 is a diagram illustrating how the hardware devices were attached to each other.

3.1.1 Reader nodes

The building blocks of the system are the reader nodes. Each consists of an RFID reader connected to a Raspberry Pi using USB. The project plan had provisioned extra time to account for the possible complications when interfacing between serial devices and single-board computers. The Raspberry Pis have USB ports that cannot power all kinds of USB devices. Another issue was whether the computers were capable of recognising and communicating with these specific RFID readers. Fortunately, these potential problems did not occur during this project. The RFID devices read identifi-

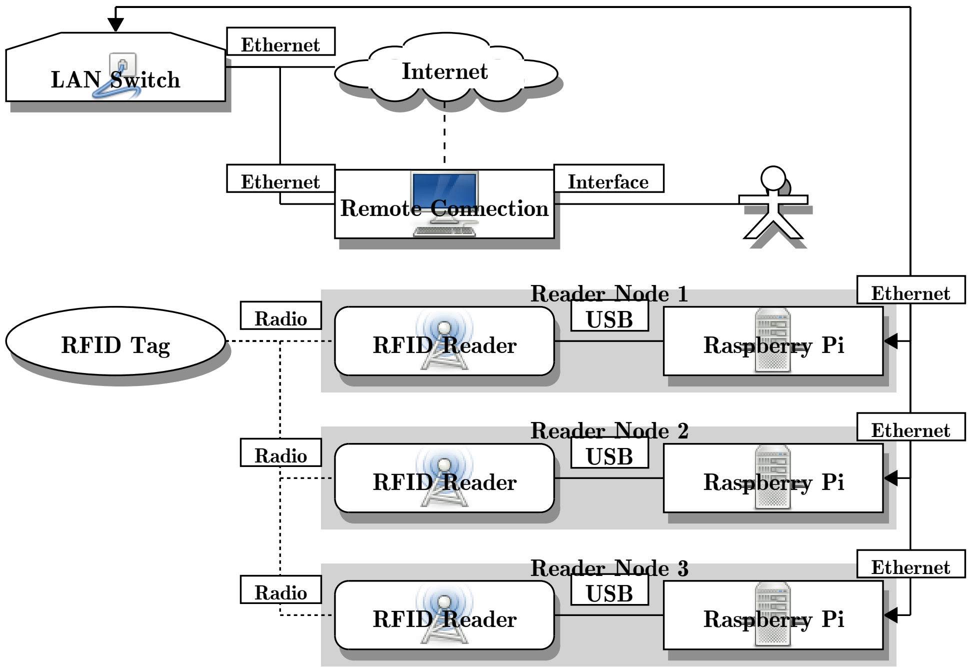


Figure 3.1: Hardware setup of the project.

cation data from the RFID tag over the air.

3.1.2 Computer Network

The Raspberry Pis were connected in a local area computer network (LAN) using Ethernet cables and a network switch. This allowed them to communicate readings to each other. The Raspberry Pi network provided means for remotely accessing, programming, and controlling the reader nodes. Remote access was established by connecting to the network using the network switch or through a wide area network, such as the Internet.

3.1.3 Alternative connectivity using Wi-Fi

During the project planning phase, an alternative connectivity method was considered. The Raspberry Pis have two USB ports. One of them was occupied by a RFID reader. The spare one could have been used for a wireless network adapter. Then, all reader nodes could connect to a wireless router or form an ad hoc network in order to communicate measurements. This way all Ethernet cabling and the network switch are not required giving a greater flexibility when positioning the reader nodes. Figure A.1 in Appendix A.1 shows this alternative hardware setup.

There are two matters that complicate this choice of connectivity. First, there is a high probability that the power fed to the USB ports is not enough to supply both an RFID reader and a Wi-Fi adapter. In which case, three powered USB hubs were needed to be purchased. Second, the Raspberry Pis have integrated Ethernet ports. Buying three wireless USB dongles is an unjustifiable expense in case these might not get powered by the single-board computers.

3.2 Problem Definition

Given the hardware setup of the system consisting of three reader nodes connected into a computer network, the problem it is trying to solve is to estimate the relative position of an RFID tag to three RFID readers. The system should accept the following inputs:

- identification information received from the RFID tag,
- RSSI measurements computed at the RFID readers,
- and positions of the reader nodes in three dimensions.

The system should output the location of the RFID tag relative to the reader nodes. The system should be able to estimate the tag's position when it is:

- stationary,
- in motion,
- unobstructed from any objects,
- occluded by a single or multiple objects.

Combinations of the above cases provide a basis for experiments to check whether the problem is solved by the system. To determine how well the system performs, the accuracy of localisation is compared to previous systems described in section 2.5.

3.3 Software Architecture

The system consists of three network nodes that need to aggregate reader measurements on a single computer in order to process the data. A server-client model was chosen because the measurements are collected in one place, which is convenient for converting RSSI values to distance and computing the tag's position. Figure 3.2 shows a conceptual diagram of the server-client model used in this project. Every Raspberry Pi is capable of being both a server and a client. A disadvantage of this model is the single point of failure of the system. If the server fails, then another node needs to become an aggregator of data otherwise the system would not function as intended.

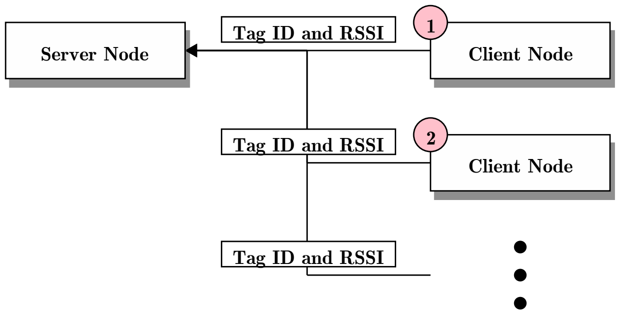


Figure 3.2: Conceptual diagram of the server-client model.

Client nodes have two main responsibilities. First, they read the identity and RSSI of the tag. Second, clients send this data on the network to the server. A server node provides the following functionality. It receives measurements from other nodes. A server also reads the tag's identity and RSSI. Then, this computer converts the RSSI values into distance. Distance measurements together with the positions of the reader nodes are input into a localisation algorithm. These steps are illustrated on Figure 3.3.

3.4 Converting RSSI to distance

One of the main challenges of this project was to find a reliable way of converting RSSI values into distance from RFID readers to a tag. As discussed in Section 2.2.2, studies have shown that RSSI is not a reliable or accurate measure of distance. Nevertheless, RSSI is one of the main parameters of this system and distance estimation is solely based on it.

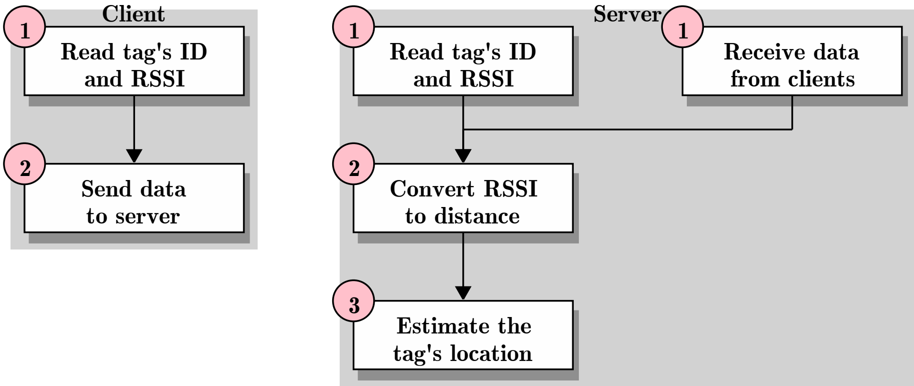


Figure 3.3: Diagram of the general server and client responsibilities.

3.4.1 Free-space Path Loss

The first attempt to provide a conversion between RSSI and distance was relying on the inverse-square physical law and free-space path loss (FSPL) formula. In free space propagation, electromagnetic waves obey the inverse-square law stating that the intensity of the emitted radiation is inversely proportional to the square of the distance from the source of the emitted radiation [Schlaikjer and Westman, 1962, p. 19]. This can be expressed as the following relation:

$$Intensity \propto \frac{1}{distance^2}$$

A more complete relationship between signal strength and distance is given by the FSPL formulation. Free-space path loss is the loss in signal strength of an electromagnetic wave propagating through free space without any obstacles [Balanis, 2012]. FSPL is proportional to the square of both distance and frequency of the radio signal. It can be expressed in terms of decibels given distance in meters and radio frequency in megahertz[1]:

$$FSPL(dB) = 20\log_{10}(d_{meters}) + 20\log_{10}(f_{MHz}) - 27.55$$

Rearranging the terms of the equation to find the distance gives:

$$d_{meters} = 10^{(FSPL(dB) - 20\log_{10}(f_{MHz}) + 27.55)/20}$$

[1]Derivation the dB version of the FSPL equation - http://www.ece.uvic.ca/~peterd/35001/ass1a/node1.html

Section 2.2.1 discussed the relationship between RSSI and received signal strength (RSS). RSSI is a unitless measurement derived from the values of the RSS. In Section 2.3.1, Figure 2.5 shows the relationship between RSSI and RSS for the RFID receivers used in this project. Consequently, RSSI values can be converted to received signal strength expressed in dBm and inserted into the equation above as $FSPL(dB)$. The frequency of the RFID project hardware is $315MHz$ (see Table 2.2 and 2.3).

There are three problems with this approach. First, it models how the signal strength decreases in a line-of-sight propagation in a free from obstacles environments. This project is concerned with location sensing in indoor spaces making the free-space path loss formula inappropriate for this setting. Second, computing the distance for the minimum ($0dB$) and maximum ($60dB$) values of the signal strength range results in distances 0.075 and 75.716 meters, which is beyond the practical range of the RFID devices. Third, the conversion from RSSI to RSS might not be appropriate in this case. The RFID equipment for this project is more affordable compared to commercial RFID equipment[2]. As a result, the online support and specifications are scarce, which raises the question to what extent these can be trusted. Moreover, the manufacturer claims a RSSI resolution of eight bits outputting values between 0 and 255. During the range experiments with this hardware a much smaller resolution was recorded. Consequently, converting RSSI unitless values to received signal strength in dBm cannot be relied upon. For details on the evaluation of the RFID devices refer to **REF**.

3.4.2 Translation table

Rather than relying on the physical relationship between electromagnetic waves and distance, a simpler and more direct approach was taken. For each RFID reader, a translation table was constructed mapping RSSI to distance. According to the manufacturer, but also observed in hardware experiments, reader measurements vary between different devices. There are two reasons for this. First, the RFID readers are handmade, which introduces small differences in their circuits. Second, the devices are not calibrated to each other when being built.

The methodology for constructing these translation tables relies entirely on RSSI measurements collected while evaluating the RFID devices. A number of experiments were conducted testing how RSSI changes as the distance between a reader and tag increases. In order to account for the characteristics of indoor environments, these

[2] How much do RFID readers cost today? - `http://www.rfidjournal.com/faq/show?86`

experiments have taken into account the orientation of the tag to the reader, line-of-sight propagation versus obscuring the tag with an obstacle, and elevation of the tag to the reader. Combining measurements from different experiments gives a more realistic representation of how RSSI changes in an indoor environment. Table 3.1 presents the translation table constructed for the first reader. Similar tables were developed for the other two devices (see Appendix A.2).

Distance	0	1	1	2	2	3	3	4	4	5	5	6	6	7
RSSI	80	65	64	62	61	57	56	53	52	48	47	46	45	44

Table 3.1: RSSI value ranges used to estimate distance when using the first reader.

As an example, the first two columns of Table 3.1 have the following meaning. When the reader and tag antennas were touching the average RSSI value from all experiments was 80. When the tag was one meter away from the reader, the average RSSI value of all experiment measurements was 65. RSSI values between 80 and 65 are linearly converted to the new range from zero to one meters as follows:

$$RSSI\ range = RSSI\ min - RSSI\ max$$

$$Distance = Distance\ min + (RSSI\ min - RSSI\ value)\ /\ RSSI\ range$$

An obvious limitation of these conversions is the size of the RSSI ranges. For example, there are 15 possible values that the first reader could measure when the tag is between zero and one meters away from it. In contrast, for a distance between six and seven meters the RSSI values change only by one unit. Following from that, the granularity of the distance estimation changes depending on the range of the RSSI measurements. This is caused by a hardware limitation of the readers when measuring RSSI and has been found during the range experiments presented in **REF**.

There is another factor contributing to the accuracy of this conversion. The RFID tag is using a battery for its power supply. During continuous operation the battery power level gradually drops, thus the tag emits a weaker radio signal as the battery is being depleted. This is reflected in the RSSI measurements making the translation tables inaccurate. In order to account for the RSSI changes, the incoming measurements were increased by an integer factor chosen based on observation. Different factors were selected for each reader due to their measurement differences.

In summary, this conversion method is not universal and probably cannot be applied to other brands of RFID devices. There are numerous factors that contribute to the variations in RSSI such as radio signal reflection, multi-path fading, and shadowing, to name a few. Nevertheless, this method was selected to translate RSSI to distance. Once the translation tables are constructed, it is a matter of calibration. As mentioned above, RSSI to distance conversion is one of the main variables of this system. How these translation tables performed is discussed in Section **REF**, where the evaluation results of the system in terms of localisation accuracy are presented.

3.5 Trilateration

Trilateration is a localisation method computing the position of an unknown object based on range measurements from three reference points at known locations. The concept of this technique was described in Section 2.4.1. This section presents the mathematical technique used in this project that provides an exact and computationally efficient solution for three-dimensional position estimation. The solution is based on the work of Manolakis [Manolakis, 1996] and the Wikipedia article on trilateration [Wikipedia, 2013].

3.5.1 Special case

In three-dimensional Cartesian coordinate system, the equations for three spheres are:

$$
\begin{aligned}
r_1^2 &= (x-x_1)^2+(y-y_1)^2+(z-z_1)^2 \\
r_2^2 &= (x-x_2)^2+(y-y_2)^2+(z-z_2)^2 \\
r_3^2 &= (x-x_3)^2+(y-y_3)^2+(z-z_3)^2
\end{aligned}
$$

where the sphere centres are $\vec{p}_1 = (x_1, y_1, z_1)$, $\vec{p}_2 = (x_2, y_2, z_2)$, $\vec{p}_3 = (x_3, y_3, z_3)$ and r_1, r_2, and r_3 are the sphere radii. Solving these equations for x, y, and z would give their intersection point. However, this is hard to do directly. In order to simplify the calculations, a special case is defined that can be later used as the basis for a general solution. There are three requirements of the special case. First, the three centres of the spheres are on the $z = 0$ plane, hence working in two dimensions. Second, one of the centres of the spheres, $\vec{p}_1$, is located at the origin. Third, another sphere centre, $\vec{p}_2$, is on the x-axis, thus two of the spheres are collinear. The equations for the three spheres can be rewritten as follows:

$$r_1^2 = x^2 + y^2 + z^2 \tag{3.1}$$

$$r_2^2 = (x-d)^2 + y^2 + z^2 \tag{3.2}$$

$$r_3^2 = (x-i)^2 + (y-j)^2 + z^2 \tag{3.3}$$

where d is the distance between sphere centres $\vec{p}_1$ and $\vec{p}_2$ and i and j are the signed magnitudes of the x and y components of the vector from $\vec{p}_1$ to $\vec{p}_3$. Figure 3.4 illustrates these components and the positions of the spheres in the $z = 0$ plane.

