HUMAN TRACKING AND LOCALIZATION IN INDOOR USING WIRELESS SENSOR NETWORKS

BY

BILAL SAEED

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER NETWORKS

May 2015

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **BILAL SAEED** under the direction of his thesis adviser and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER NETWORKS**.

Thesis Committee

Dr. Tarek R. Sheltami (Adviser)

Dr. Ashraf S. Hassan Mahmoud (Member)

Dr. Hosam Rowaihy (Member)

Dr. Ahmad AlMulhem
Department Chairman

Dr. Salam A. Zummo
Dean of Graduate Studies

1/6/15

Date

To my loving parents and wife

ACKNOWLEDGMENTS

I would like to express my special appreciation and thanks to my advisor Professor Dr. Tarek R. Sheltami, you have been a tremendous mentor for me. I would like to thank you for encouraging my research and for allowing me to grow as a research scientist. Your advice on both research as well as on my career have been priceless. I would also like to thank my committee members, Dr. Ashraf Mahmoud and Dr. Hosam Rowaiy for serving as my committee members even at hardship. I also want to thank you for letting my defense be an enjoyable moment, and for your brilliant comments and suggestions, thanks to you. I would especially like to thank Mr. Ya'u, Dr. Anas, Mr. Farouq Aliyu, Mr. Bilal Jehanzeb, Muhammad Ijaz Jr, Muhammad Ijaz Sr, Muhammad Musaddiq, Shehryar, and all my friends at KFUPM. All of you have been there to support me when I felt down. Thank you so much.

A special thanks to my family. Words cannot express how grateful I am to my mother, father, my mother-in-law, and father-in-law for all of the sacrifices that you've made on my behalf. Your prayer for me was what sustained me thus far. I would also like to thank all of my friends who supported me in writing, and incented me to strive towards my goal. At the end I would like express appreciation to my beloved wife who spent sleepless nights with and was always my support in the *moments when there was no one to answer my queries. Thank you Tayyebah.*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

THESIS ABSTRACT

NAME: Bilal Saeed

TITLE OF STUDY: Human Tracking and Localization in Indoor using Wireless Sensor Networks

MAJOR FIELD: Department of Computer Engineering

DATE OF DEGREE: May 2015

The last advances in wireless communications and electronics have motivated the appearance of Wireless Sensor Networks. These networks are formed by a new kind of low-power and low-cost sensors able to operate across short ranges. Their simplicity and autonomy have motivated the development of many final applications in a large variety of fields. Nevertheless, sensor nodes are equipped with limited data processing and communication capabilities. Hence, several design challenges appear when an application has to be developed. These restrictions justify the design of highly distributed and energy-efficient applications.

Localization and tracking algorithms are one of those emerging applications that have become an interesting field to the researchers. The information routing is often supported by their localization. Besides, the location knowledge gives to the

data sensed a geographic sense. Instead of using the existing global localization methods, such as GPS, that are more complex and costly, recent advances have demonstrated the viability of local methods.

In this MS thesis, we have focused our study of the localization and tracking algorithms for WSN on the RSS and Time Difference of Arrival-based distributed approaches. One of the major issues is to obtain the simplest possible method, and RSS range measurements have become the simplest existing measurements. Besides, we have also presented methods that are able to optimize the trade-of between accuracy versus energy-efficiency.

First, RSSI-based cooperative localization algorithms in static indoor networks are considered. The use of RSSI measurements requires the knowledge of a propagation model in order to obtain inter-node distance estimates. We introduce an on-line path loss estimation method that obtains the model by means of RSSI measurements. Hence, we avoid the need of an a priori estimation of the propagation model.

Fine-Grained localization algorithm is developed using RSSI Secondly, TDOA-based localization algorithm is considered. TDOA requires WSN nodes to be equipped with RF and Ultrasonic transceivers, hence more energy is needed to operate these nodes. The good thing about TDOA estimation is no calibration is required among wireless sensor nodes and no a priori measures need to be taken.

Finally, we have considered the evaluation of these two algorithms. The results have shown that TDOA estimations are more accurate than RSSI based algorithm, *but TDOA consumes more energy hence there is a trade-off between accuracy and energy efficiency.*

ملخص الرسالة

الاسم الكامل: بلال سعيد

عنوان الرسالة: تتبع حركة ومكان الاشخاص في الاماكن الداخلية المغلقة باستخدام شبكات المجسات اللاسلكية

التخصص: هندسة الحاسب الالي

تاريخ الدرجة العلمية: أيار 2015

ان التطور الملحوظ في الاتصالات اللاسلكية والالكترونيات ساهم في ظهور شبكات المجسات اللاسلكية. تتكون شبكات المجسات اللاسلكية من نوع متطور من المجسات قليلة الثمن وموفرة في استهلاك الطاقة ولديها القدرة على العمل والاتصال في نطاق قصير. ان بساطة هذه الشبكات وقدرة التحكم الذاتي فيها ساعد على تطوير تطبيقات عديدة في مجالات مختلفة. بالرغم من التطبيقات العديدة للمجسات الا ان قدرتها على المعالجة والاتصال محدودة. لذلك فان الكثير من التحديات تظهر عند تصميم مثل هذه التطبيقات. ان هذه التحديات تبرر تصميم تطبيقات موزعة وفعالة من ناحية استهلاك الطاقة.

لقد اصبحت خوارزميات تتبع وتحديد مواقع المجسات من المواضيع المهمة لدى الباحثين. التوجيه المبني على المعلومات هو احد تطبيقات خوارزميات التتبع. معرفة الموقع يعطي المعلومات بعدا جغرافيا بالاضافة الى البعد المعرفي. التطورات الاخيرة في مجال تحديد المواقع اثبتت امكانية استعمال وجدوى الطرق المحلية بدلا من استعمال الطرق العالمية المعقدة وغالية الثمن مثل تقنية GPS.

في هذا العمل، كان التركيز على خوارزميات التتبع و تحديد المواقع المحلية في شبكات المجسات اللاسلكية بالاعتماد على نهج قوة الاشارة المستقبلة (RSS) وفارق الوقت المستقبل من الانظمة المحيطة. ان نهج RSS سهل وبسيط ويمكن الحصول عليه بطرق سهلة. بالاضافة الى ذلك فقد قدمنا طرق جديدة لتحسين المواءمة بين الدقة وفعالية الطاقة في طرق تحديد الموقع للمجسات.

في هذا العمل تم اعتبار خوارزمية تحديد المواقع التعاونية المعتمدة على تقنية RSS في الشبكات الثابتة والداخلية. ان استعمال تقنية RSS يحتاج الى استخدام نموذج انتشار من اجل تحديد المسافات بين عقد النظام. قدمنا ايضا طريقة لتقدير فقدان المسار (Path Loss) باستخدام قراءات RSS. بذلك نستطيع الاستغناء عن التقدير الاولي لنموذج الانتشار.

بالاضافة لذلك قمنا بتقديم خوارزمية لتحديد الموقع بالاعتماد على TDOA. ان طريقة TDOA تتطلب ان تحتوي عقدة النظام على نظام البث والاستقبال من نوع RF و اخر من نوع الفوق صوتي (Ultrasonic) وبالتالي هناك زيادة في استهلاك الطاقة. الشيء الجيد في طريقة TDOA هو انها لا تتطلب تعيير عقد النظام اللاسلكية ولا حاجة الى اجراءات اولية.

اخيرا، قمنا بتقييم الخوارزميات المذكورة اعلاه. اظهرت النتائج ان تقديرات خوارزمية TDOA اكثر دقة من تقديرات خوارزمية RSSI لكن طريقة TDOA تستهلك طاقة بكمية اكثر حيث انه هناك مواءمة بين الدقة واستهلاك الطاقة.

CHAPTER 1

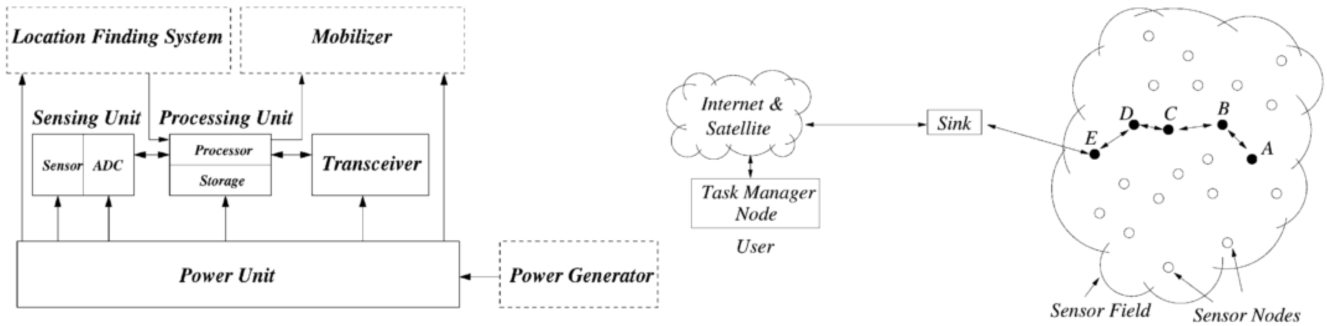
INTRODUCTION

The advancement in micro-electro-mechanical systems (MEMS) and wireless networking technologies have made it possible to design small, inexpensive, and smart sensors. These sensors can be deployed in any environment (i.e. terrestrial/surface, underwater, underground and air) in order to collect some information, process it, and then transmit it to the sink.