To solve these equations, first subtract 3.2 from 3.1 and solve for x:

$$x = \frac{r_1^2 - r_2^2 + d^2}{2d}$$

Next, subtract 3.3 from 3.1 and solve for y:

$$y = \frac{r_1^2 - r_3^2 + i^2 + j^2}{2j} - \frac{i}{j}x$$

Finally, use 3.1 to solve for z:

$$z = \pm\sqrt{r_1^2 - x^2 - y^2}$$

These three equations provide the coordinates of the intersection point (x, y, z) of the three spheres.

3.5.2 General solution

The solution of the aforementioned case cannot be applied in a general three-dimensional case because its requirements are not met. This is overcome by treating the sphere centres, $\vec{p}_1$, $\vec{p}_2$, and $\vec{p}_3$, as vectors from the origin:

$$\vec{p}_1 = (x_1, y_1, z_1)$$
$$\vec{p}_2 = (x_2, y_2, z_2) = \vec{p}_1 + \hat{e}_x d$$
$$\vec{p}_3 = (x_3, y_3, z_3) = \vec{p}_1 + \hat{e}_x i + \hat{e}_y j$$

where $\hat{e}_x$, $\hat{e}_y$, and $\hat{e}_z$ are the basis unit vectors in the x, y, and z original coordinate system; d, i, and j are the same variables defined above. The unit vectors and variables are expressed as follows:

$$d = \|\vec{p}_2 - \vec{p}_1\|, \qquad i = \hat{e}_x \cdot (\vec{p}_3 - \vec{p}_1), \qquad j = \hat{e}_y \cdot (\vec{p}_3 - \vec{p}_1)$$

$$\hat{e}_x = \frac{\vec{p}_2 - \vec{p}_1}{d}, \qquad \hat{e}_y = \frac{\vec{p}_3 - \vec{p}_1 - \hat{e}_x i}{\|\vec{p}_3 - \vec{p}_1 - \hat{e}_x i\|}, \qquad \hat{e}_z = \hat{e}_x \times \hat{e}_y$$

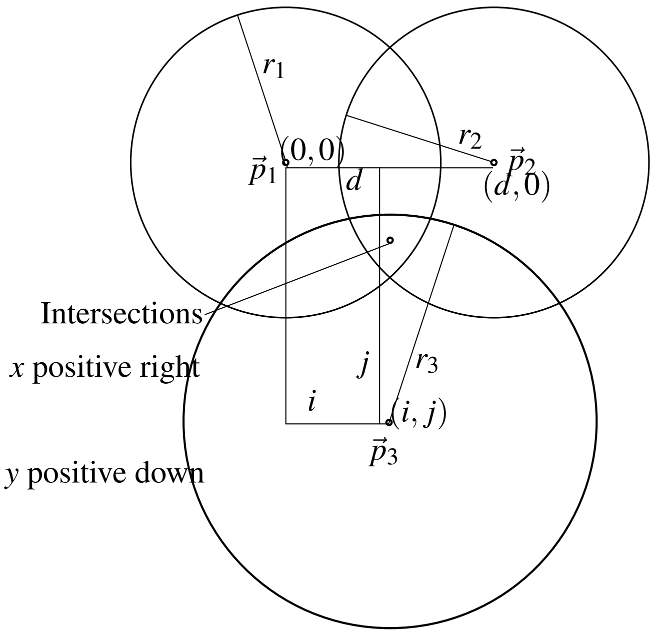


Figure 3.4: Figure showing three intersecting spheres in the plane containing their centres. Figure from [Wikipedia, 2013].

Thus, the intersection point of the three spheres is:

$$\vec{p}_{1,2} = \vec{p}_1 + \hat{e}_x x + \hat{e}_y y \pm \hat{e}_z z$$

The number of solutions to this problem depend on the z component. Recall that $z = \pm\sqrt{r_1^2 - x^2 - y^2}$. Consequently, there is only one intersection point when z equals zero. If z equals to plus or minus the square root of a positive number, then there are two intersection points. In case z equals to the square root of a negative number, no solutions exist.

There are a number of special cases that need to be taken into account. For instance, when d is zero, the sphere centres $\vec{p}_1$ and $\vec{p}_2$ share the same coordinates, which means that these two spheres are concentric. Consequently, no solution to the problem exists because these spheres cannot intersect when they have different radii and are the same sphere if they have equal radii. In another case, if the three reader nodes are positioned in a line ($j = 0$), then their spheres are collinear, which violates one of the requirements of the technique. Nevertheless, an intersection point $\vec{p}_1 \pm \hat{e}_x r_1$ exists, if:

$$\|\frac{\vec{p}_2}{\vec{p}_1 \pm \hat{e}_x r_1}\| - r_2 = 0, \quad \|\frac{\vec{p}_3}{\vec{p}_1 \pm \hat{e}_x r_1}\| - r_3 = 0$$

This mathematical method is used to estimate the position of an RFID tag. It is efficient because trigonometric functions are not used. Instead, this method relies on elementary arithmetic operations [Manolakis, 1996]. Section 4.7.2 is concerned with the implementation details of this method.

3.6 Summary

This chapter presented the overall hardware and software architecture of the system. It also described the methodology for converting RSSI to disntace and computing the relative position of an unknown object to three reference points using their distance estimations to this object. The next chapter details the hardware issues that were faced. In addition, it explains how the system was implemented in software.

Chapter 4

Implementation

This chapter describes how the system was built. Sections 4.1 and 4.2 present the software engineering tools and practices employed to aid the development process. Section 4.3 explains the hardware problems that were encountered and how these were dealt with. Next, section 4.4 goes through the steps taken to construct each reader node. Section 4.5 is concerned with how data was stored and managed in this system. Then, section 4.6 describes how the reader nodes communicated RFID measurements over the network. Section 4.7 explains how the methods for converting RSSI to distance and estimating the tag's location were implemented. Finally, section 4.8 presents the web interface that was used to visualise the outputs of the system.

4.1 Project management

A number of considerations were taken into account when deciding how to manage this project. First, the system operates using a server-client model. This means that different software components are executing on multiple processing nodes. As a result, changes in one node need to be propagated in the whole system ensuring the consistency of the software. Second, the software implementation is making use of different programming languages, multiple programming libraries, and a database management system. In order to ensure an iterative development process, where software components are constructed, debugged, and packaged together, it was decided to use the GIT version control system[1]. This system keeps a distributed repository of all software and database files so that each node stores a copy of not only the whole software system, but also a complete history of changes. In addition, the use of a version control system

[1] GIT version control system - `http://git-scm.com/`

stimulates the developer to merge a number of important changes into versions of the software. In this way, it becomes easy to track and monitor the project progress. The source code and documentation were hosted in a private repository on GITHUB[2] with access granted to the people involved in developing and supervising the project.

4.2 Software Engineering Practices

A number of software engineering practices were of significant help when developing the RFID location sensing system. This section presents them and explains the problems that they solve.

4.2.1 Project decomposition

It would have been a serious challenge to approach the project's task directly. The system consists of pieces of hardware that had to be orchestrated to solve a common problem. Therefore, it was very important to identify the system's components from early on. Hierarchical relationships between these parts were also defined. These steps ensured that the project could be divided into stages in order to systematically solve the main task. Regular deliveries of working components provided a more manageable way of constructing the final solution. For example, the work plan, devised before the start of the project, consisted of the following key activities:

1. Prepare the single-board computers
2. Construct functional RFID reader nodes
3. Receive information from the active RFID tag
4. Establish a network communication between nodes
5. Develop the localisation algorithm

Iterative construction of the system aided the development process. Problems and challenges were appearing gradually which helped solving them one at a time.

[2]The private project repository - `https://github.com/sandio/raspi-rfid-tracking`

4.2.2 Object-oriented design

This location sensing system is a combination of different software technologies. For instance, the system required an interface between a single-board computer and an RFID receiver. It also required means of communication between processing nodes. Logically, these and other requirements could be grouped into sets of functions, which is a motivation for employing an object-oriented design. This software methodology was used from the beginning of the project. Similar functionality is organised in a class. A class is responsible for all procedures concerning a particular part of the system. As a result, software is split into categories of functions, which makes it easy to address the class in charge of certain functionality.

Another benefit of the object-oriented design is modularity. For example, once input data is collected from all nodes it could be processed by a localisation algorithm in order to estimate the tag's position. Trilateration was chosen as the technique for computing locations. Object-oriented software development provides an easy way to experiment with different algorithms by substituting one class with another.

4.2.3 System scalability

In this project, three single-board computers collaborate by exchanging RFID readings to localise a tagged object. Three reference points are needed in order to use trilateration in two dimensions [Zhang et al., 2009]. Nevertheless, more reader nodes could be used, in case multileration is implemented, to give a better approximation of a tag's position. Another scenario involves nodes disconnecting and later reappearing into the network. These possible cases show the dynamic nature of the system. It could scale up as the system grows, but also scale down if a reader node is faulty. This is an important property of the system, which was noted at the start of the project. To ensure scalability of the server-client model, the multi-threading programming model was used. It allows multiple threads to exist within the context of a single process. As a result, the system could concurrently receive RFID measurements from multiple reader nodes, update data structures, and compute the location of the unknown object.

4.2.4 Documentation

Writing documentation was an important part of this project. The source code of the system has been systematically documented throughout the development process. Using the inline comments specifying how the software components work, an Application Programming Interface (API) was constructed using SPHINX[3], a PYTHON documentation generator. The API contains specifications of data structures, variables, and functions. It is a valuable source of information that provides a quick reference of how the system's components work and interact with each other. In addition, a manual for future users of the system was written. It gives a quick introduction of how to set up and use the system. The API and user manual can be viewed in Appendix **REF TODO**.

This project consists of both hardware and software components. In order to clearly understand how hardware components are connected and how software objects interact, a number of diagrams were used in Chapter **REF** and in the user manual. These diagrams were generated using BLOCKDIAG [4] , a diagram image generator written in PYTHON.

4.3 Hardware Issues

This section describes two hardware problems that were identified while the project was running. It also explains how these were solved.

4.3.1 Antenna Design

When the hardware equipment for this project arrived, everything was in order except for the RFID tag. It was missing its antenna, which is a coiled wire. The antenna needed to have specific length ($2cm$) and width ($8mm$) of the coil. The tag was being detected by the RFID readers, but only in close range. Consequently, an antenna was required for the project to continue. A number of antennas were designed and soldered to the tag. Unfortunately, these were preventing the tag from being detected by the readers because they did not conform to the exact antenna specifications. The system could only receive measurements when these antennas were being touched by a human acting as an antenna extension. Another possible explanation was that the wires were composed of a bundle of smaller wire strings, which might have introduced interference in the radio transmission.

After a number of unsuccessful designs, an antenna was carefully constructed following the specifications of the manufacturer. Its wire was single and thick to ensure

[3]SPHINX - a Python documentation generator - `http://sphinx-doc.org/index.html`
[4]BLOCKDIAG - simple diagram images generator - `http://blockdiag.com/en/`

a strong signal would be emitted. Fortunately, this antenna worked perfectly and even increased the transmission range of the tag from eight meters as advertised by the manufacturer to 13 meters as identified during the hardware evaluation. The final antenna design can be seen on Figure 2.3b.

4.3.2 Serial to USB converters

The RFID readers communicate their measurements using a RS-232 serial port. The authors of the SpotON localisation system have identified the limitations of such serial connections [Hightower et al., 2000, p. 6]. This type of cabling is not universal and has limited length. Moreover, RS-232 serial ports are neither present on the Raspberry Pi computers, nor on most modern computers. In order to provide a convenient way of communication between an RFID reader and a Raspberry Pi, serial to USB converters were ordered along with the readers.

A problem was detected during the initial serial communication experiments. When a serial connection is established, a flood of old identification and RSSI data filled the software input buffer of each Raspberry Pi. After in-depth research of the issue, there was a strong indication that this could be caused by the chip of the serial to USB converters. This chip was sending data with high speeds, although the tag transmits its identity every two and a half to three seconds.

This unexpected behaviour needed further investigation. One of the converters was taken apart, as seen on Figure 4.1. On the one hand, the sign on its case was indicating that it is model U-232-P9 manufactured by MCT Corp.. On the other hand, the chip model was PL2303HX detected by the Linux kernel as PL2303 manufactured by Prolific Technology, Inc.. Logically, one might ask what the brand of the converters was. In Linux, there is driver support for U-232-P9 as well. Attempts to use this driver instead of the automatically detected one resulted in a system crash. Further research indicated that this chip (PL2303HX) is an imitation of a genuine Prolific Technology chip [5]. At this point, it was known that these converters were cheaper counterfeits. Nevertheless, they did work, but flooded the input buffers of the computers at the start of every serial connection.

The solution that worked best is to repeatedly drain the operating system input buffer until it holds a few bytes, an indication that the devices are communicating normally. A drawback of this approach is the time lost until the communication stabilises.

[5] Prolific Technology Inc. PL2303 Windows Driver Download - `http://www.prolific.com.tw/US/ShowProduct.aspx?p_id=225&pcid=41`

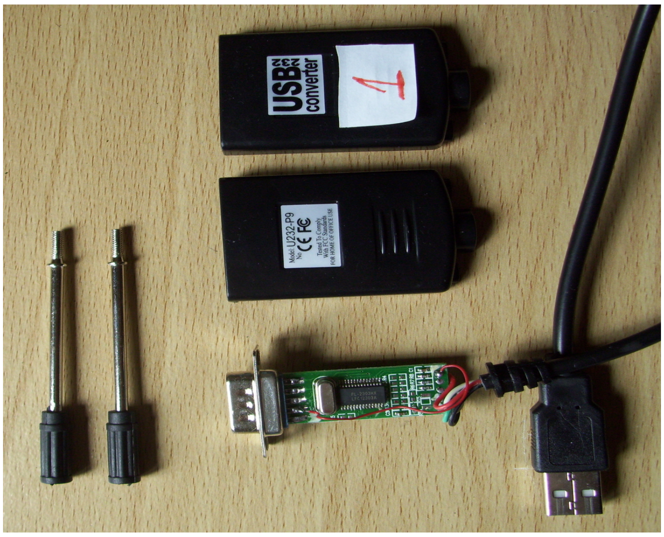


Figure 4.1: The disassembled serial to USB converter.