1.1 Wireless Sensor Networks (WSN)

A WSN is typically formed by deploying many sensor nodes in an ad hoc manner. These nodes sense physical characteristics of the world. The sensors could be measuring a variety of properties, including temperature, acoustics, light, and pollution. Base stations are responsible for sending queries to and collecting data from the sensor nodes.

Some of the main characteristics of a networked sensor include: (1) small physical size, (2) low power consumption, (3) limited processing power, (4)



(a) A Sensor Node Architecture

(b) An example of WSN

Figure 1.1: A Sensor and Wireless Sensor Network [1]

short-range communications, and (5) a small amount of storage. The typical size of today's networked sensor is a couple square inches, but the ultimate goal of the SmartDust [2] project is to incorporate sensing, communication, processing, and power source all into the space of a few cubic millimeters.

A sensor has four main components [1]: a sensing unit, a processing unit with memory, a power source (usually AA batteries), and a transceiver. Depending on the application, it can also have one or more optional components, such as an actuator, positioning system, mobilizer, and power generator (see Figure-1(a)). These resources of sensor nodes are very limited.

These sensor devices collect environmental information, process it, and then transmit it to the sink that is connected to a task manager node through satellite or Internet [1] as shown in Figure-1(b). Individually, these resource-constrained devices appear to be of little value. Deploying these sensors in large scale across an area of interest, however, is when they can be most effective. Placing sensors in hostile or inaccessible regions may allow for data collection which was previously impossible. Spatial and temporal processing as well as dense monitoring is now feasible. The sensors must be able to form an ad hoc network and use collaborative techniques to monitor an environment and respond to users when appropriate.

1.1.1 Applications of WSN

Wireless sensor networks provide the means to link the physical world to the digital world. The mass production of integrated, low-cost sensor nodes will allow the technology to cross over into a myriad of domains. In the future, applications of wireless sensor networks will appear in areas we never dreamed. Listed below are just a few places where sensor networks can and will be deployed.

- Earthquake monitoring
- Environmental monitoring
- Factory automation
- Home and office controls
- Inventory monitoring
- Medicine
- Security

Although still in its infancy, wireless sensor network applications are beginning to emerge. A recent study on Great Duck Island in Maine used sensor networks to perform monitoring tasks without the intrusive presence of humans [3], When monitoring plants and animals in the field, researchers have become more concerned about the effects of human presence. In the Smart Kindergarten project [4], using pre-school and kindergarten classrooms as the setting, the plan is to embed networked sensor devices unobtrusively into familiar physical objects, such as toys. The environment will be able to monitor the interactions of children and teachers in order to promote the development of skills. Researchers at Wayne State University believe implanted biomedical devices called smart sensors have the potential to revolutionize medicine. Proposed applications include an artificial retina, glucose level monitors, organ monitors, cancer detectors, and general health monitors [5].

Realization of sensor networks needs to satisfy the constraints introduced by factors such as fault tolerance, scalability, cost, hardware, topology change, environment, and power consumption. Because these constraints are highly stringent and specific for sensor networks, new wireless ad hoc networking techniques are required [6].

1.1.2 Challenges in WSNs

Although some applications have shown promise, the field of wireless sensor networks still provides many challenges to researchers:

Data storage

Sensors are sampling the environment continuously. With the limited storage capacity of the networked sensors, volumes of data cannot be stored permanently. Data has to be compressed and filtered, aggregated with data from other nodes, and stale data must be purged. Should the data be stored in the network or should it be routed offline to a central server?

Energy efficiency

Some form of battery typically powers networked sensors. When large networks of sensors are deployed, they are expected to run unattended for long periods of time. Writing energy-efficient algorithms that conserve the battery could extend the lifetime of an application by months. Energy conservation techniques are to be designed at all of the networking layers, from the physical layer to the application layer, and for various applications.

Fault tolerance

In early generations of networked sensors, there are high malfunction and failure rates. In most sensor applications, it is not feasible for a human to physically traverse a region to repair and replace nodes. A significant percentage of sensor nodes may fail when deployed in hostile environments. Therefore, techniques must be provided by the system so that the application continues running without interruption when nodes become faulty or die.. . .

Localization

Using wireless sensor networks to locate or track things is an application that is attracting much attention lately. There are many sensor network protocols and applications that assume every node knows its location. How is this possible? If every node were equipped with a GPS component, both the financial and energy cost of a large sensor network would become exorbitant. If a small fraction of the nodes are aware of their location, is it possible for the remaining nodes to discover their location? . . .

Scalability

The applications that are envisioned for sensor networks in the near future will use thousands of sensors. How do you get thousands of nodes to self-organize and work together? Centralized algorithms must sometimes give way to distributed algorithms when applications are being considered for networks of this scale. The deployment and management of thousands of tiny devices are issues that must be addressed.

Security

Any network application that uses a wireless medium inherently assumes a security risk. Eavesdropping to obtain information and jamming to deny service [7] are a couple of ways that a sensor network system may be attacked. The SPINS protocol [8] proposed basic building blocks for authenticated and private communication in sensor networks, but the traditional encryption techniques are not always plausible for the resource-constrained devices. What can be done to make sure a wireless sensor network provides important features such as availability, reliability, freshness, and privacy? . . .

1.2 Importance of the work

Wireless tracking and localization has become very important in recent times. These kind of systems introduce new services in automation known as object location detection. There are many applications for wireless tracking and locating such as tracking products in a mall or a warehouse, medical care applications, like tracking the health status of patients or tracking hospital equipment, navigation, security for tracking suspects and tracking cars for transportation companies.

In the last decade major progress has been made in tracking and localization in both research and industrial applications. Wireless systems have been growing and are used in many fields like medical, industrial, transport and logistics. Since in many of these applications tracking people or locating objects is needed, accurate location is very important for both indoor and outdoor environments [9], [10]. The procedure for locating and tracking objects is called localization, location sensing, position location, geolocation or radiolocation.

The main purpose of a localization or a tracking algorithm is to estimate the position of those non-located nodes with the following information: a priori knowledge of some nodes positions and inter-sensor measurements, such as time difference of arrival, angle of arrival or connectivity. Hence, the majority of existing localization methods applied in WSN tries to achieve the best accuracy considering the restrictions that this kind of networks imposes. Although nowadays there are many methods of localization in wireless networks, such as GPS or radar-based geolocation techniques, there exist many challenges that limit their usage. Wireless sensor nodes have normally low-cost hardware with a limited computational capability. Hence, the localization algorithms have to take into account them in order to achieve the best trade-off between cost, size, energy consumption and accuracy. For that reason, localization has become a challenging field.

1.3 Motivation

The motivation behind this work is the wide range of applications for wireless sensor networks tracking and localization in many fields like navigation, security and healthcare. If the accuracy of wireless tracking is increased, the application and the use of this technique would be increased and extended to other fields of use.

The crucial technical issue is that current wireless localization technology does not offer sufficient resolution to permit tracking to this level of accuracy, or it is expensive. The work in this thesis will advance the state of the art of wireless localization by a combination of theoretical analysis and simulation all with the unifying aim of permitting the tracking of mobile targets within the network. Improvements in accuracy is obtained by considering all the factors that may affect RSSI, a recently proposed strategy for improving indoor localization, and the ability to analyze data afterwards to improve its quality.

1.4 Significance

The significance of this work is the improvement of the accuracy of tracking and localization in WSN, which was done by building a network and tracking the location of mobile nodes using two major measurement models such as Time Difference of Arrival, Time of Arrival, Radio Signal Strength Indicator, and Angle of Arrival.

Moreover, this thesis is mainly based on pragmatic approaches. Performance of algorithms is evaluated on real network and are implemented on real WSN motes. The results have shown that the accuracy of these algorithms is much better than any existing system.

1.5 Problem Statement

For outdoor environments, GPS (Global Positioning System) plays a dominant role in localization [4]. However, it does not work well in indoor scenarios. This inefficiency is due to the weakness of signals emitted by GPS and their disability to penetrate most building materials. Therefore, GPS does not fit well in indoor environments where people spend most of their time. Even though GPS devices are becoming more and more promising and ponderable in the future and are able to provide sufficient precision for outdoor use, other effective technologies are demanded for indoor human/object tracking. To fulfill this requirement, various indoor localization technologies have been developed in the literature [5]. However, due to the complexity of indoor environments, the development of an indoor localization technique is always accompanied with a set of challenges, e.g. NLOS (none line of sight), multipath effect, and noise interference. These challenges result mainly from the influence of obstacles (e.g. walls, equipments, and human beings) on the propagation of electromagnetic waves. We need simple yet energy efficient localization algorithms that can provide better accuracy in the presence of aforementioned factors.

1.6 Research Objectives

Following are the research objectives of this thesis.

- Implementation of RSSI based localization and tracking algorithm on Telosb motes.
- Implementation of TDOA based localization and tracking algorithm on Arduino Due.
- Optimize the messages to be sent by target node. Every target node is equipped with an accelerometer, the purpose of accelerometer is that we do not want to overwhelm our anchor nodes with extra and useless messages. Target node only sends messages, if it moves from one position to another.
- Design and implementation of Ultrasonic receiver and transmitter

1.7 Methodology

To achieve the milestones of this research, following methodology will be used:

1.7.1 Survey of Related Work

A survey of the recent literature is done to identify the requirements for designing an algorithm in wireless sensor network for tracking and localizing targets in indoor.

1.7.2 Implementation of existing algorithms

RSSI and TDOA have been implemented on Telosb and Arduino respectively.

1.7.3 Designing of New Algorithm

Based on the study of existing algorithms, we have implemented TDOA algorithm on Arduino platform and as per our knowledge this work has not been done before on Arduino.

1.7.4 Prototype

After achieving above milestones, a prototype is developed using Telosb sensor motes and Arduino Microcontroller are programmed in TinyOS and Arduino Sketch respectively.

CHAPTER 2

AN OVERVIEW OF LOCALIZATION TECHNIQUES

2.1 Localization

Indoor Positioning or location estimation in indoor environments which use Wireless Sensor Networks (WSN) is an emerging research area in the field of wireless communications. There are many existing localization techniques and algorithms which accurately locate the target in outdoor. For example, GPS and Assisted GPS (A-GPS) are the most prominently used positioning technologies which have an accuracy range of 3-50 feet for outdoor [11].

However, these kinds of accuracy ranges would do no good to position a target in indoor environment. The accuracy in indoor environment needs to be more than what these technologies can provide. The accuracy of GPS in indoor environments can be increased, but not without the aid of additional hardware and software requirements, which is in fact a disadvantage. Our goal is to achieve the required accuracy using the existing software and hardware devices in Wireless Sensor Networks.

Indoor localization finds numerous applications like inventory tracking in large warehouses, tracking prisoners etc[12]. Positioning is also being used in offices using WSN as a security measure to decide whether a user is to be given access or denied access depending on his position.

2.2 Localization Techniques

Localization techniques can be classified into two major categories: Indoor and outdoor localization. The following sections briefly discuss both localization systems.

2.2.1 Global Positioning System (GPS)

The Global Positioning System (GPS) is a radio navigation system which offers reliable positioning, navigation and timing services. All a user required to utilize GPS system is a GPS receiver. Moreover, this GPS service is free for all. Let us see the working principles of GPS in the following discussion. The working of GPS needs 3 components:

- Satellites revolving around the earth.
- GPS Base Stations on the earth which monitors the processing on earth.

- A GPS receiver.

A GPS receiver needs information about the following to calculate its location.

- The timestamp at which the satellite sent the message.
- The orbital information.
- Rough orbits estimation of all the GPS satellites

Using the timestamp information sent by the satellite, the GPS receiver calculates the time taken by the signal to travel from the satellite to itself. By using this transit time information, the GPS receiver calculates the distance to each of the satellite. Now, using these distances, Geometric Trilateration is used to find out the location of the GPS receiver [13].

Trilateration is a geometric method used to calculate the common area of intersection of three circles given the center and radii of all the circles [14]. The concept of trilateration is quite simple. The distance estimation of the satellites to the GPS receiver (say r1, r2, r3) are calculated as discussed above and the positions of the satellites (say S1, S2, and S3) are known to the GPS receiver. The locus of a point at a fixed distance from a fixed point is nothing but a circle. Geometrically these circles can be represented in Fig. 2.1.

Two circles and their intersection yield two possible points of intersection which means we cannot be sure which of these two intersection points the target lies. This is where the third circle comes into place. The third circle is needed to find

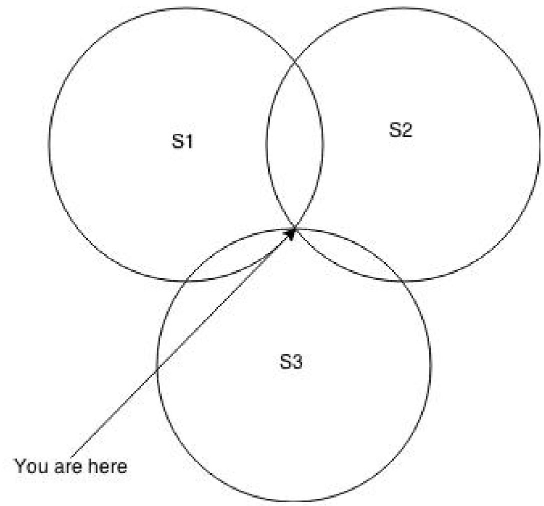


Figure 2.1: Trilateration example

out the user position as shown in the figure 2.1 accurately. This is the case for finding out the location of the used in two dimensions.

2.3 Indoor Localization and its elements

Above mentioned localization system GPS works perfectly when it is used in outdoors but the very system gives very poor performance when used in indoor applications. This is due to GPS uses GEO satellites to locate a target in 3-d space and signals from GEO satellites suffers very high interference in indoor because of concrete and steels. Our idea is to use low cost and power Wireless Sensor Networks instead of GPS in indoor to detect and localize a target.

In any localization system we first get the distance distance measurements based on, RSSI, angle, and time between reference and unknown nodes, then the location of unknown sensors is determined using geometric techniques. Hence, the most important elements of any localization system are distance measurements, geometry constraints, time measurements and angle measurements. In the following sections we discuss the most used localization techniques.

2.3.1 Received Signal Strength Indication (RSSI)

The main characteristic of radio propagation is that the signal strength attenuates as the distance between transmitter and receiver increases. The power of the signal decreases exponentially as the distance increases and receiver can determine this attenuation based on RSSI in order to measure the distance to the transmitter. Receiver measures the power of the signal based on RSSI. Based on the transmitter power the propagation model and propagation loss are determined and they can be mapped into distance measurements. RSSI method is used mainly for RF signals.

In [15], they propose radio propagation models and these models are used to measure the RSSI at any given distance away from the transmitter. Propagation model in free space is given below.

$$P_r = \frac{P_\lambda G_t G_r \lambda^2}{4\pi d^n L} \tag{2.1}$$

This model measures the received power in terms of the distance between transmitter and receiver. In this model, P_r is the received power at distance d away from the transmitter, P_λ is the transmitter power, G_t is the transmitter gain, G_t is the receiver gain, L is the system loss, and λ is the wavelength of the system.

In [16], authors use WINS sensor nodes to estimate the distance based on RF signal strength. In their experimental work, they try different configurations, including different power levels in transmitter and sensor deployment strategies to determine the distance between transmitter and receiver based on RSSI.

As it has been discussed earlier in this section the signal strength decreases exponentially with the increase in distance between transmitter and receiver. This can be written in terms of the following equation,

$$PL(d) \propto \left(\frac{d}{d_o}\right)^n \tag{2.2}$$

where PL is the path loss, n is the path loss exponent, d is the distance between transmitter and receiver, and d_o is the reference distance approximately equals to 1m.

equation 2.2 can also be written in logarithmic form as,

$$P(d) = P_o(d_o) - 10n \log(\frac{d}{d_o}) + X_\sigma \tag{2.3}$$

where $P(d)$ is the received power at distance d away from transmitter, $P_o(d_o)$ is the reference power at $d_o = 1m$, X_σ is the zero mean Gaussian random variable with standard deviation of σ. The value of path loss exponent n largely depends on the surroundings. With clear Line of Sight the value of n can be 2. In the presence of obstacles, n will have a larger value. Table 2.1 shows path loss exponent n values according to the environment. In our calculations, to make things easier we have ignored factors such as human movement, short/long term fading, shadowing, and other factors.

environment	**Path Loss exponent, n**
Free Space	2
Urban area Cellular radio	2.7 to 3.5
Shadowed urban cellular radio	3 to 5
In-building Line of sight	1.6 to 1.8
Obstructed in-building	4 to 6
Obstructed in factories	2 to 3

Table 2.1: Path Loss exponent n values in various environment

It is possible to find distance d based on the knowledge of parameters in equation 2.3 [15]. The distance, however, can not be very accurate because path loss exponent is not the only parameter that attenuates signal strength, factors such as shadowing, multi-path, and reflection also degrade the signal quality [14]. The relation between Received power in dBm vs distance is shown in Fig. 2.2. This figure is drawn in Matlab.

The adoption of path loss exponent model is motivated by experimental results such as those provided by [[17], [18]] and the analytical study by [19]. Some results in terms of accuracy are presented in [20]. The authors carried out a measurement campaign using TelosB motes [21]. The average error achieved in the measured distances is 2.25 m, for distance between 1 and 8 meters. The RSS-based estimates achieve worse accuracy compared to that achieved with time measurements. However, the major advantage is that RSS provides a lower complex solution. Moreover, the accuracy of the distances estimates can be improved if a more accurate propagation model is used [19].