Up to 58 initial buffer drains were recorded. A third of a second was chosen as a waiting time between buffer drains to ensure the operating system had time to perform this operation. This resulted in a maximum of 17.4 seconds ($58 \times 0.3s$) initial waiting time before the system could start its normal operation. In addition, each of the nodes had to not only wait between zero and a maximum time, but also ensure the other nodes had drained their buffers. This is because readings from all three nodes were needed for estimating the position of the tag. Due to the hardware nature of the problem, time would always be lost until different converters are tested.

4.4 Reader nodes construction

Constructing the reader nodes consisted of three main steps. First, an operating system had to be installed on each of the Raspberry Pi computers. The Raspbian Linux distribution[6] was selected because it is specifically optimised for these single-board computers and has a rich software base compiled for the ARM processor architecture. Second, the RFID readers and Raspberry Pi computers were connected through USB. Third, a software interface between the devices was implemented.

Most of the system was programmed in the PYTHON programming language. All functions concerned with the serial communication of the devices were grouped together in a class called SerialConnection. The PYSERIAL module was extensively used

[6] Raspbian Linux website - `http://www.raspbian.org/`

to implement the required functionality. The class consisted of methods for initialising a serial connection on a given port (`/dev/ttyUSB0`), opening and closing this port for communication, flushing the input buffer, and reading incoming data. The last function had to be implemented to parse the information arriving from the readers. Most serial devices separate their individual readings by a newline character (`\n`), carriage return (`\r`), or a combination of both (`\n\r`). These readers, however, separate measurements by a space character. As a result, PYSERIAL functions for reading data could not be used. The solution implemented reads incoming information character by character until the separator is encountered. All methods provided by the SerialConnection class are illustrated as a sequence diagram on Figure 4.2.

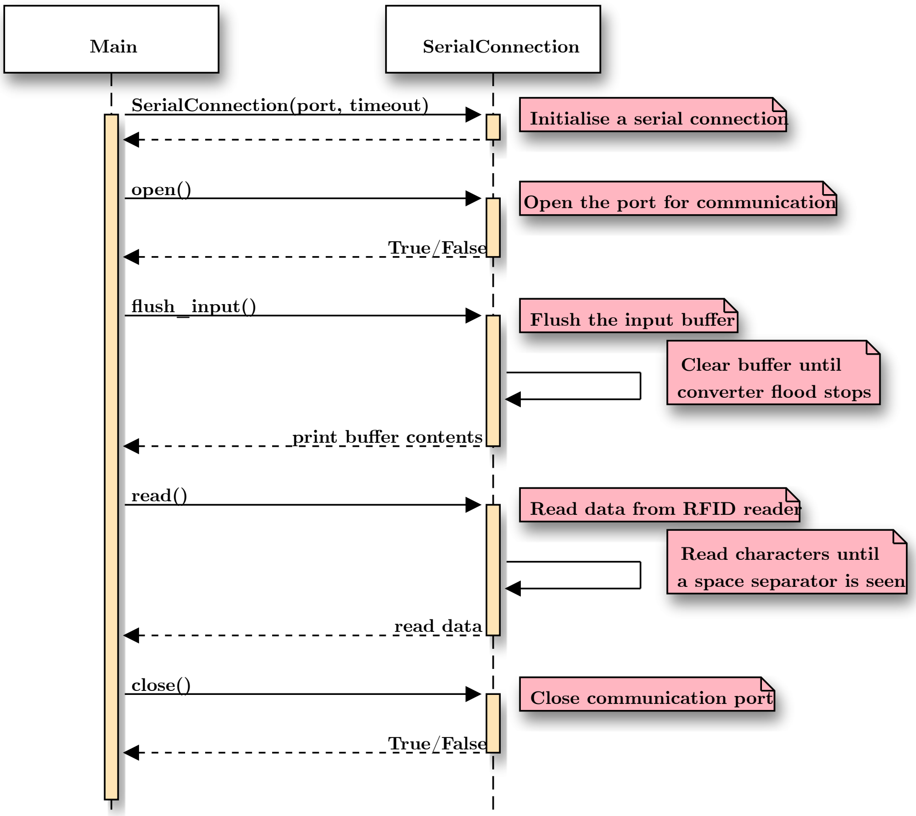


Figure 4.2: A sequence diagram illustrating the methods of the SerialConnection class. The usual sequence in which they are called is shown from top to bottom.

4.5 Data store and management

In this project, there were a number of important data fields that need to be accessed by processes and threads. For example, as discussed in section 4.2.3, the system employed a server-client model where the server consisted of threads that simultaneously receive RFID readings from the other two nodes. These measurements needed to be stored, but also retrieved by the localisation algorithm or by the website visualiser, presented in section 4.8. This was a motivation for employing a data structure that was independent of the software components of the system. Support for simultaneous access of the data was also required.

The SQLITE[7] database management system met the above requirements. It is a file-based system that operates without a server process, which spares system resources. SQLITE serialises transactions so that all changes appear atomic and do not overlap in time. Database transactions might happen in parallel, but are processed in a sequential order by the database management system. Queries to the database were implemented in functions part of the DatabaseHandler class. Instances of this class were created in other classes where access to data was required.

Important data fields were the RFID readings, the positions of the reader nodes, and the estimated location of the RFID tag. These were stored in database tables. Examples of such data are shown in Table 4.1.

Reading	Node №	Tag Id	RSSI
1	0	1Fwt	44
2	2	1Fwt	78
3	1	1Fwt	48
4	0	1Fwt	43

(a) Top four reader measurements.

Node	Node №	x	y	z	radius
Reader	0	0.0	3.0	0.0	4.0
Reader	1	3.0	0.0	0.0	0.786
Reader	2	0.0	0.0	0.0	5.0
Tag	3	5.563	3.0	0.0	0.0

(b) Object positions in two dimensions.

Table 4.1: Sample data from the database of the system.

[7] SQLITE website - `http://www.sqlite.org/`

4.6 Network communication

The overall software design of the system was introduced in section 3.3. It defined the server-client model used to communicate RFID readings over a computer network. This section is concerned with detailing how this network communication was implemented.

Two of the Raspberry Pi computers acted as nodes that gathered information from their RFID readers to immediately send it over the network to the third node. This node was designated as the server of the system collecting readings and using this information to compute the tag's position. The network communication was implemented using socket programming. Python's SOCKET module offered all required functions.

At the client side, the NetworkClient class was responsible for initialising a client streaming socket. A client uses a client socket to connect to a server socket. A streaming socket was chosen because nodes communicate a continuous stream of data to the server node. After this socket is created, a connection request is sent to a server socket specifying the network address and port of the server computer.

At the server side, the NetworkServer class creates a server streaming socket. Next, this socket is bound to the network address of the machine on a specified port. Then, the server socket starts listening for incoming connections. The server checks for connection requests using a non-blocking approach. This means that the programme tests if a request is available without indefinitely waiting for one to arrive, thus being able to terminate if requested. Once a client tries to connect, the NetworkServer accepts it by creating a client socket at the server side. At this point the only thing left to serve the incoming request is for the server to spawn a thread that handles the connection from this moment on. The motivation for using a threaded network server is discussed in section 4.2.3.

Each network server thread receives incoming data from one client node. Listening on the socket for new information is again implemented using a non-blocking socket. If the receive buffer is empty, the receiver thread will not block indefinitely until data arrives. Rather, this buffer is checked constantly, but the thread could still be controlled by its parent process. If new data is available, it is read and inserted into the database using an instance of the DatabaseHandler class.

In case the system is interrupted by the user, the network server sends a stop signal to each of its threads and waits for them to terminate individually before terminating itself. This is the behaviour of the join(timeout) function of the Python's threading module. All the functionality described above is illustrated as a sequence diagram on Figure 4.3.

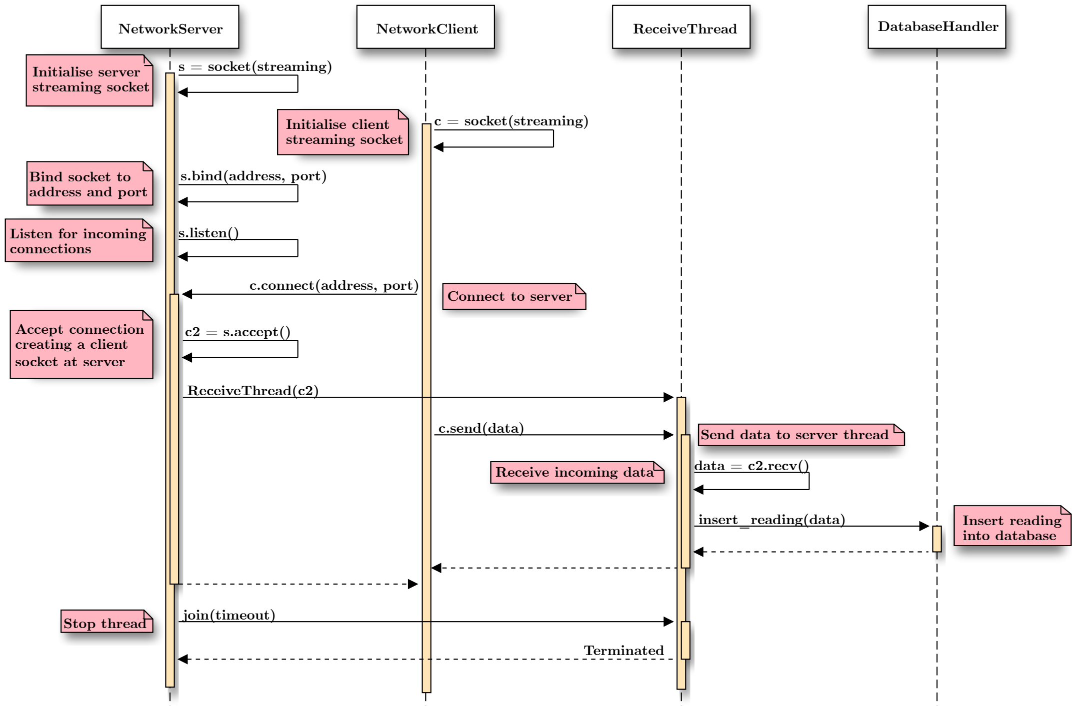


Figure 4.3: A sequence diagram illustrating the methods of the NetworkServer, NetworkClient, and ReceiveThread classes. The usual sequence in which they are called is shown from top to bottom.

4.7 Location estimation

This section presents the implementation details of converting RSSI to distance and computing the unknown position of the RFID tag using the trilateration technique.

4.7.1 RSSI to distance conversion

The implementation of converting RSSI to distance relied on the methodology described in section 3.4.2. More specifically, the programme uses an instance of the DatabaseHandler class to get the three most recent RFID readings from the database. For each of these, an RSSI value is converted to distance in meters based on the translation table of the reader at which the measurement was recorded. This is needed because of the slight hardware differences between the RFID readers. Next, an integer factor is added to an RSSI value to account for the battery power drop as it is being used. Then, the position of an RSSI measurement in the translation table is determined. Finally, RSSI is linearly converted to distance for the specific range.

4.7.2 Trilateration

The trilateration position estimation technique was implemented relying on the general three-dimensional solution of the method presented in section 3.5. The algorithm accepts as inputs the positions of the three reader nodes and their RSSI measurements. The positions are fetched from the database and the RSSI values are converted to distance using the aforementioned steps. All equations of the trilateration method were directly implemented with the aid of the MATH (mathematical functions) and NUMPY (scientific computing) modules. It was noted that a number of recurring intermediate computations could be computed once and inserted in the equations as variables. Such computations included $\vec{p}_2 - \vec{p}_1$, $\vec{p}_3 - \vec{p}_1$, and $\hat{e}_x i$. The algorithm outputs the estimated location of the unknown object. This result is inserted into the database so that it could be visualised on the web interface. Figure 4.4 is a sequence diagram showing the functions of the localisation algorithm and how it interacts with other software components.

4.8 Web Interface

The web interface is a website that runs on the server node. It main purpose is to provide a convenient interface to interact with the system. The web page is automatically refreshed every three seconds, which is roughly the interval at which the RFID tag transmits its identity. The website provides the following information:

- the positions of the readers and tag,
- a form to change the positions of the reader nodes,
- the error in meters of the estimated position to the true one,
- a graph with reader positions and their distance from the tag,
- a table with the three most recent RFID readings,
- a table providing status information about the Raspberry Pis.

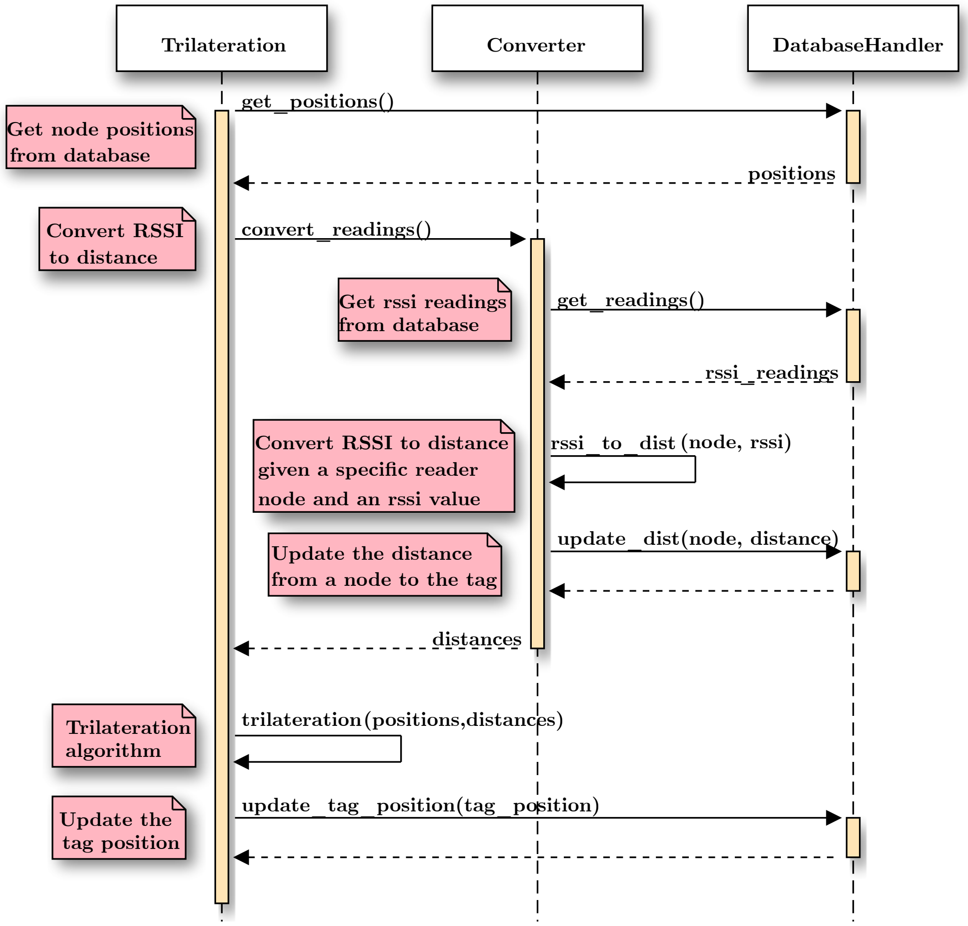


Figure 4.4: A sequence diagram illustrating the methods of the Trilateration class. The usual sequence in which they are called is shown from top to bottom.

The website is implemented in the PHP programming language. All information is provided from the SQLITE database. The website runs on an APACHE HTTP web server installed on the server node. The website was password protected to allow only people involved in this project to access it through the Internet. Figure 4.5 is screen capture of the web interface.

4.9 Summary

This chapter detailed the software practices that helped develop this system. It also described hardware issues that were faced throughout the project. Next, implementation details of all parts of the system were presented. Figure 4.6 is a diagram of all system components. The next chapter discusses the evaluation of the system's hardware component and the overall system's performance in terms of localisation accuracy.

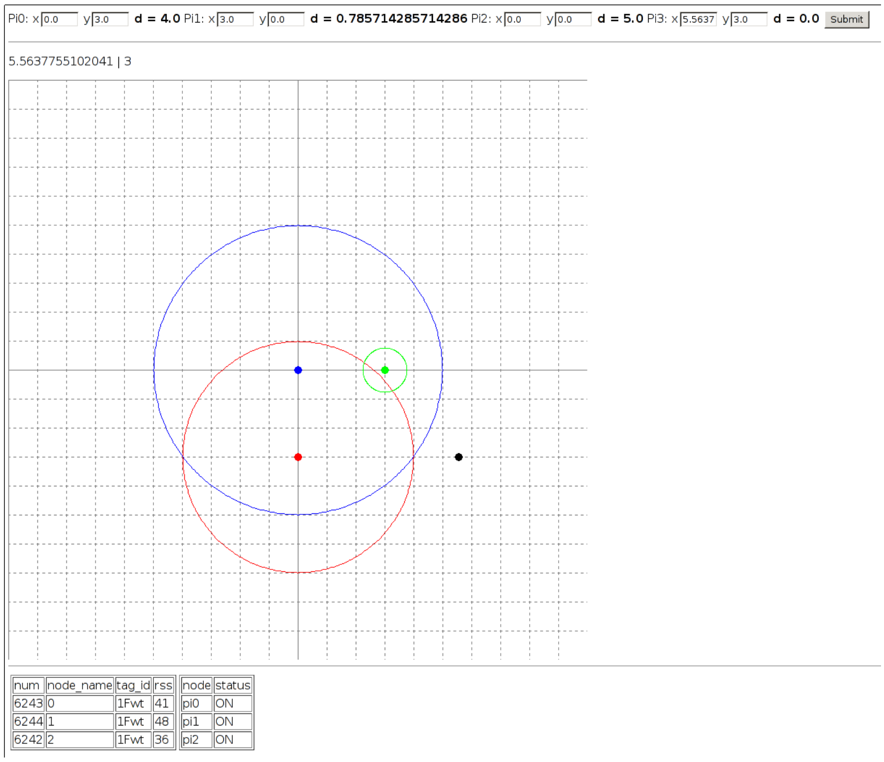


Figure 4.5: The web interface running on the server node.

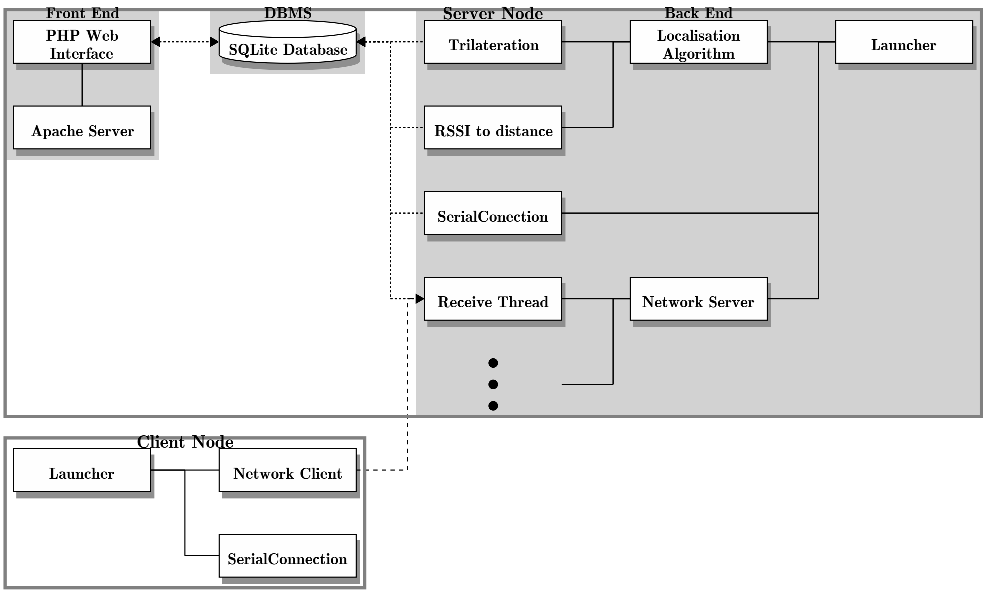


Figure 4.6: A diagram illustrating all software components of the system. Both server and client nodes are shown.

Chapter 5

Evaluation

This chapter describes the experiments conducted to evaluate the RFID hardware and localisation accuracy of the system. Each experiment description consists of justification, setup, results, and analysis. Section 5.1 presents tests of the RFID hardware that provide an understanding of the mapping between RSSI and distance. Then, section 5.2 details the experiments that check whether the problem, described in section 3.2, is solved by the system. It also provides results in terms of localisation accuracy and how these compare to results of the related work.

5.1 Hardware Evaluation

This section presents the experiments conducted in attempt to correlate received signal strength indicator (RSSI) and distance. This was accomplished by positioning the RFID tag at increasing distances and recording RSSI measurements from the RFID readers. Distance measurements were marked by a tape measure and RSSI values were recorded using the SerialConnection class, described in section 4.4. All experiments had taken place in two indoor environments, an apartment and a university computer laboratory. These places differ in the amount of open space and the number and placement of objects.

In indoor environments, the attenuation of a radio signal is strongly influenced by the physical effects of reflection, refraction, multipath and shadow fading. Following from that, in order to construct a more complete notion of the relationship between RSSI and distance, a number of experiments were carried out. These include how RSSI changed as distance increased at different orientations of the tag to the readers and elevations of the tag from the floor. In addition, the wall penetration and range capabilities of the hardware were evaluated. The following subsections describe these experiments.

5.1.1 Range

The aim of this experiment was to measure the range capabilities of the RFID equipment in an indoor environment. In addition, RSSI values were recorded as distance between the readers and tag increased. The devices were keeping a line of sight between each other. Then, the test was repeated by obscuring the tag with an object, in this case a large whiteboard, to determine if it was affecting the communication range of the devices and the RSSI values reported by the readers. The experiments were carried out for each of the readers in a computer laboratory that had a 13-meter-long stretch of open space. The results of these experiments are illustrated on Figure 5.1.

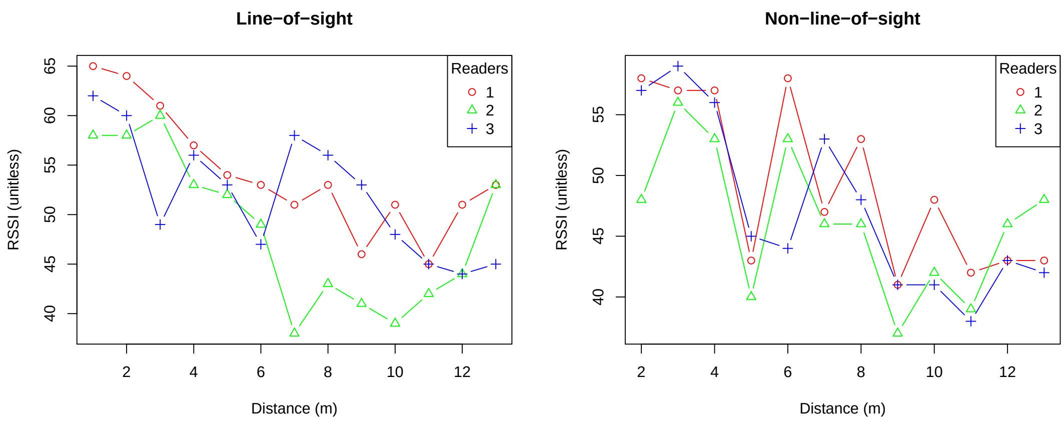


Figure 5.1: Two plots of RSSI measurements at increasing distances. On the x-axis, the distances from one to 13 meters are shown. The y-axis is the RSSI range. Notice how the RSSI ranges differ between the plots. The left graph shows RSSI value changes with a line-of-sight signal propagation between devices. The right graph illustrates the same experiment but with a non-line-of-sight signal propagation, that is the tag was occluded by an object.

Three observations could be made by looking at the above results. First, in both experiments all three readers were able to detect the tag when located 13 meters away. The manufacturer reported a transmission range of 8 meters. Consequently, the antenna, designed in this project, had definitely increased the transmission range of the RFID tag.

Second, these experiments confirm that the readers measure different RSSI at equal distances. As distance increases, the average difference of RSSI values is 4.93. RSSI ranges on the y-axis are between 20 and 30. Following from that, if the measurements reported by the readers differ in an average of around five units given the above ranges, then these readers are not calibrated to each other. This was observed in subsequent experiments as well. It was the motivation for constructing separate translation tables when mapping RSSI to distance.

Third, it is known that radio signals suffer from the effects of various physical phenomena when propagating in indoor environments. Figure 5.1 shows that RSSI measurements do not decrease steadily and often fluctuate as the distance increases. This could be contributed to the characteristics of indoor spaces, where the shape of the space and the objects in it could cause reflection, refraction, and absorption of radio signals. Introducing an object that occluded the RFID tag had definitely decreased the RSSI values reported by the readers. It also caused the measurements to vary in unexpected ways.

5.1.2 Orientation of tag to reader

This experiment investigated whether placing the tag at different positions around the readers, would affect RSSI measurements as distance increases. Outcomes of this test contributed to the understanding of the factors that determine how RSSI changes in different cases. The experiment started by placing the tag so that it touched antennas with a reader. Then, the tag was moved away at half a meter marks until the distance was three meters. Next, the tag was rotated counterclockwise around a reader at 45-degree steps. The experiment was done for each reader. In addition, the tests were repeated by placing a filled suitcase in front of the tag to simulate the tag being carried by a human or attached to an object. The concept of the experiments is illustrated on Figure 5.2.

The results from the experiments are illustrated on Figures 5.3 and 5.4. Figure 5.3 shows the changes in RSSI as distance increases for different orientations of the tag with respect to the first reader in a line-of-sight radio propagation. Figure 5.4 is a plot of a similar experiment but with the tag being obscured by an object. Notice that the minimum distance between devices is half a meter, which is needed to place the obstacle. Plots of the results using the other two readers are shown in Appendix A.3.

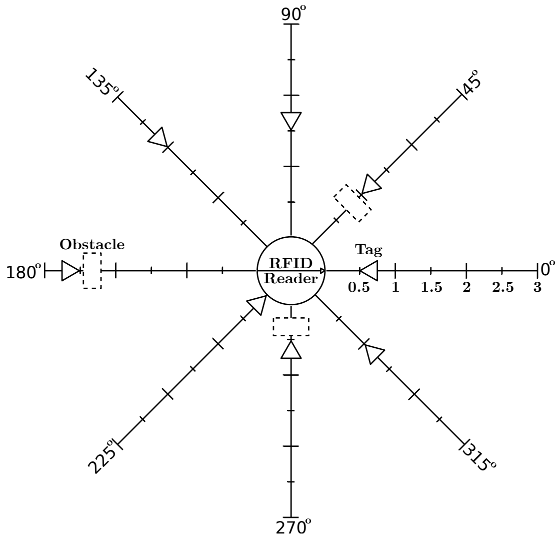


Figure 5.2: Examples positions of the tag and obstacle around one of the readers.

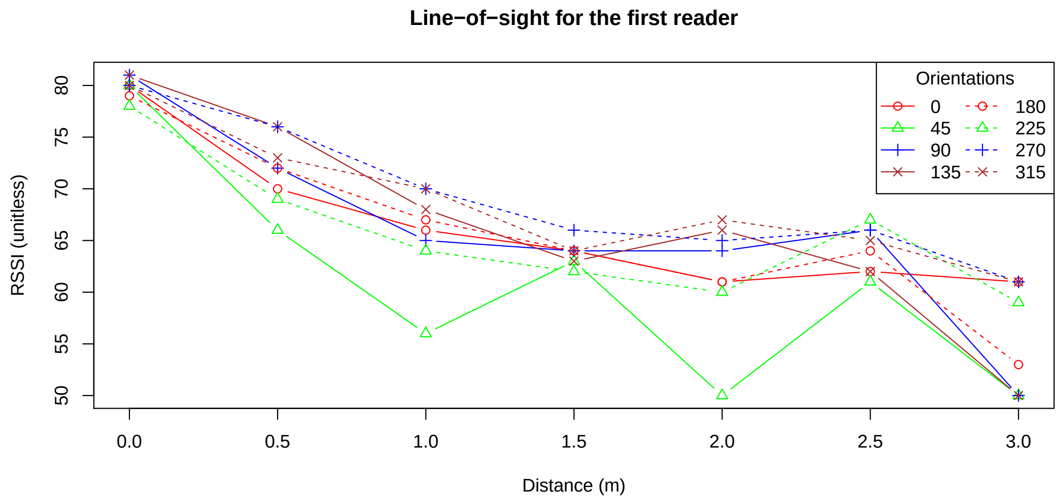


Figure 5.3: Plot of RSSI measurements of different orientations of the tag with respect to the first reader. There was a line of sight between the RFID devices.