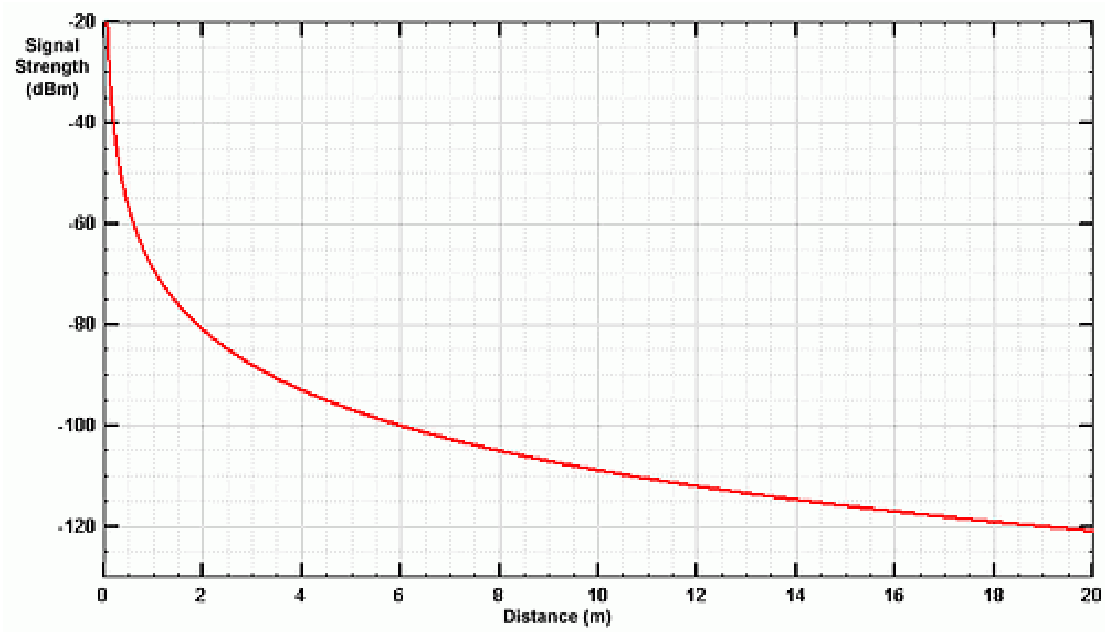


Figure 2.2: The power of the received radio signal strength attenuates exponentially with the increase of distance between the transmitter and receiver.

2.3.2 Time of Arrival (TOA)

Time of arrival (TOA) measurement reects the time at which any kind of signal arrives at a receiver [10]. More specically, TOA is equal to the transmission time plus the propagation delay between a transmitter and a receiver, which reects the inter-node distance between them. Hence, assuming a constant propagation velocity, it is easy to obtain an estimated distance. This velocity depends on the type of signal, e.g. 1 ms becomes to 0.3 m for an acoustic propagation and 1 ns becomes 0.3 m for an RF signal.

The major sources of error that affect time of arrival measurements are noise and multipath signals. Moreover, another limitation that imposes TOA measures is the synchronization between nodes. Maintaining the same reference clock among all the network is a must because if nodes do not have it, the estimated propagation time at the receiver will have an inherent error.

errors about 2% are achieved over a communication range of 3-6 m [22]. Moreover, the nodes clocks resolution should be of the order of nanoseconds. TDOA is a range measure usually used in wireless or satellite networks [23] (in which base stations and mobile nodes are synchronized), but a recent trend uses time measurements approaches with Ultra Wide Band signals [24]. The UWB signal achieves a high accuracy because the transmitted pulses have a wide bandwidth, hence a very short pulse waveform. With the recovery of this transmitted pulse it is possible to estimate the distance between receiver and transmitter. Moreover, UWB signals achieve very high temporal resolutions.

2.3.3 Time Difference of Arrival (TDOA)

The synchronization requirement of the TOA measurements could be an important dis- advantage that limits their use in WSN. Although time measurements are also affected by noise and multipath the obtained accuracy is high. For this reason another time measurement approach is presented. This new approach avoids the necessity of having an entire network synchronized. This method is known as time difference of arrival (TDOA). TDOA is based on two different ideas [25].

- The first TDOA method [26] is based on the measurement of the difference between the arrival times of a signal sent by a transmitter at two receivers. This method assumes that the receiver locations are known and that the two receivers are perfectly synchronized (it could also be two synchronized transmitters and one receiver). It is not necessary to synchronize the entire network, only the receivers. For that reason, it is mostly used in the up-link cellular networks where the complexity of base stations is considerably relaxed.

- TDOA can also be understood as the difference of the arrival times of two different signals sent by a unique sender to a unique receiver (see [[27], [28]]). With this idea is avoided the necessity of synchronizing any node inside the network, neither senders nor transmitters. In TDOA method, transmitter transmits two difference different signals - Radio frequency and Ultrasonic signals. This is necessary in order to achieve this time difference at the arrival time. Hence, it is possible to measure the one-way propagation time. It can be done if nodes include extra hardware - Radio Frequency and Ultrasonic Transceivers in order to transmit different signals.

As already said, the first case is more appropriate in cellular networks due to the necessity of having the receiving terminals synchronized. In this kind of networks, base stations have less stringent requirements in terms of complexity or cost. Hence, it is possible to synchronize them having a priori knowledge of their position; then they are able to act as the receivers for estimating the TDOA information.

On the other hand, the second TDOA approach could be more suitable in a WSN. The major disadvantage is the necessity of including extra hardware in the entire network, thus, increasing the cost of the terminal. Moreover, WSN nodes are usually battery-powered terminals. The requirement of sending two signals will increase the energy consumption.

Both approaches obtain good accuracy. In [18] the results show an average error of the distance estimates between 29 cm and 8 cm. However, both methods do not increase the complexity of the network (first TDOA method requires synchronized reference nodes) neither the cost of nodes (second TDOA method requires more integrated hardware in order to transmit two different signals).

2.3.4 Round-trip Time of Arrival (RTOA)

As previously discussed, WSN nodes are simple low-power devices. For that reason, any algorithm that has to be applied to these devices is highly restricted. Both time measurement techniques presented, although they could be used in WSN, increases the cost and the complexity of the nodes. The previous time measurement techniques achieve good distance estimates but impose a higher cost and complexity to the network. In order to take advantage of the good accuracy achieved by time measurements but trying to minimize these high requirements, a two-way time measurement technique is presented.

The Round-Trip Time method [29] avoids the synchronization constraint that TOA or first TDOA methods impose, and also the hardware requirements of the second TDOA method. The measurement starts when a node A sends a packet to a node B. When the node B receives the packet, it re-transmits it to the node A. At the end, node A receives the packet; hence, it can calculate the propagation time because the difference between the sending time and the receiving time at node A is twice the propagation time plus the processing time at node B (obtained from specifications or estimated at calibration time).

The major advantage of RTT method is the avoidance of the clock synchronization requirement or the inclusion of extra hardware. Nevertheless, a double transmission is necessary in order to obtain a time measurement. Furthermore, the time delay between the reception and the re-transmission is not perfectly known, hence range measurements will have an inherent error. Numerical results based on different experimental setups are presented in [30]. The results show that RTT measurements lead to an RMS error between a minimum of 75 cm and a maximum of 2.51 m. The difference in accuracy compared to that achieved by TOA or TDOA measurement is remarkable.

Furthermore, although the hardware and synchronization requirements are eliminated this technique needs the exchange of more packets in order to estimate the inter-node distance.

2.3.5 Angle of Arrival (AOA)

Compared to time or RSS measurements, AOA techniques do not estimate distances. AOA measurements estimate the direction of arrival of the signal transmitted by neighbor nodes [11].

In [31] two different ways of estimating angles are presented. First method, which is the most common, estimates the angle of arrival by means of using a sensor array; hence array signal processing is employed. each sensor requires two or more sensors placed at a known location. These sensors could be microphones, if acoustic signals are sensed, or it could be antennas, if RF signals are used. The angle is esti- mated following the same approach as in time-delay estimation. On the other hand, in [2] an AOA algorithm based on antenna arrays is presented. The measurement consists of two phases. At first phase, anchors transmit their location and a short omnidirectional pulse. Then, they transmit a beacon with a rotating radiation pattern. Taking the advantage of beamforming techniques, the anchors are able to transmit directional pulses every T seconds and changing the direction of the signal by a constant angular step $\triangle\beta$ Sensors have to register the arrival time between the first omnidirectional signal and the time of arrival of the pulse with maximal beacon power. This difference in time ($\triangle t$) allows the sensor to estimate the angle of arrival as:

$$\beta = \triangle\beta \frac{\triangle t}{T} \tag{2.4}$$

The accuracy achieved in [1] is an average error of 2 m in a scenario with 6 anchors and 100 non-located nodes uniformly distributed in a 50 m x 50 m area. Increasing the anchors one can achieve root mean square errors (RMSe) in the localization below the 1.5 m.

In [6], RSS measurements from directional antenna arrays on each node were also used to estimate arrival angles. Through the ratio RSS between these two antennas differently oriented it is possible to extract the angle of arrival of the signal. Accuracy errors below 1 meter are achieved compared to that achieved with distance-based algorithms.

However, the necessity of a higher number of antennas in the nodes makes these measurements not easy to implement. The increase of cost and size of the nodes makes the AOA a more complex solution although achieves good results in terms of localization accuracy.
The AOA measurements are used with the triangulation localization algorithm. The triangulation estimates the node position by means of angles between xed node and reference nodes.

CHAPTER 3

LOCALIZATION IN 3-DIMENSION - GEOMETRIC

In this theses we have used multilateration to locate a target in 3-dimensional space. In the following sections we cover localization techniques in terms of geometry. In this chapter we start our calculations with 3-D space, but in our experiments we have only considered 2-D space.