The first thing to notice is that when the tag touches antennas with any of the readers, they reported RSSI values of around 80 units. This was unfortunate because the manufacturer had advertised an RSSI resolution from zero to 255. Clearly, the maximum values were much lower, which certainly had an impact on the localisation accuracy of the system. Nevertheless, the aim of the experiment was different. It tried to test whether the position of the tag around any of the readers had any significance on the measured received signal strength indicator. Figure 5.3 is hard to follow but it is evident that regardless of the orientation of the tag, the first reader had measured similar RSSI values as distance increased. There are some points that lie outside of the main areas of values, which might be attributed to physical conditions in the indoor environment or in voltage variations in the battery of the tag. Another observation is that along opposite orientations of the tag, such as 90 and 270 degrees, measurements follow similar paths and have small differences in their RSSI values. This had no real significance for this project but is surely a pattern that could be spotted.

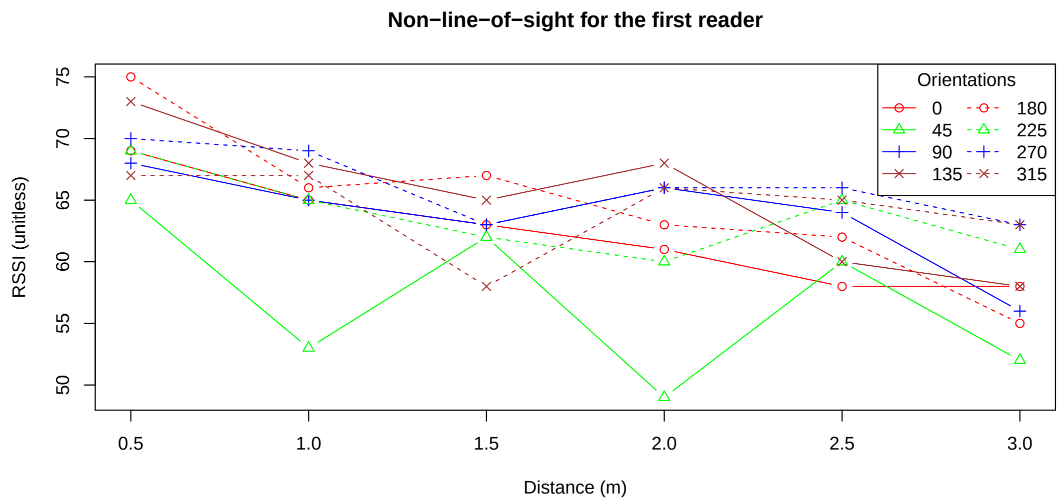


Figure 5.4: Plot of RSSI measurements of different orientations of the tag with respect to the first reader. The tag was obscured by an object.

As explained above, the experiment was repeated by positioning an object in front of the tag to determine the impact of such occlusion on the RSSI measurements. Similar to the range experiments in section 5.1.1, lower RSSI values were recorded at close distances but the overall range of values on Figure 5.4 is very similar to the one in the previous experiment. In contrast, RSSI measurements are more spread out at most distances, which is an indication that obstructing bodies do have an effect on the amount at which RSSI varies.

5.1.3 Grid of positions

The purpose of this experiment was to obtain further information on the relationship between RSSI and distance. More specifically, it was checked how RSSI would change when the tag was rotated around itself while being moved on a four by four grid of positions. Each reader was placed at the upper left corner (`position(0,0)`) and the distance between adjacent points on the grid was one meter. The tag was rotated at 90 degree intervals starting by facing the right side of the grid. The results from this experiment using the first reader are shown on Figure 5.5. Results obtained from the other readers are plotted in Appendix A.4.

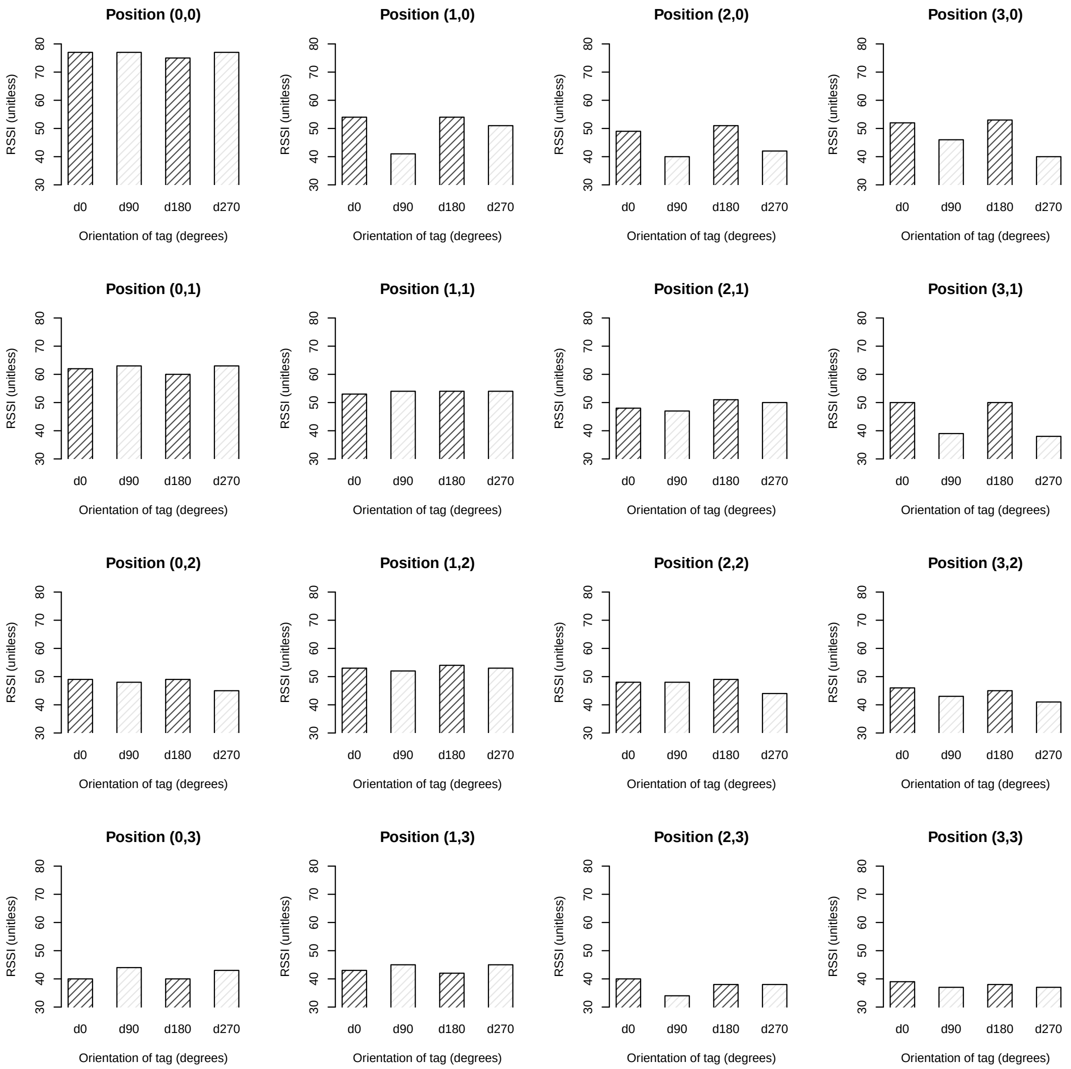


Figure 5.5: Sixteen plots organised into a four by four grid. Each plot represents the RSSI measurements of the **first** reader when the tag was placed at different positions on the x and y axes of the grid. The positions of the tag were measured in meters. The four bars in each plot show the RSSI readings when the tag was facing right (0°), up (90°), left (180°), and down (270°).

Using the knowledge obtained from previous experiments, the range of RSSI values on the y-axis was limited between 30 and 80 RSSI. In this way, the differences in the measurements from various tag orientations are easier to spot. It also tells the reader that there were around 50 units of resolution provided by the RSSI module of the RFID readers. The manufacturer had claimed 256 steps of RSSI sensitivity, which was never achieved in this project. The leftmost column on Figure 5.5 illustrates the change in RSSI when the tag was placed in front of the reader. It can be seen that regardless of the rotation of the tag around its base, the RSSI values gradually drop. In contrast, the top row contains bar plots showing the tag being moved to the right of the reader. When it was facing up (90°) and down (270°), the tag was detected with up to 10 units less than in the other two orientations. The reason for this might have been that the radio waves emitted by the tag had to first bounce of walls or objects before being received by the reader. The conclusion that can be made is that although the devices were situated only one to three meters away, certain orientations of the tag might be perceived by the reader as the tag being further that it really was. Consequently, the orientation of the tag had an impact in some cases, which ultimately influences the localisation accuracy of the system.

5.1.4 Elevation from floor

The aim of this experiment was to investigate how RSSI would change if the tag was positioned at different heights relative to the room floor. The readers were placed at 6, 60, and 130 centimetres from the ground. Measurements were taken at these heights as the distance between the devices increased from one to four meters in half a meter steps. The setup of this experiment is illustrated on Figure 5.6.

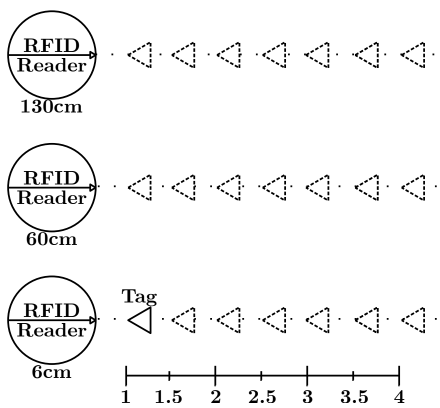


Figure 5.6: Positions of the tag when being elevated from the floor as distance increased. The readers were located at 6, 60, and 130 centimetres from the floor.

Experimental results can be seen on Figure 5.7. Each plot represents the RSSI measurements of the three readers with the tag being elevated at a specific height from the floor. On the first plot, the tag is very close to the ground (6cm). Logically, the third reader, which is also at the same height, generally measured higher RSSI values as the tag was being moved away. The other readers scored similar results with certain fluctuations in the measurements, which might be caused by the batteries, motion in the indoor environment, or simply by the quality of the devices. Similar trends are observed in the other two plots. It is interesting to note that as the tag is elevated higher in a room, reader measurements become more stable and large variations occur less often. An explanation might be that the upper parts of indoor spaces are usually free from obstacles, which provides a more reliable propagation medium.

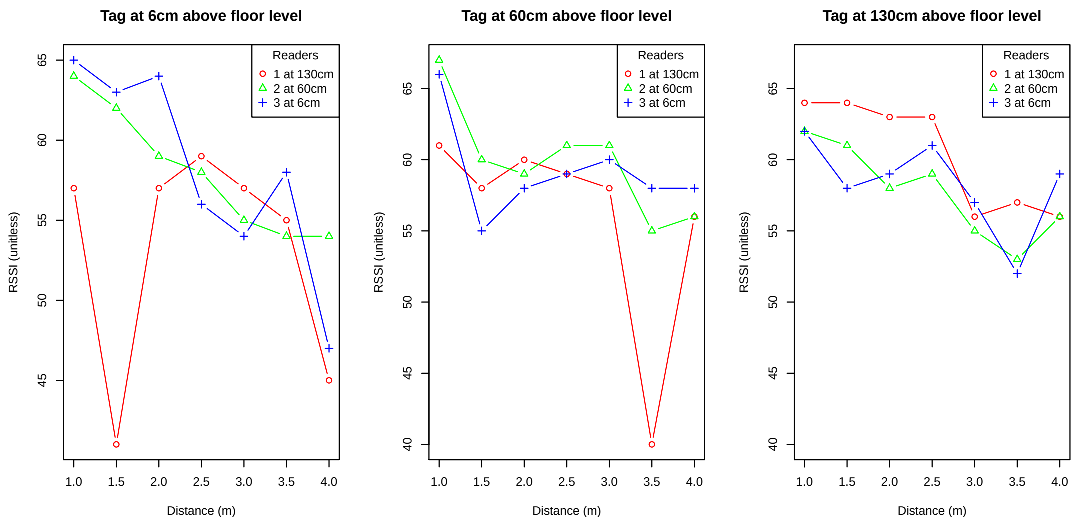


Figure 5.7: Three plots of RSSI measurements at increasing distances with the readers at different elevation from the floor. The first graph shows readings when the tag is placed 6cm above the floor level. The second and third graph show the same experiment but the tag is at 60cm and 130cm from the ground.

5.1.5 Wall penetration

This last experiment of the RFID hardware involved testing the wall penetration capabilities of the radio signal emitted from the tag. The setup for this test is shown on Figure 5.8. One of the readers was placed in a room. The tag was first positioned in the adjacent room and reported 41 RSSI. Then, it was moved in the next room, where it was detected with 51 and 40 RSSI in the nearest and furthest points of this space.

Finally, the transmitter was carried outside the building. The reader was being able to detect it through two thinner inside and one thicker outside walls. This experiment was conducted to get an idea of the range capabilities of these RFID devices. This simple test showed that if the readers are positioned in different rooms they might be able to detect transmitters.

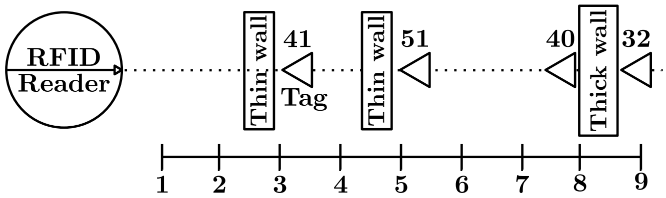


Figure 5.8: Experimental setup of the wall penetration test. The tag was positioned at 3, 4.8, 7.5, and 8.6 metres from one of the readers. The numbers above the tag are the measured RSSI.

5.2 System evaluation

This section presents the experiments conducted to test the localisation system developed in this project. They tried to check the hypothesis, which is to evaluate the possibility of building a system capable of estimating the position of an unknown object with a certain accuracy using affordable RFID equipment and Raspberry Pi computers. The results of these experiments consist of the errors in meters between the computed and real positions of the tag. At the end of this chapter, the average localisation error is compared to the results of the related work. The following subsections describe the experiments carried out to evaluate this system.

5.2.1 Grid of positions with line-of-sight propagation

The aim of this experiment was to investigate the performance of the system when all readers had a line of sight to the tag and the devices were lying on the same plane, thus working in two dimensions. This was the case where the system was expected to output its best results because the setup had omitted obstacles sitting in between the devices. A grid of 16 positions was marked on the floor of an indoor space. The distance between any of the adjacent points was one meter, hence forming a three by three meter square. The RFID readers were positioned at the upper left, upper right, and bottom left corners of the grid. These were positions (0,0), (3,0), and (0,3) if all points of the grid had coordinates starting from the upper left corner with x positive to the right and y positive down. The tag was consequently placed in all positions. In addition, it was rotated around itself at 90 degree steps. This was done check if the orientation of the tag had an impact on the accuracy of the system. For every orientation, the localisation error in x and y was recorded.