3.1 Solution Techniques

The obvious approach in solving this positioning technique is to treat the coordinates of the target in three dimensional space $T(x, y, z)$ as the point of intersection of several spheres, whose centers are the locations of the n reference Nodes $R(x_i, y_i, z_i)$ for $i = 1, 2, ...k$. The exact distances between the reference and target node in 3-dimensional space, r_i, are the radii of the individual spheres.

The equation for any of these spheres is given below.

$$(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 = r_i^2 \tag{3.1}$$

The point of intersection of the surfaces of n of these spheres is obtained by letting $i = 1, 2, 3, ..., k$, and solving the resulting k non linear equations simultaneously to eliminate two coordinates. This solution is not feasible as it produces nonlinear equations of higher degrees. Furthermore, since the equations are quadratic, many cases for the signs would have to be considered.

Linearizing the system of equations geometrically converts the problem into one of finding the point of intersection of several planes. When the exact distances from four reference nodes are available, the solution of linear system is completely determined. There are three equations of, three unknowns and exactly one solution. Consequently, the theoretical minimum number of reference nodes is four. When approximate distances are used, the position that is calculated by the direct solution of the linear equations is no longer acceptable. The sophistication is needed when working with approximate distance is dealt with in the linear least squares and nonlinear least squares solution techniques.

3.2 Linearized System of Equations

The solution of the linear system $\tilde{A}\vec{x} = \vec{b}$ is an easier approach over solving for the intersection of spheres. However, it is unacceptable because it does not the determine the locations within tolerance of five feet when used with approximate distances. The linear system which is developed below in eq. 3.9 can be used with exact distances and four arbitrarily reference nodes to accurately calculate an unknown location by determining the point of intersection of three planes. However, the straightforward solution of any three equations of the linear system eq. 3.9 will produce unacceptable results when approximate distances are used. The constraints are the equations of the sphere with radii x_{ij},

$$(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 = r_i^2 \quad (i = 1, 2, 3, ..., n) \tag{3.2}$$

The j^{th} constraint is used as a *linearizing* tool. Adding and subtracting x_j, y_j, z_j in equation 3.2 gives

$$(x - x_j + x_j - x_i)^2 + (y - y_j + y_j - y_i)^2 + (z - z_j + z_j - z_i)^2 = r_i^2 \quad (i = 1, 2, 3, ..., n) \tag{3.3}$$

with $(i = 1, 2, \ldots, j-1, j, j+1, \ldots, n)$.

Expanding and regrouping the terms in eq. 3.3 leads to

$$(x - x_j)(x_i - x_j) + (y - y_j)(y_i - y_j) + (z - z_j)(z_i - z_j)$$

$$= \frac{1}{2}[(x - x_j)^2 + (y - y_j)^2 + (z - z_j)^2 - r_i^2 + (x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2]$$

$$= \frac{1}{2}[r_j^2 - r_i^2 + d_{ij}^2] = b_{ij}, \tag{3.4}$$

where

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \tag{3.5}$$

is the distance between reference node i and j.

Since it does not matter which constraint is used as a linearizing tool, we arbitrarily select the first constraint ($j = 1$). This is analogous to selecting the first reference node. Since $i = 1, 2, 3, \ldots, n$, this leads to a linear system of $(n-1)$ equations with three unknowns:

$$(x-x_1)(x_2-x_1)+(y-y_1)(y_2-y_1)+(z-z_1)(z_2-z_1) = \frac{1}{2}[r_1^1 - r_2^2 + d_{21}^2] = b_{21} \tag{3.6}$$

$$(x-x_1)(x_3-x_1)+(y-y_1)(y_3-y_1)+(z-z_1)(z_3-z_1) = \frac{1}{2}[r_1^2 - r_3^2 + d_{31}^2] = b_{31} \tag{3.7}$$

$$\vdots$$

$$(x-x_1)(x_n-x_1)+(y-y_1)(y_n-y_1)+(z-z_1)(z_n-z_1) = \frac{1}{2}[r_1^1 - r_n^2 + d_{n1}^2] = b_{n1} \tag{3.8}$$

The linear system in eq. 3.8 can also be written in matrix form.

$$A\vec{x} = \vec{b} \tag{3.9}$$

with

$$A = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ \vdots & \vdots & \vdots \\ x_n - x_1 & y_n - y_1 & z_n - z_1 \end{bmatrix}, \vec{b} = \begin{pmatrix} x - x_1 \\ y - y_1 \\ z - z_1 \end{pmatrix}, \vec{x} = \begin{pmatrix} b_{21} \\ b_{31} \\ \vdots \\ b_{n1} \end{pmatrix} \tag{3.10}$$

The linear system Eq. 3.9 has $(n-1)$ equations with three unknowns. Therefore, theoretically only four reference nodes ($n = 4$) are needed to determine the unique position of the target in three dimensional space; provided no more than two reference nodes are co-linear.

3.3 Linear Least Squares

The coordinates of positions obtained by applying the linear least squares method to the linear system of equations 3.9 are generally more accurate than the coordinates obtained by solving four equations simultaneously. However, the accuracy of the coordinates calculated with the linear least squares method are unacceptable because they are not within the tolerance of five feet when used with approximate distances.

Since they distances r_i are only approximate, the problem requires the determination of $\vec{x}$ such that $\mathbf{A}\vec{x} \approx \vec{b}$. Minimizing the sum of the squares of the residuals,

$$S = \vec{r}^T \vec{r} = (\vec{b} - \mathbf{A}\vec{x})^T (\vec{b} - \mathbf{A}\vec{x}) \tag{3.11}$$

The above equation leads to the *normal* equation. 3.12

$$\mathbf{A}^T \mathbf{A}\vec{x} = \mathbf{A}^T \vec{b} \tag{3.12}$$

There are several methods to solve Eq. 3.12 for $\vec{x}$. The condition of $\mathbf{A}^T\mathbf{A}$ defines which method to use.

If $\mathbf{A}^T\mathbf{A}$ is non-singular then $\vec{x}$ can simply be obtained by

$$\vec{x} = (\mathbf{A}^T \mathbf{A})^{-1} A^T \vec{b} \tag{3.13}$$

If $\mathbf{A}^T\mathbf{A}$ is singular or poorly conditioned then the normalized **QR**-decomposition of $\mathbf{A}$ is generally used [32]. In this method $\mathbf{A} = \mathbf{QR}$, where $\mathbf{Q}$ is an orthogonal matrix and $\mathbf{R}$ is upper-triangular matrix. The solution for $\vec{x}$ in the normalized **QR**-decomposition is then can be determined from

$$\mathbf{R}\vec{x} = \mathbf{Q}^T \vec{b} \tag{3.14}$$

by back substitution when $\mathbf{A}$ is full rank.

3.4 Singular Value Decomposition (SVD)

Singular value decomposition takes a rectangular matrix of gene expression data (defined as **A**, where **A** is a n x p matrix) in which the n rows represents the genes, and the p columns represents the experimental conditions. The SVD theorem states:

$$A_{n*p} = U_{n*n} S_{n*p} V_{p*p}^T \tag{3.15}$$

Where

$$U^T U = In * n \tag{3.16}$$

$$V^T V = I_{p*p} \tag{3.17}$$

Where the columns of **U** are the left singular vectors (gene coefficient vectors); S (the same dimensions as **A**) has singular values and is diagonal (mode amplitudes); and $\mathbf{V}^T$ has rows that are the right singular vectors (expression level vectors). The SVD represents an expansion of the original data in a coordinate system where the covariance matrix is diagonal.

Calculating the SVD consists of finding the eigenvalues and eigenvectors of $\mathbf{AA}^T$ and $\mathbf{A}^T\mathbf{A}$. The eigenvectors of $\mathbf{A}^T\mathbf{A}$ make up the columns of **V** , the eigenvectors of $\mathbf{AA}^T$ make up the columns of **U**. Also, the singular values in **S** are square roots of eigenvalues from $\mathbf{AA}^T$ or $\mathbf{A}^T\mathbf{A}$. The singular values are the diagonal entries of the **S** matrix and are arranged in descending order. The singular values are always real numbers. If the matrix **A** is a real matrix, then **U** and **V** are also real.

CHAPTER 4

HARDWARE

In this theses we have implemented two localization techniques one is based on Received Signal Strength Indication (RSSI) and other is based on Time Difference of Arrival (TDOA). Both techniques are implemented on completely different hardware platforms. RSSI based localization system is implemented using TinyOS while TDOA is implemented on Arduino.

Our design has two main component one is the hardware component and other one is software. The hardware component for each technique is different, however,the software component is the same for both techniques.

In the following sections we briefly discuss the implementation of each technique and talk about software and hardware components that are used.

4.1 Hardware design of TDOA

In our work TDOA technique is implemented on Arduino. In the following sections we describe the hardware components that are used in its implementation. Table. 4.1 shows the list of components that are used for the implementation of TDOA technique.