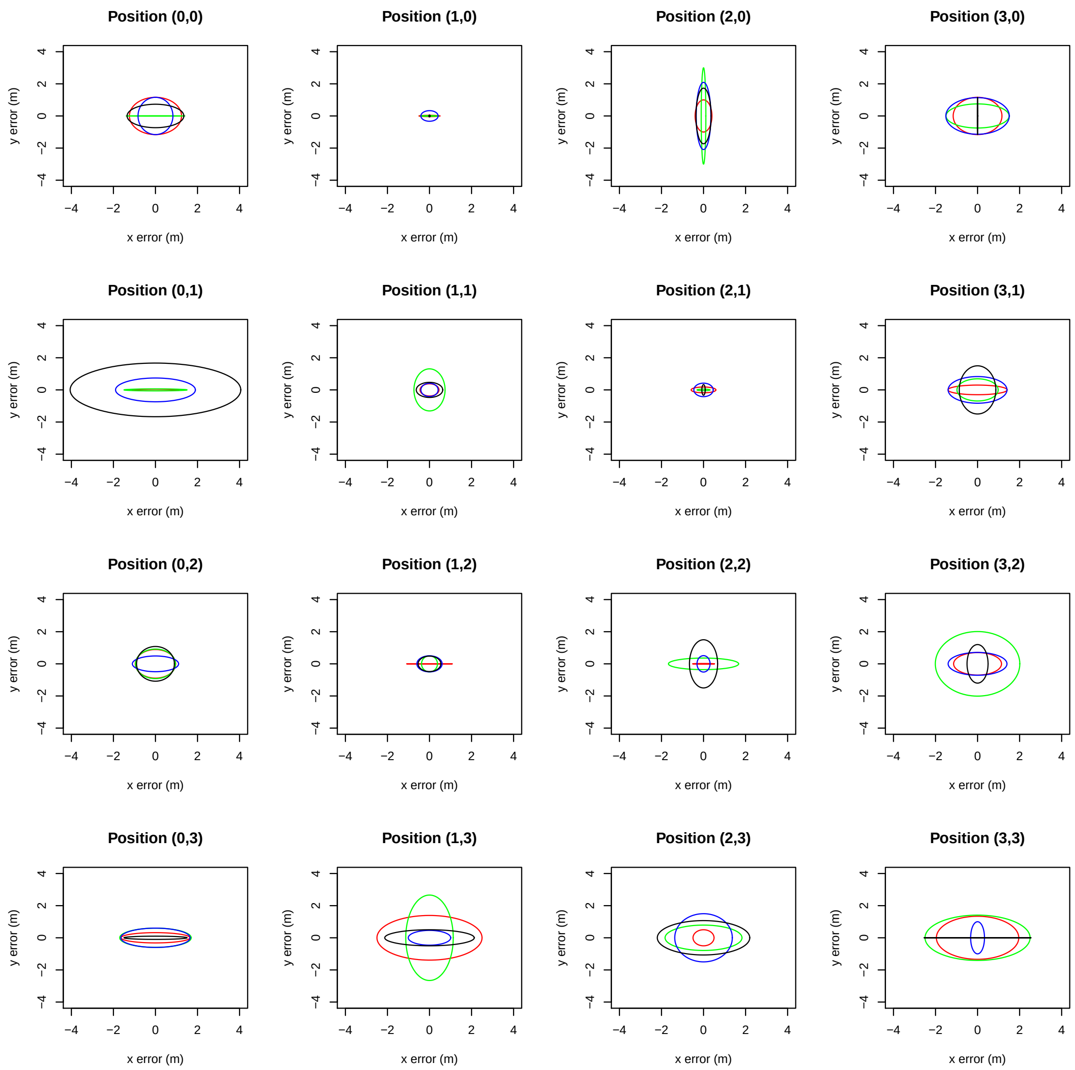


Figure 5.9: Sixteen plots are organised into a four by four grid. The readers are placed at positions (0,0), (3,0), and (0,3). Each plot represents the error in meters between measured and estimated location when the tag is placed at different positions on the grid. Each plot consists of four ellipses that illustrate the x and y error when the tag is facing right (0°), up(90°), left (180°), and down (270°). The colours of the ellipses are red, green, blue, and black, respectively.

Figure 5.9 illustrates the results of this experiment. Every position of the grid is represented as a plot of the error in x and y for the four possible orientations. The centres of the plots mark a zero localisation error. Errors are drawn as ellipses that are stretched in the direction of the error. Different colours label the orientation of the tag when the measurements were recorded. For instance, when the tag was at position (1,3) and was looking to the right (0°), the error, coloured in red, was around two and half meters in x and around a meter in y.

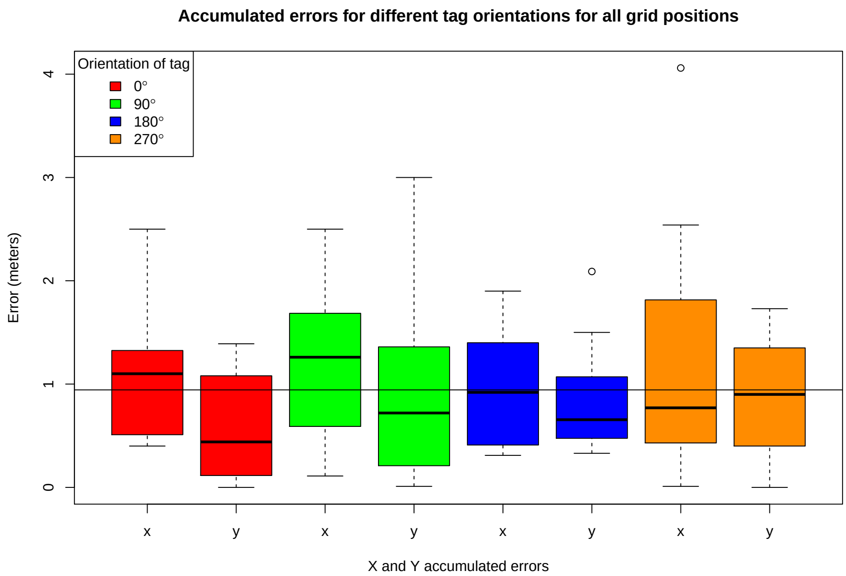


Figure 5.10: A box plot showing errors between measured and estimated locations. The boxes are organised in four groups. Each group consists of errors in x and y for a particular orientation of the tag. The horizontal line across the plot is the mean (0.943) of all errors regardless of the tag orientation.

In general, by looking at a plot for a certain position, one could tell the overall amount of error. Using this approach, it is evident that the system was localising the tag with a sub meter accuracy when the tag was placed equidistantly from the three readers. In positions where the tag was close to one reader but far from the other receivers, the system seemed to perform worse. The reason for this is in the way RSSI is converted to distance, rather the performance of the localisation algorithm. Section 3.4.2, discussed a drawback of the translation tables. As the distance grows, RSSI values decrease at a slower rate. For example, for distances between zero and one meters, the second reader would measure 14 different RSSI values. In contrast, if the tag is five to six meters away, the RSSI changes by one unit (data from Appendix A.2). Consequently, as distance increases, the accuracy of the RSSI to distance conversion decreases.

Figure 5.10 is an alternative visualisation of the experimental results. It is a plot of the errors in x and y grouped by orientation disregarding the position of the tag on the grid. The plot shows that different orientations produce various amounts of error. For instance, when the tag was facing to the left (180°) in all positions, the mean error was under one meter in both x and y. This figure also contains a horizontal line marking the average error (0.943m) of all measurements of this experiment.

5.2.2 Three-dimensional grid of positions

The purpose of this experiment was to test the localisation performance of the system in a three-dimensional setting. In a university computer laboratory, a space, that was four meters wide, six meters long, and two meters high, was selected. The walls of the room were lined with desks, chairs, and computers. One of the corners, where desks met, was marking the start of the coordinate system used by the system. The reader nodes were positioned on the sides of this space. They were also elevated at three different heights from the room floor. The first reader was set up at 2.4 meters to the right and 1.7 meters above ground from the point marking the origin. The second reader had the following coordinates (3.4m , 5.7m , 0.75m), which was the furthest from the origin and its receiver was placed on a table. The last reader was 3.2 meters down from the start of the coordinate system and eight centimetres above the floor.

After the readers were set up, the tag was repeatedly moved in a three-dimensional grid of positions. The x and y coordinates of the grid were starting half a meter from the origin in order to avoid obstacles that were in the way of the stand, used to lift the tag off the floor. The three heights at which the readers were positioned determined the z component of the grid. This experimental setup is visualised on Figure 5.11. It shows the positions of the readers and tag.

In order to produce a three-dimensional position for the tag, the system had to be modified slightly. Section 3.5 described how trilateration worked. In three dimensions, the number of solutions is determined by the z component. In rare cases when the sphere surfaces touch in one point, there is a single solution to the positioning problem. In most cases, however, the three spheres intersect, which results in two points of intersection, as shown on Figure 5.12. In this experiment, the middle point of the line segment formed by the two intersection points was used as the estimated position of the tag. Then, it was subtracted from the real position of the tag, manually inputted into the system. This gave the localisation error in three dimensions:

$$error = (|x_{real} - \frac{x_1 + x_2}{2}|,\ |y_{real} - \frac{y_1 + y_2}{2}|,\ |z_{real} - \frac{z_1 + z_2}{2}|)$$

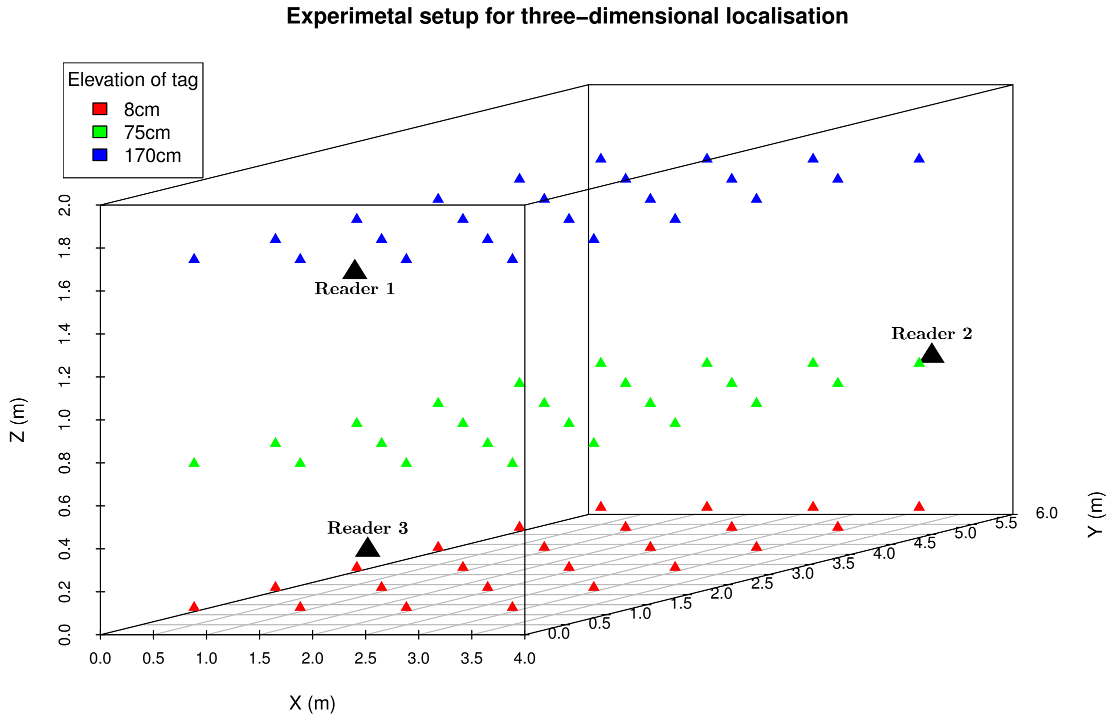


Figure 5.11: Experimental setup for testing the localisation system in three dimensions. The positions of the readers are shown as black triangles. Their coordinates are (2.4, 0.0, 1.7), (3.4, 5.7, 0.75), and (0.0, 3.2, 0.08). Three tag elevations are marked by colours to visualise the grid of positions better.

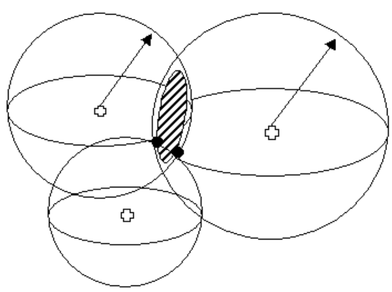


Figure 5.12: The intersection of two spheres is a circle. A third sphere narrows down the intersection to two points[1].

The results from this experiment are plotted on Figures 5.13 and 5.14. Figure 5.13 is a plot of the errors in x, y, and z for every elevation of the tag from the room floor. The data was visualised in this way in order to investigate which horizontal plane in a room provides the best propagation medium. The first three boxes in the plot illustrate the three-dimensional localisation error for the positions of the tag when it was eight centimetres from the ground. It can be seen that the spread of the errors in both x and y was significant, although its mean was close to one meter. The reason for this might be that there are a lot of obstacles in the lower parts of an indoor space, such as chairs and tables. These objects decrease the signal strength of the radio signal emitted from the tag. Another factor is the vertical distance between a reader and transmitter. Looking at diagram 5.6 used in a previous experiment, it is evident that radio signals have to travel greater distances to reach receivers positioned at a different height.

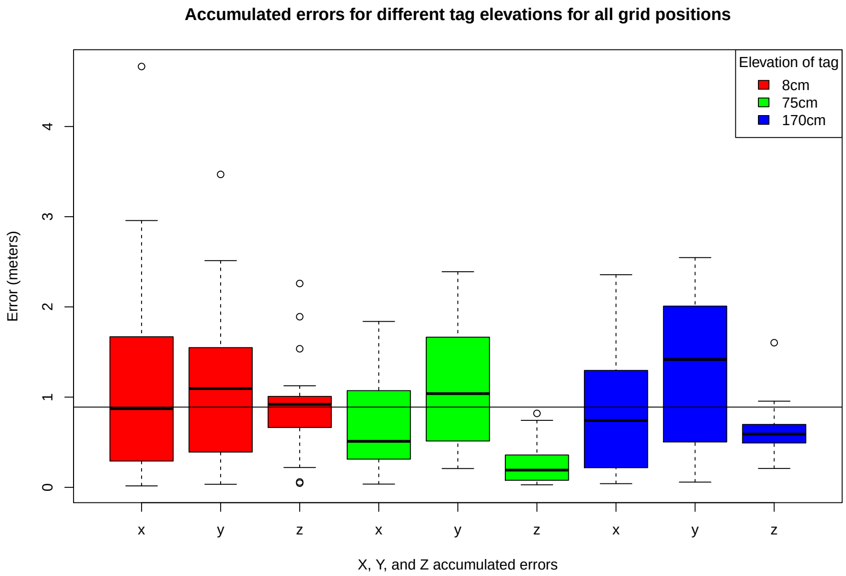


Figure 5.13: A box plot showing errors between measured and estimated locations. The boxes are organised in three groups. Each group consists of errors in the x, y, and z for a particular elevation of the tag. The horizontal line across the plot is the mean (0.8895182) of all errors regardless of the tag elevation.