Components	**Quantity**
Xbee Shield	6
Arduino Due Board	6
Xbee Explorer	1
5V Battery	6
MPU6050 Accelerometer	1
40khz Ultrasonic Transmitter	1
40khz Ultrasonic Receiver	3

Table 4.1: Hardware used for the implementation of TDOA

4.1.1 Arduino Due

The Arduino Due is a microcontroller board based on the Atmel SAM3X8E ARM Cortex-M3 CPU (datasheet). It is the first Arduino board based on a 32-bit ARM core microcontroller. It has 54 digital input/output pins (of which 12 can be used as PWM outputs), 12 analog inputs, 4 UARTs (hardware serial ports), a 84 MHz clock, an USB OTG capable connection, 2 DAC (digital to analog), 2 TWI, a power jack, an SPI header, a JTAG header, a reset button and an erase button. It contains everything needed to support the microcontroller; simply connect it to a computer. Table. 4.2 shows the specifications of Arduino Due micro-controller. Figure. 4.1 shows the picture of Arduini due.

Microcontroller	AT91SAM3X8E
Operating Voltage	3.3V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-16V
Digital I/O Pins	54 (of which 12 provide PWM output)
Analog Input Pins	12
Analog Outputs Pins	2 (DAC)
Total DC Output Current on all I/O lines	130 mA
DC Current for 3.3V Pin	800 mA
DC Current for 5V Pin	800 mA
Flash Memory	512 KB all available for the user applications
SRAM	96 KB (two banks: 64KB and 32KB)
Clock Speed	84 MHz
Length	101.52 mm
Width	53.3 mm
Weight	36 g

Table 4.2: Specifications of Arduino Due

4.1.2 Zigbee Network

Zigbee and 802.15.4 both provides infrastructure for wireless sensor networks. Zigbee defines application and network layer whereas 802.15.4 provides data link and physical layer specifications. Zigbee always makes a Personal Area Network (PAN). Each device can only join one PAN network maintained by the coordina-

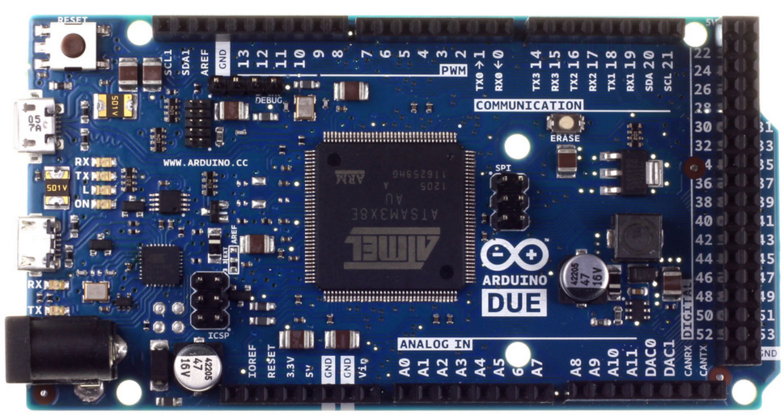


Figure 4.1: Arduino Due front

tor device. Zigbee defines three device types: coordinator, router and end devices.

- **Coordinator** is the most powerful device in the Zigbee network. It selects a channel and PAN id to start the network, have capability to allow routers and end devices to join the network, assists in routing of data and can never go to sleep mode. Each PAN could have only one coordinator.
- **Router** s less powerful devices as compared to coordinator, can allow routers and end devices to join the network. As name suggest, it is used for routing of information. It also can never go to sleep.
- **End device** the least powerful device. It must join PAN network to send or receive information or data. It cannot allow devices to join the network. Cannot directly communicate with other end devices, can go to sleep mode.

Zigbee supports multiple network topologies i.e. star, cluster tree and mesh topology. In star topology coordinator is at the center of network and will relay all information from one node to another. In cluster tree topology, multiple cluster star topologies are connected together to make a cluster tree topology and in mesh topology all devices are connected to each other in their communication range. These topologies can be seen in figure. 4.2

4.1.3 Xbee Shield

Xbee radios are an excellent way to add wireless capability to Arduino. Xbee shield is used as intermediate layer between Arduino and Xbee RF module. The

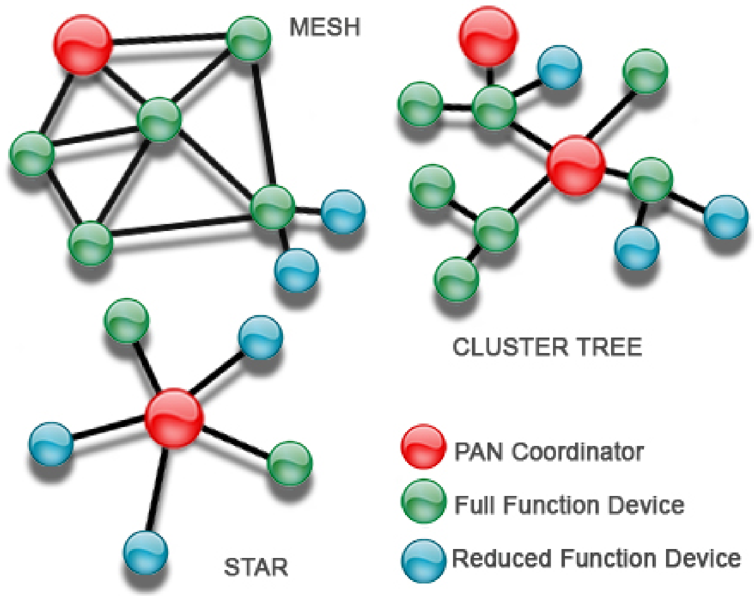


Figure 4.2: Different Zigbee Topologies

shield form-factor mates directly with any dev. board that has an Arduino standard footprint and equips it with wireless communication capabilities using the XBee RF module [27].

The serial pins (DIN and DOUT) of the XBee are connected through an SPDT switch, which allows you to select a connection to either the UART pins (D0, D1) or any digital pins on the Arduino (D2 and D3 default). Power is taken from the 5V pin of the Arduino and regulated on-board to 3.3VDC before being supplied to the XBee. The shield also takes care of level shifting on the DIN and DOUT pins of the XBee. Figure 4.3 shows Xbee shield.

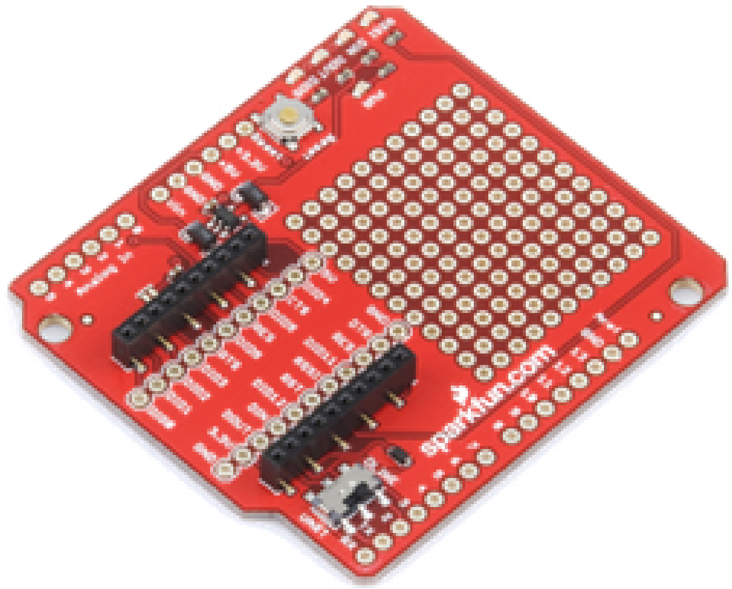


Figure 4.3: Xbee Shield

4.1.4 Xbee Explorer

Xbee explorer is a simple to use, USB to serial base unit for the XBee RF modules. Plug the unit into the XBee Explorer, attach a mini USB cable, and you will have direct access to the serial and programming pins on the XBee unit [64]. Xbee explorer and Xbee explorer with Xbee RF module is shown in figures 4.4 and 4.5. respectively.

A software from Digi International called X-CTU is used for testing and configuring XBee RF modules. Features of this software are as follows:

- Integrated terminal window.
- Easy to use loopback range test.
- Display of Receive Signal Strength Indicator (RSSI).

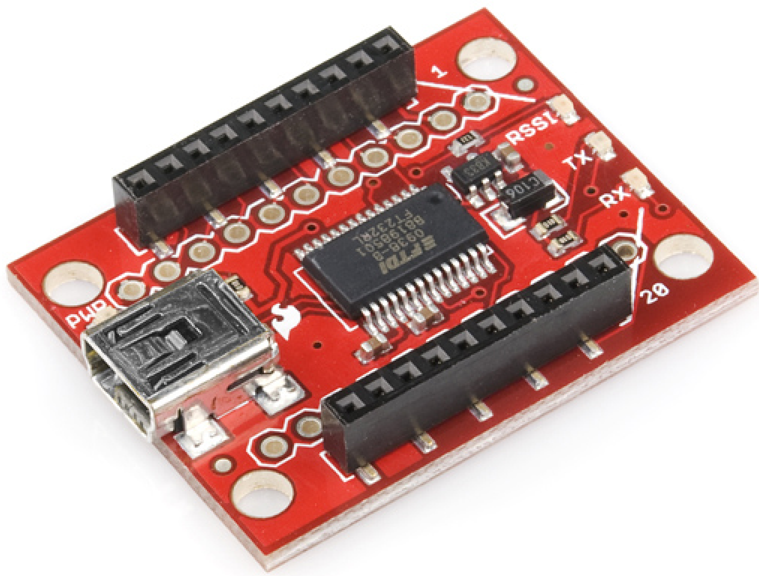


Figure 4.4: Xbee Explorer

Figure 4.5: Xbee Explorer with RF module

- Display both ASCII and hexadecimal characters in terminal window.
- Compose test packets in either ASCII or hexadecimal for transmitting in terminal interface.
- Save and retrieve commonly used module configurations (profiles).
- Automatically detect module type.
- Restore factory default parameters.
- Display help about each of the radio parameters.
- Program radio profiles in a production environment using command line interface.