The same reasoning can be applied to the case when the tag was high above ground (170cm). The last three columns of figure 5.13 show the distribution of the localisation error for this elevation. Again, the errors in x and y had a substantial spread. The error

[1]Global Positioning System - `http://pegasus.cc.ucf.edu/~jweisham/pcb5937/GPS/GPS.html`.

in z appeared to be low regardless of the elevation. Although, the upper parts of an indoor space are usually free from obstacles, this experiment showed that the system produced a considerable localisation error. By placing the tag at desk level (75cm), the system demonstrated the least deviation from the real positions of the tag. A possible explanation is that the tag was in the middle of the grid of positions, which enabled it to be read from below, middle, and above. These RSSI values provided a better estimation of the distance between the devices.

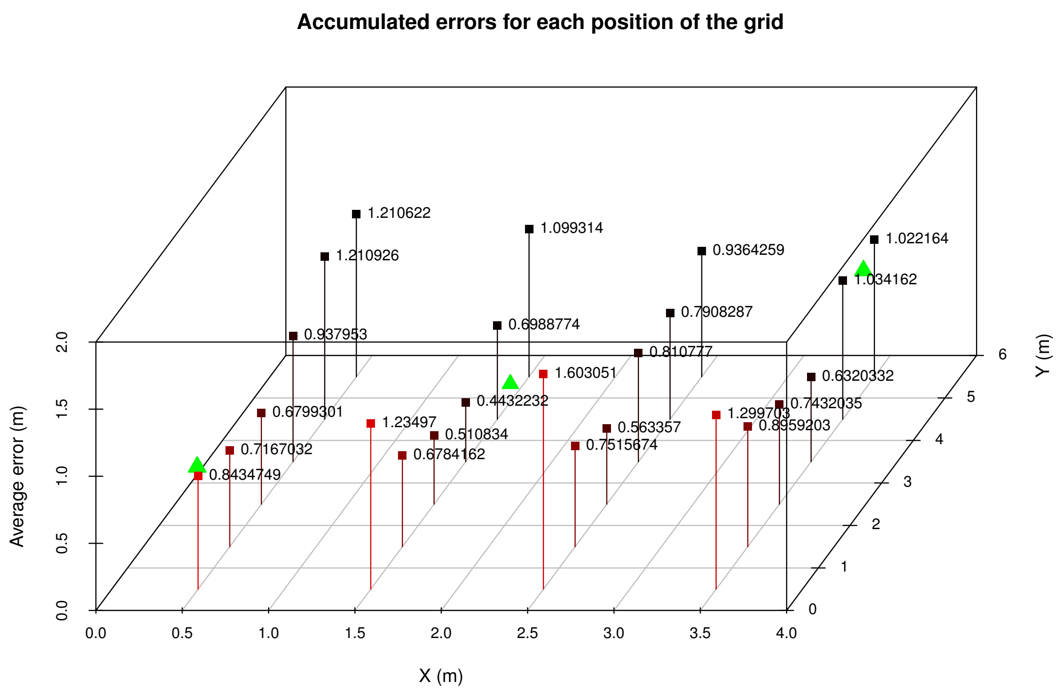


Figure 5.14: A three-dimensional scatter plot of the average error for every position in the grid regardless of the elevation of the tag. The x and y axes are the coordinates of the space and the z-axis is the average error in meters. The numbers to the right of every position are the mean errors. Three green triangles mark the positions of the readers, which are roughly on the front, back, and left faces of the rectangular parallelepiped.

Figure 5.14 is a three-dimensional scatter plot of the mean error for every position in the grid regardless of the elevation of the tag. This data visualisation shows the places where the system estimated the location of the tag with a certain accuracy. Recall that the walls of this indoor environment were lined with objects and that the readers were set up at three of the sides of the test space. These obstacles were a contributing factor that increased the localisation error at the boundaries of the grid. Similar to the results of the previous experiment, lower errors were observed in the central parts of the grid, where tag was at similar distances from the receivers. When the transmitter was close to one but far from the other readers, the errors spiked due to inaccuracies in the conversion of RSSI to distance as the gap between devices increased.

5.3 Discussion

Pi0 x:2.4, y:0.0, z:1.7 Pi1 x:3.4, y:5.7, z:0.75 Pi2 x:0.0, y:3.2, z:0.08

Rss battery factor adjusted during experiment to correct when a reader was misbehaving. Automate as future work

better place readers in middle in 3d, better as distance in 3d is smaller than if at 0.8 to 1.7m.

also readers better be in middle of room on y to have spread of propagation

readers close to walls where are desks furniture...

tag if too low in 3d, many objects (tables, chairs).

if at desk level good,but there might be people.

if high good, not many objects

also readers better be in middle of room on y to have spread of propagation

Future work: average the last 10 readings from readers to gain a better position

Not only overall RSSI resolution of the devices is around 50 units, but also the RSSI does not change evenly throughout the localisation range of the receivers.

5.4 Summary

Chapter 6

Conclusion

6.1 Contribution

6.2 Future Work

6.3 Summary

Appendix A

Supplementary Information

A.1 Hardware setup using Wi-Fi connectivity

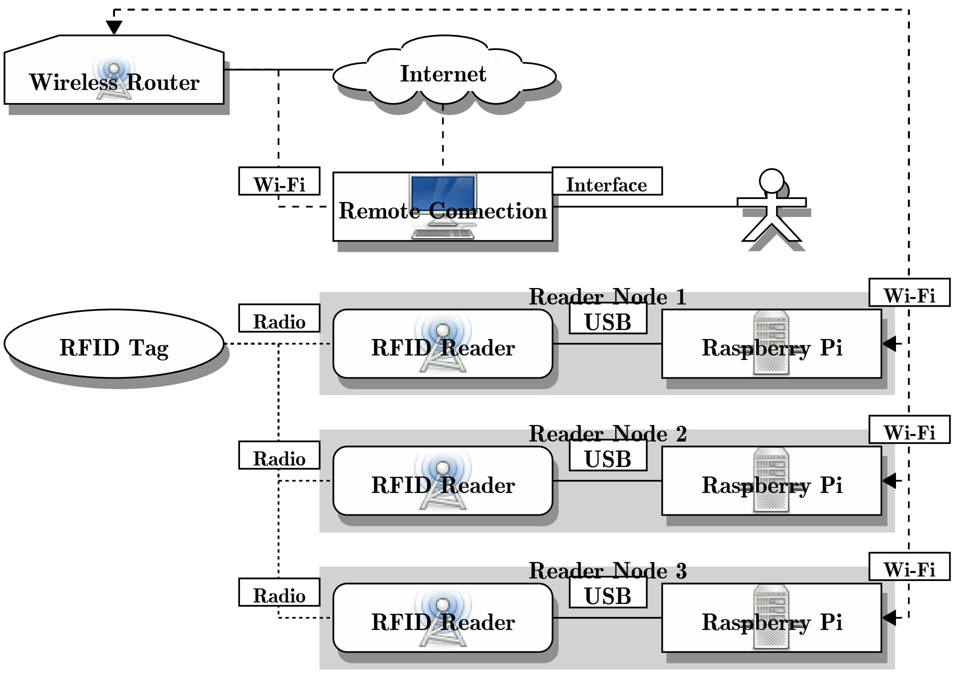


Figure A.1: Hardware setup with wireless connectivity between reader nodes.

A.2 Translation tables from RSSI to distance

Distance	0	1	1	2	2	3	3	4	4	5	5	6	6	7
RSSI	80	65	64	62	61	57	56	53	52	48	47	46	45	44

Table A.1: RSSI value ranges used to estimate distance when using the first reader.

Distance	0	1	1	2	2	3	3	4	4	5	5	6	6	7
RSSI	77	63	62	58	57	55	54	53	52	49	48	47	46	44

Table A.2: RSSI value ranges used to estimate distance when using the second reader.

Distance	0	1	1	2	2	3	3	4	4	5	5	6	6	7
RSSI	78	64	63	60	59	56	55	54	53	49	48	45	44	43

Table A.3: RSSI value ranges used to estimate distance when using the third reader.

A.3 Orientation of tag to reader

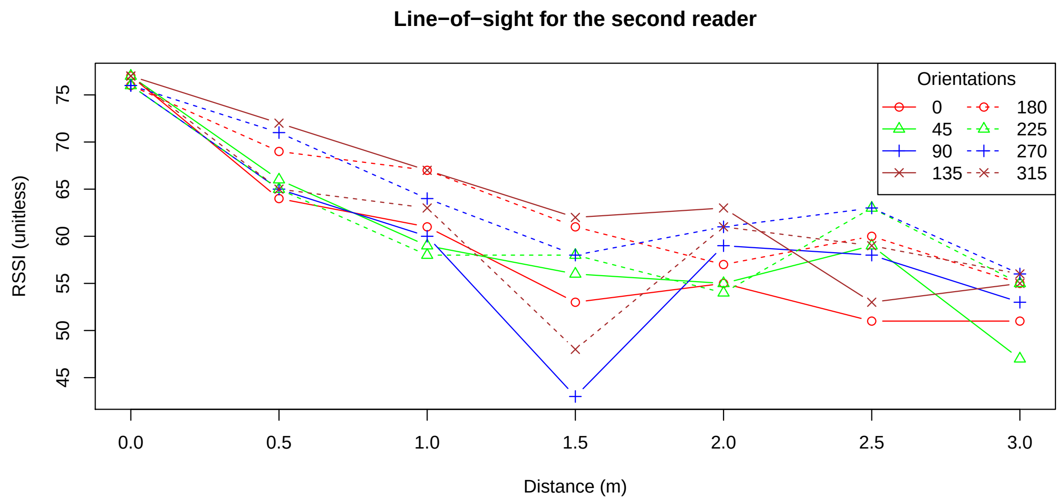


Figure A.2: Plot of RSSI measurements of different orientations of the tag with respect to the second reader. There was a line of sight between the RFID devices.

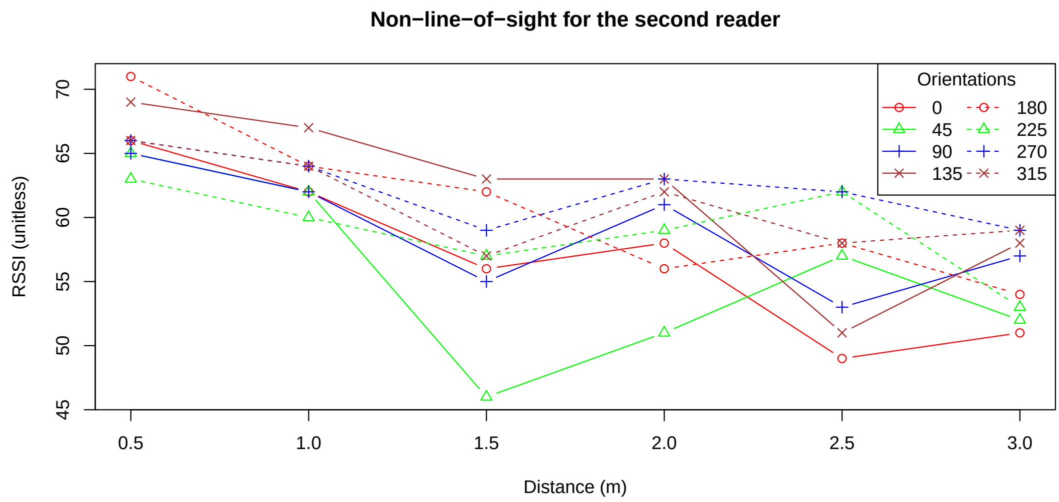


Figure A.3: Plot of RSSI measurements of different orientations of the tag with respect to the second reader. The tag was obscured by an object.

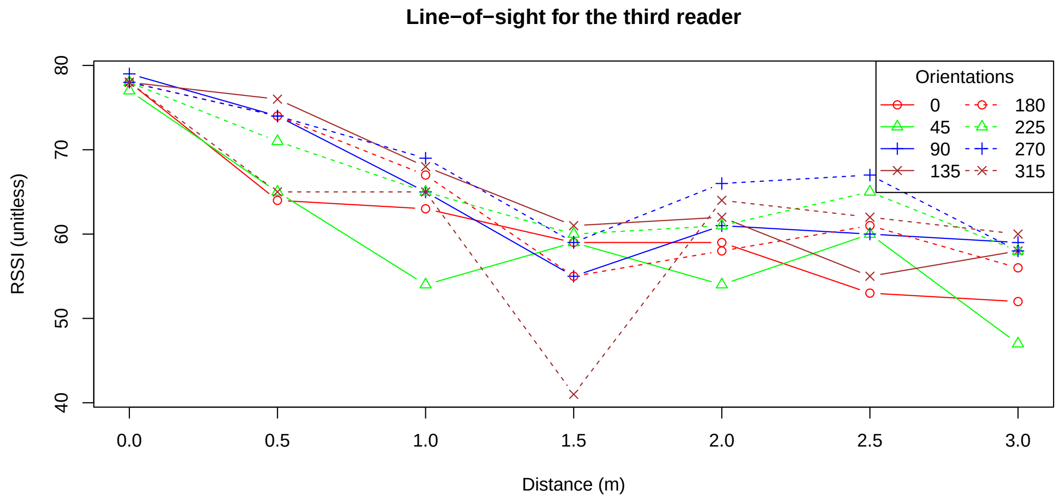


Figure A.4: Plot of RSSI measurements of different orientations of the tag with respect to the third reader. There was a line of sight between the RFID devices.