The software is easy to use and allows to test the radio modems in the actual envi- ronment with just a computer and the items included with the radio modems [ref]. X-CTU can be seen in figure 4.6.

4.1.5 MPU6050 Accelerometer + Gyro Sensor

The InvenSense MPU-6050 sensor contains a MEMS accelerometer and a MEMS gyro in a single chip. It is very accurate, as it contains 16-bits analog to digital conversion hardware for each channel. Therefor it captures the x, y, and z channel at the same time. The sensor uses the I2C-bus to interface with the Arduino.

The MPU-6050 is not expensive, especially given the fact that it combines both an accelerometer and a gyro. Figure 4.7 shows how MPU6050 accelerometer + gyro looks like.

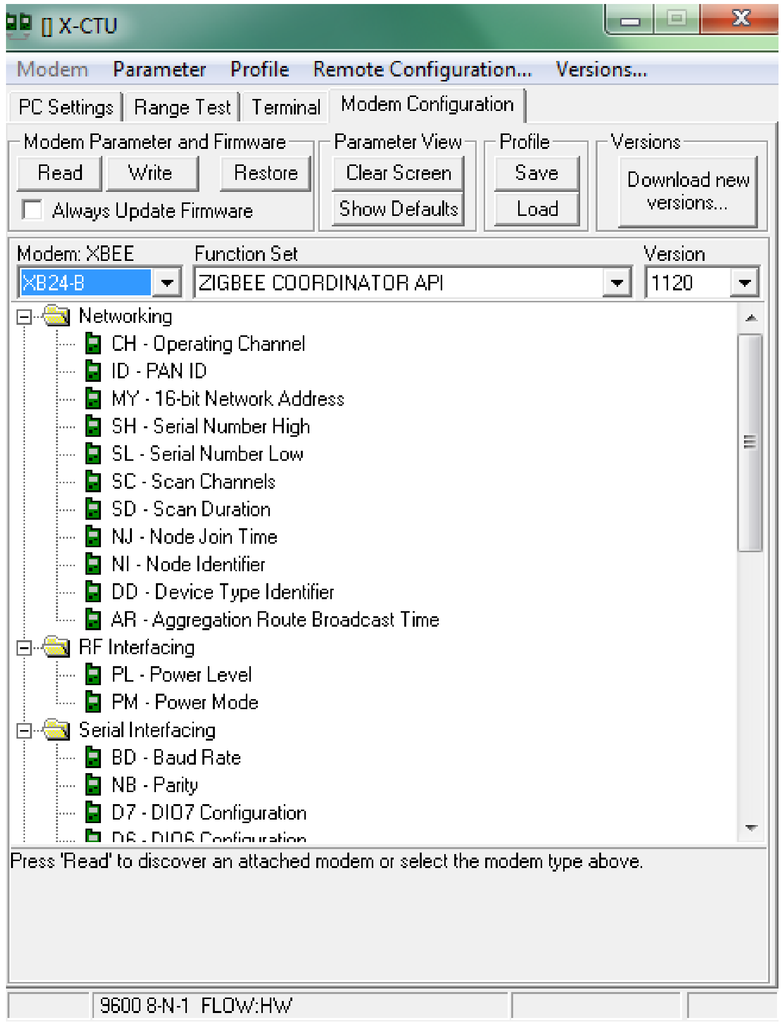


Figure 4.6: XCTU software

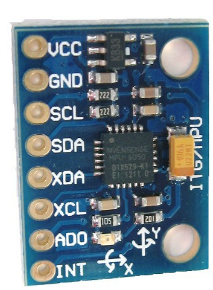


Figure 4.7: MPU6050 accelerometer + Gyro

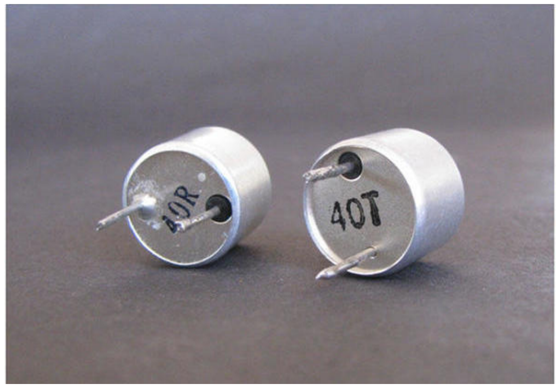


Figure 4.8: Ultrasonic Transmitter and Receiver

This MPU6050 has been interfaced with Target node in TDOA technique to control the traffic in the network. In the next chapter we have discussed how this all has been done.

4.1.6 Ultrasonic Transmitter

To generate software controlled ultrasonic pulse of 125us, 40 kHz piezo-electric open air ultrasonic transmitter is driven at 12 V [ref]. Our system uses voltage multiplier to generate the required 12 V from the available 3 V input voltage. Figure 4.8 shows ultrasonic transmitter and receiver. They both work on kHz frequency.

4.1.7 Ultrasonic Receiver

Open-air type 40 kHz piezo-electric ultrasonic receiver is used to receive transmitted ultrasonic by the target node. The Output of this sensor is sent to the three stage amplifier having voltage gain between 70 to 78 dB [ref]. This amplified voltage is compared with the preset threshold voltage to find ultrasound detection.

4.2 Hardware used for RSSI based localization system implementation

For for implementation of RSSI based localization system we have only used Telosb motes by crossbow. Most of the work in this technique is done on programming these motes. No additional hardware has been used for the implementation of this technique.

4.2.1 Telosb

Crossbow's TelosB mote (TPR2400) is an open source platform designed to enable cutting-edge experimentation for the research community. The TPR2400 bundles all the essentials for lab studies into a single platform including: USB programming capabil- ity, an IEEE 802.15.4 radio with inte- grated antenna, a low-power MCU with extended memory and an opti- onal sensor suite (TPR2420). TPR2400 offers many features, including:

- EEE 802.15.4/ZigBee compliant RF transceiver
- 2.4 to 2.4835 GHz, a globally compatible ISM band
- 250 kbps data rate
- Integrated onboard antenna

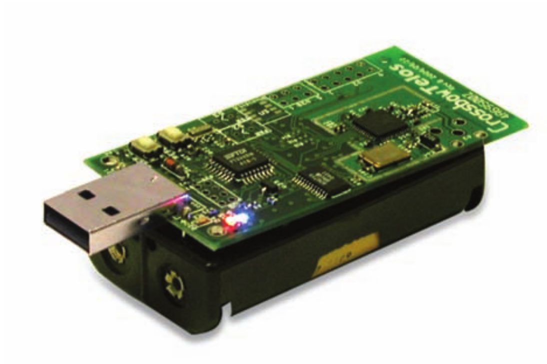


Figure 4.9: Telosb Mote

- 8 MHz TI MSP430 microcontroller with 10kB RAM
- Low current consumption
- 1MB external flash for data logging
- Programming and data collection via USB
- Optional sensor suite including integrated light, temperature and humidity sensor (TPR2420)
- Runs TinyOS 1.1.10 or higher

TPR2400 is powered by two AA batteries. If the TPR2400 is plugged into the USB port for programming or communication, power is provided from the host computer. If the TPR2400 is always attached to the USB port no battery pack is needed.

TPR2400 provides users with the capability to interface with additional devices. The two expansion conne- ctors and onboard jumpers may be configured to control analog sensors, digital peripherals and LCD displays.

CHAPTER 5

IMPLEMENTATION OF LOCALIZATION TECHNIQUES

In this chapter we describe implementation details of our localization systems. We implemented all three systems on our own and this work has not been done before on hardware.

5.1 Implementation of RSSI based localization system

As we mentioned in the last chapter, RSSI based localization technique was implemented on Telosb motes by crossbow and no additional hardware was used. Three methods have been tested on RSSI based localization system:

- Self calibration reference nodes
- Fingerprinting based Localization System

5.1.1 Self Calibrating Algorithm

Distance estimation based on RSSI are not efficient due to environmental changes and for a given distance RSSI won't be the same if it is measured multiple times. Equation 5.2 shows the relationship between RSSI and distance

$$RSSI(d) = RSSI(d_o) + 10nlog(\frac{d}{d_o}) \tag{5.1}$$

$$d = 10^{\frac{RSSI(d)-RSSI(d_o)}{10n}} d_o \tag{5.2}$$

In above equations $RSSI(d)$ is the RSSI at distance d away from the target node, $RSSI(d_o)$ is the reference RSSI at reference distance d_o and n is the path loss exponent.

In this method, $RSSI(d_o)$ is measured periodically. Figure 5.1 shows network topology used for RSSI localization system. There are three types of nodes in the network: 1. Target Node, 2. Reference Nodes, and 3. Sink Node.

- **Target Node** is based on Telosb. This node is programmed in TinyOS and the code is very simple to implement. This node has two interfaces.