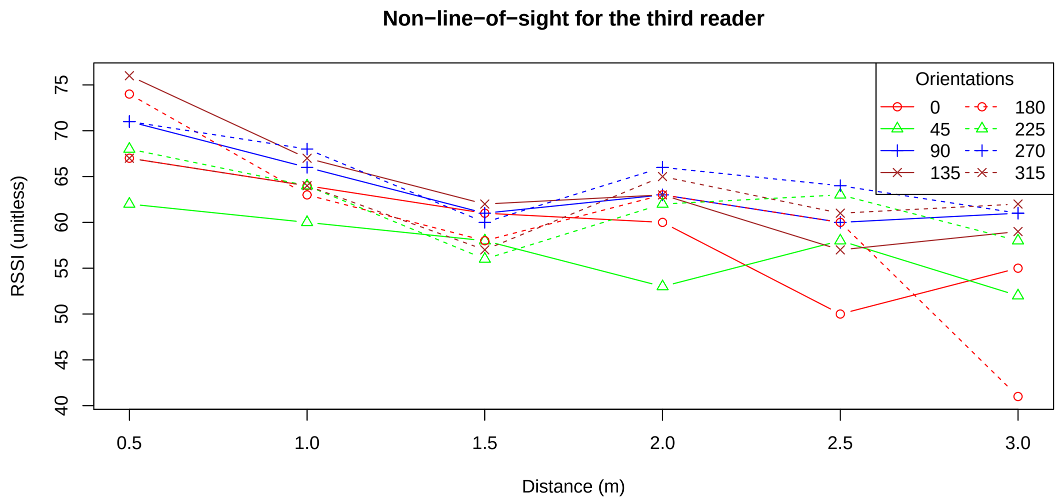


Figure A.5: Plot of RSSI measurements of different orientations of the tag with respect to the third reader. The tag was obscured by an object.

A.4 Grid of positions

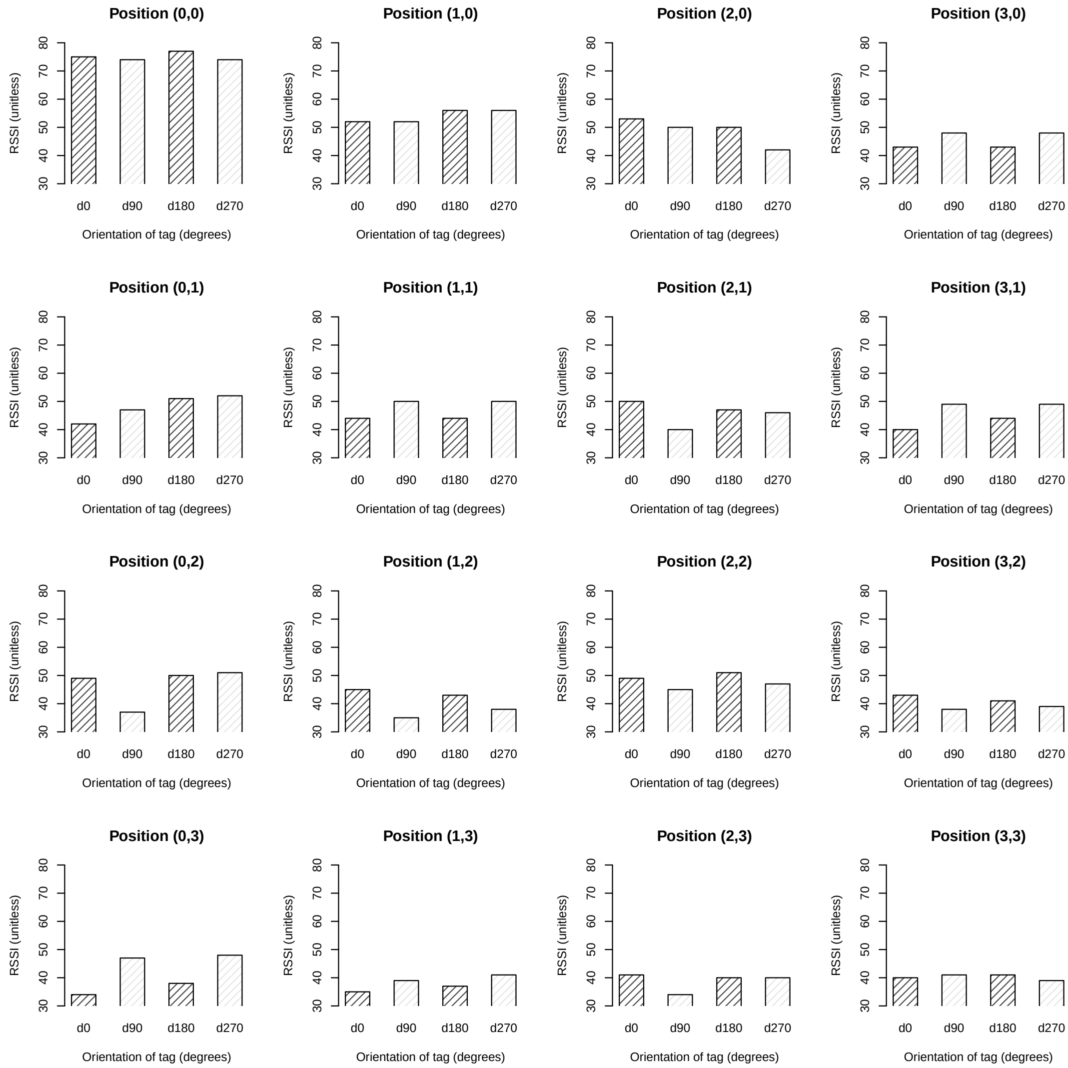


Figure A.6: Sixteen plots organised into a four by four grid. Each plot represents the RSSI measurements of the **second** reader when the tag was placed at different positions on the x and y axes of the grid. The positions of the tag were measured in meters. The four bars in each plot show the RSSI readings when the tag was facing right (0°), up (90°), left (180°), and down (270°).

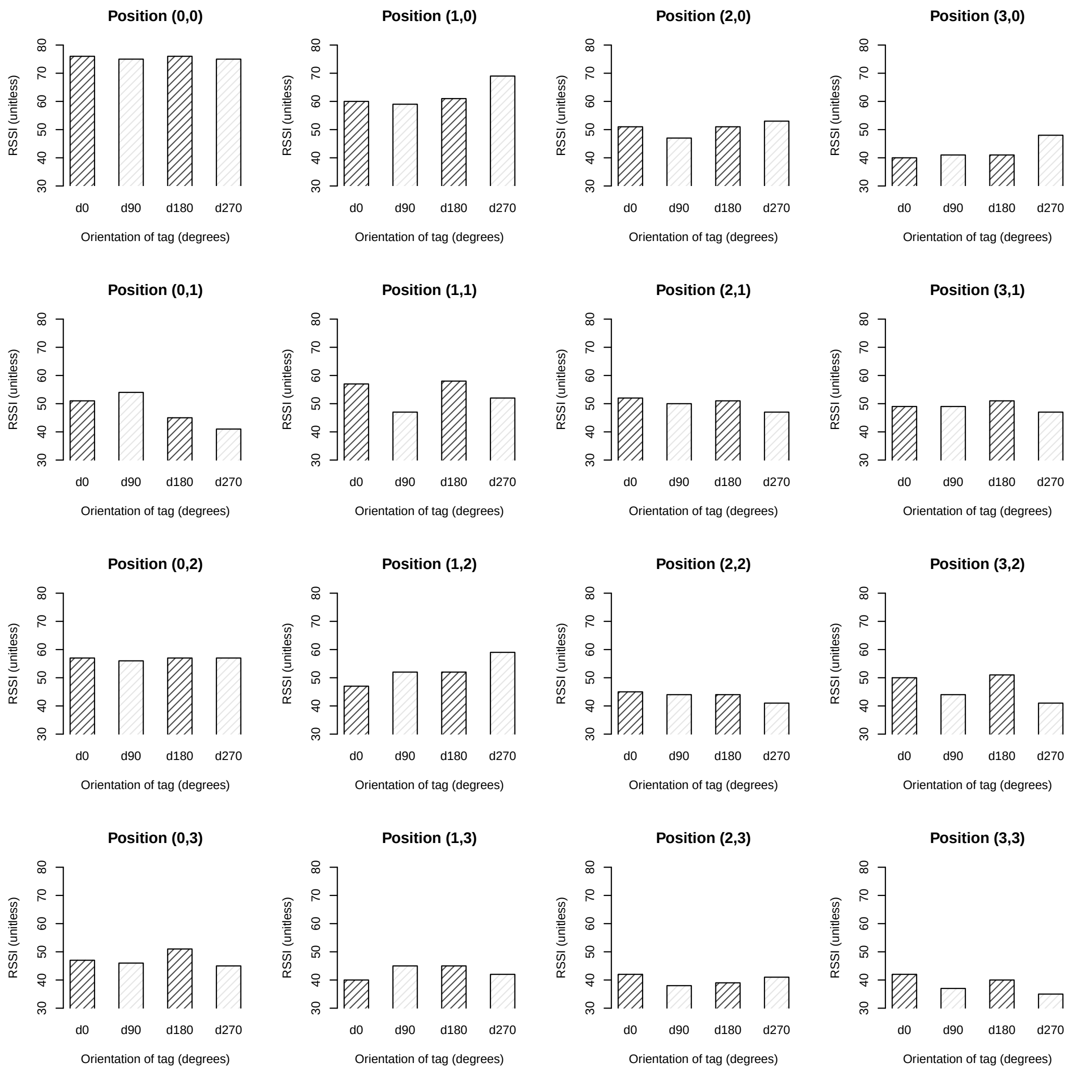


Figure A.7: Sixteen plots organised into a four by four grid. Each plot represents the RSSI measurements of the **third** reader when the tag was placed at different positions on the x and y axes of the grid. The positions of the tag were measured in meters. The four bars in each plot show the RSSI readings when the tag was facing right (0°), up (90°), left (180°), and down (270°).

Bibliography

[Balanis, 2012] Balanis, C. A. (2012). *Antenna theory: analysis and design*. John Wiley & Sons.

[Bolic et al., 2010] Bolic, M., Simplot-Ryl, D., and Stojmenovic, I. (2010). *RFID systems: research trends and challenges*. Wiley.

[Bouet and dos Santos, 2008] Bouet, M. and dos Santos, A. L. (2008). RFID tags: Positioning principles and localization techniques. In *Wireless Days, 2008. WD'08. 1st IFIP*, pages 1–5. IEEE.

[Cangialosi et al., 2007] Cangialosi, A., Monaly, J. E., and Yang, S. C. (2007). Leveraging RFID in hospitals: Patient life cycle and mobility perspectives. *Communications Magazine, IEEE*, 45(9):18–23.

[Chintalapudi et al., 2010] Chintalapudi, K., Padmanabha Iyer, A., and Padmanabhan, V. N. (2010). Indoor localization without the pain. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, pages 173–184. ACM.

[Cook et al., 2005] Cook, B., Buckberry, G., Scowcroft, I., Mitchell, J., and Allen, T. (2005). Indoor location using trilateration characteristics. In *Proc. London Communications Symposium*, pages 147–150.

[Elnahrawy et al., 2004] Elnahrawy, E., Li, X., and Martin, R. P. (2004). The limits of localization using signal strength: A comparative study. In *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, pages 406–414. IEEE.

[Finkenzeller, 2010] Finkenzeller, K. (2010). *RFID handbook: fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication*. Wiley.

[Hightower and Borriello, 2001a] Hightower, J. and Borriello, G. (2001a). A Survey and Taxonomy of Location Sensing Systems for Ubiquitous Computing. UW CSE 01-08-03, University of Washington, Department of Computer Science and Engineering, Seattle, WA.

[Hightower and Borriello, 2001b] Hightower, J. and Borriello, G. (2001b). Location systems for ubiquitous computing. *Computer*, 34(8):57–66.

[Hightower et al., 2000] Hightower, J., Want, R., and Borriello, G. (2000). SpotON: An indoor 3D location sensing technology based on RF signal strength. *UW CSE 00-02-02, University of Washington, Department of Computer Science and Engineering, Seattle, WA*, 1.

[Hunt et al., 2007] Hunt, V. D., Puglia, A., and Puglia, M. (2007). *RFID: a guide to radio frequency identification*. Wiley-interscience.

[Khan and Antiwal, 2009] Khan, M. A. and Antiwal, V. K. (2009). Location estimation technique using extended 3-D LANDMARC algorithm for passive RFID tag. In *Advance Computing Conference, 2009. IACC 2009. IEEE International*, pages 249–253. IEEE.

[Landt, 2005] Landt, J. (2005). The history of RFID. *Potentials, IEEE*, 24(4):8–11.

[Lui et al., 2011] Lui, G., Gallagher, T., Li, B., Dempster, A. G., and Rizos, C. (2011). Differences in RSSI readings made by different Wi-Fi chipsets: A limitation of WLAN localization. In *Localization and GNSS (ICL-GNSS), 2011 International Conference on*, pages 53–57. IEEE.

[Manolakis, 1996] Manolakis, D. E. (1996). Efficient solution and performance analysis of 3-D position estimation by trilateration. *Aerospace and Electronic Systems, IEEE Transactions on*, 32(4):1239–1248.

[Nath et al., 2006] Nath, B., Reynolds, F., and Want, R. (2006). RFID technology and applications. *Pervasive Computing, IEEE*, 5(1):22–24.

[Ni et al., 2004] Ni, L. M., Liu, Y., Lau, Y. C., and Patil, A. P. (2004). LANDMARC: indoor location sensing using active RFID. *Wireless networks*, 10(6):701–710.

[Parameswaran et al., 2009] Parameswaran, A. T., Husain, M. I., and Upadhyaya, S. (2009). Is rssi a reliable parameter in sensor localization algorithms: An experimental study. In *Field Failure Data Analysis Workshop (F2DA09)*.

[Rida et al., 2010] Rida, A., Yang, L., and Tentzeris, M. M. (2010). *RFID-Enabled sensor design and applications*. Artech House Publishers.

[Schlaikjer and Westman, 1962] Schlaikjer, J. and Westman, H. (1962). *Reference Data for radio engineers*. Internat. Telephone and Telegraph Corporation.

[Swartz, 2004] Swartz, N. (2004). Tagging toothpaste and toddlers. *Information Management Journal*, 38(5):22.

[Want, 2006] Want, R. (2006). An introduction to RFID technology. *Pervasive Computing, IEEE*, 5(1):25–33.

[Weinstein, 2005] Weinstein, R. (2005). RFID: a technical overview and its application to the enterprise. *IT professional*, 7(3):27–33.

[Wikipedia, 2013] Wikipedia (2013). Trilateration — Wikipedia, The Free Encyclopedia. `http://en.wikipedia.org/w/index.php?title=Trilateration&oldid=564380696`. [Online; accessed 31-July-2013].

[Youssef and Agrawala, 2005] Youssef, M. and Agrawala, A. (2005). The Horus WLAN location determination system. In *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, pages 205–218. ACM.

[Zhang et al., 2009] Zhang, Y., Yang, L. T., and Chen, J. (2009). *RFID and Sensor Networks: Architectures, Protocols, Security and Integrations*. CRC.

[Zhao et al., 2007] Zhao, Y., Liu, Y., and Ni, L. M. (2007). VIRE: Active RFID-based localization using virtual reference elimination. In *Parallel Processing, 2007. ICPP 2007. International Conference on*, pages 56–56. IEEE.