  1. It periodically broadcasts a radio packet which contains the target ID.

  2. It receives sleep messages from reference nodes.

- **Reference Node** is based on Telosb. This node has four interfaces:

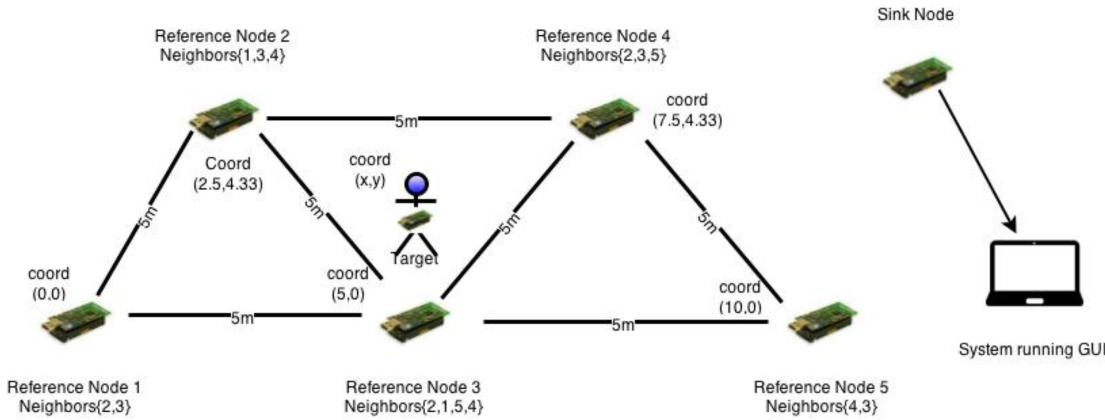


Figure 5.1: Telosb Based RSSI Localization System

1. sends calibration packets.

2. receives calibration packets from neighbor reference nodes.

3. sends sleep messages to the target node.

4. sends reports to the sink node

- **Sink Node** is based on Telosb as well. This node has two interfaces.

   1. It receives radio packets from reference nodes.

   2. It forwards received radio packets to the Java Application using serial interface.

Reference Nodes Calibration

In equation 5.2 we can see that $RSSI(d_o)$ is the reference RSSI at reference distance d_o. In self calibration algorithm, this parameter is calculated on the run time. To calculate this parameter on the run time we assume that each reference node has the information of its neighbors.

All the reference nodes periodically send calibration packets to their neighbors.

As we can see in fig 5.1 that each reference node has the information of its neighbors and they only send calibration packets to their neighbors only. It can also be seen that the distance between each reference node is 5m and each of three reference nodes make a triangle.

To avoid collision between target node packets and calibration packets, each reference node broadcasts sleep messages. When the target node receives this sleep message, it halts the transmission and goes to sleep mode for 5 seconds.

Right after sending sleep messages, all the reference nodes send calibration packets to their neighbors. The packet contains reference node ID only. When a node receives calibration packet, it first extracts the node id checks this ID in its neighbor table. If the node ID is in its table, it measures the RSSI of this calibration packet and stores it in its local database, otherwise it discards the packet.

When a reference node has received calibration packets from all of its neighbors, it extracts all the RSSI values from its database and calculate the average. This average is nothing but reference RSSI at d_o. And d_o in our case is 5m which is fixed distance. All other nodes do the same process upon receiving calibration packets. For instance, in fig 5.1 when a node 3 receives calibration packets, it first check whether it is coming from its neighbors (neighbor table = 1,2,5,4). After verifying the packet, this node saves RSSI from each neighbor and after that it calculates the average of saved RSSI values. This is shown in eq 5.3, where a is the index of an array and k is the size of the neighbor table. For example $RSSI_1$ corresponds to the first RSSI value in the array.

$$RSSI(d_o) = \frac{1}{k}\sum_{a=1}^{k} RSSI_a \tag{5.3}$$

Distance Measurements

To locate the target, we require distances from at least three reference nodes. After obtaining three distances we use trilateration to compute the coordinates of the target node. Hence, the accuracy of the system is mainly depended on how accurately we obtained these distances.

In equation 5.2 it is apparent that we require three unknowns to calculate the distance to the target node: $RSSI(d), RSSI(d_o)$ and n. Since n mainly dependent on the indoor environment and it varies from environment to environment. We deployed Telosb motes in a corridor and there were only a few obstacles, thus we chose $n = 2.5$. Thus we are now left with two unknowns; $RSSI(d)$ and $RSSI(d_o)$.

$RSSI(d)$ is the RSSI at reference node away from target node. When the target node broadcasts a radio packet, reference nodes receive this packet and extract the RSSI from the packet meta data. When it receives packet from the target node, it creates a new packet which contains $RSSI(d)$ from the target node, $RSSI(d_o)$ which is calculated through calibrating reference nodes, the target ID, and its own ID. It then unicast this message to the sink node. All the distance estimation and calculations are done at sink node except for the calibration part.

When a sink node receives packets from reference nodes, it immediately forwards them to the Java application through serial interface. All the calculations are done at java application.

As we have already mentioned, RSSI is the most disturbing parameter and even for the same distance it varies too much. we use moving average to smooth both $RSSI(d)$ and $RSSI(d_o)$ in the equation 5.1 to calculate the distance. Since the speed of target is very slow, we take moving average of 100 samples.

Target Node Coordinates Calculation

We use Linear Least Squares technique to calculate the coordinates of target node. In LS we require at least three reference node to computer the coordinates. As it can be seen in fig 5.1 that there are six reference nodes and each three consecutive reference nodes make a triangle. There are a maximum of three triangles can be formed by these six reference nodes. The problem occurs in deciding which set of three nodes to use for LS.

Now let's consider the case when the target node is in the middle triangle which means the RSSI value from this target at reference nodes 2,3, and 4 much be larger than at triangles formed by 1,2,3 and 3,4,5.

In at application side when a the Java application receives all the information from Sink Node it groups the data according to reference nodes ID's and their corresponding RSSI values. In fig 5.1 we can see there are 3 possible triangles, so java application forms three abstract triangles: $triangle_1$(1,2,3, $RSSI_1$,$RSSI_2$,$RSSI_3$), $triangle_2$(2,3,4, $RSSI_2$,$RSSI_3$,$RSSI_4$),and $triangle_3$(3,4,5, $RSSI_3$,$RSSI_4$,$RSSI_5$).

After grouping the data now the application decides in which triangle our target lies. This decision is made on the run time on the basis on the equation.

$$\Delta = \max(\sum valuesoftriangle_1 RSSI, \sum valuesoftriangle_2 RSSI, \sum valuesoftriangle_3 RSSI) \tag{5.4}$$

Based on Δ determined from equation 5.4 application decides which set of three reference nodes to use for LS calculations.

Java Application

Java application has been developed in Eclipse. The application is connected with the sink node through serial connection and receives data from it. All the distance estimation and coordinates calculation is done at the application and the results of all the calculations are shown in Graphical User Interface.

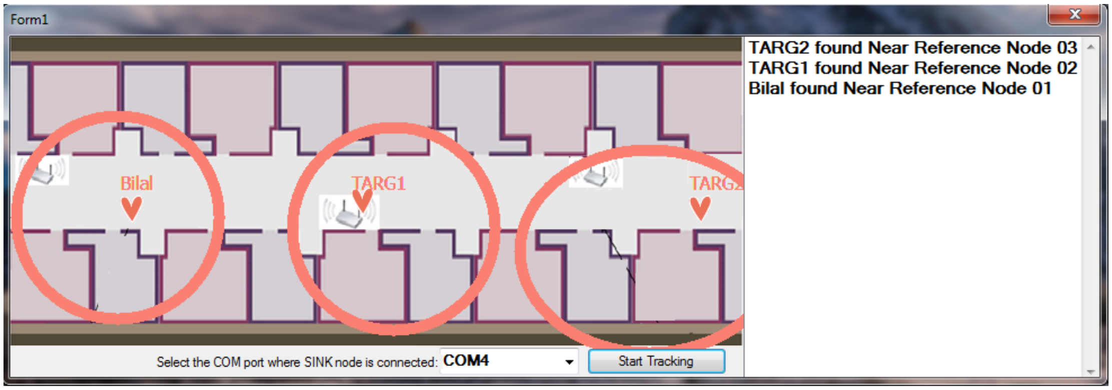


Figure 5.2: GUI for the Self calibration RSSI based Localization sysem

Figure 5.2 shows the Graphical User Interface (GUI) for the Self calibration RSSI based localization algorithm. This GUI is in offline mode, meaning that there is no device in the network. The GUI consists of three panels: Top, middle and lower panel. The top panel consists of a text box followed by a connect button. In the text box we type the serial address of the Sink Node connecting to our laptop/PC. The address is in the form of linux hardware addressing because we developed this GUI on linux platform. After typing the Sink node address which is connected through USB cable,

The middle panel consists of 5 boxes, each box corresponds to the reference node. When the application receives RSSI values from the network, it creates line graph of the received RSSI and calculate the variance of RSSI from each reference node. The lower panel consists of the map of the topology. As it can be seen in figure 5.1, the reference in this panel are placed according to their placement in real environment. The target ID is 0 in our case and to initialize the system we assign it the coordinates of (4,0). As application starts receiving data from the network,

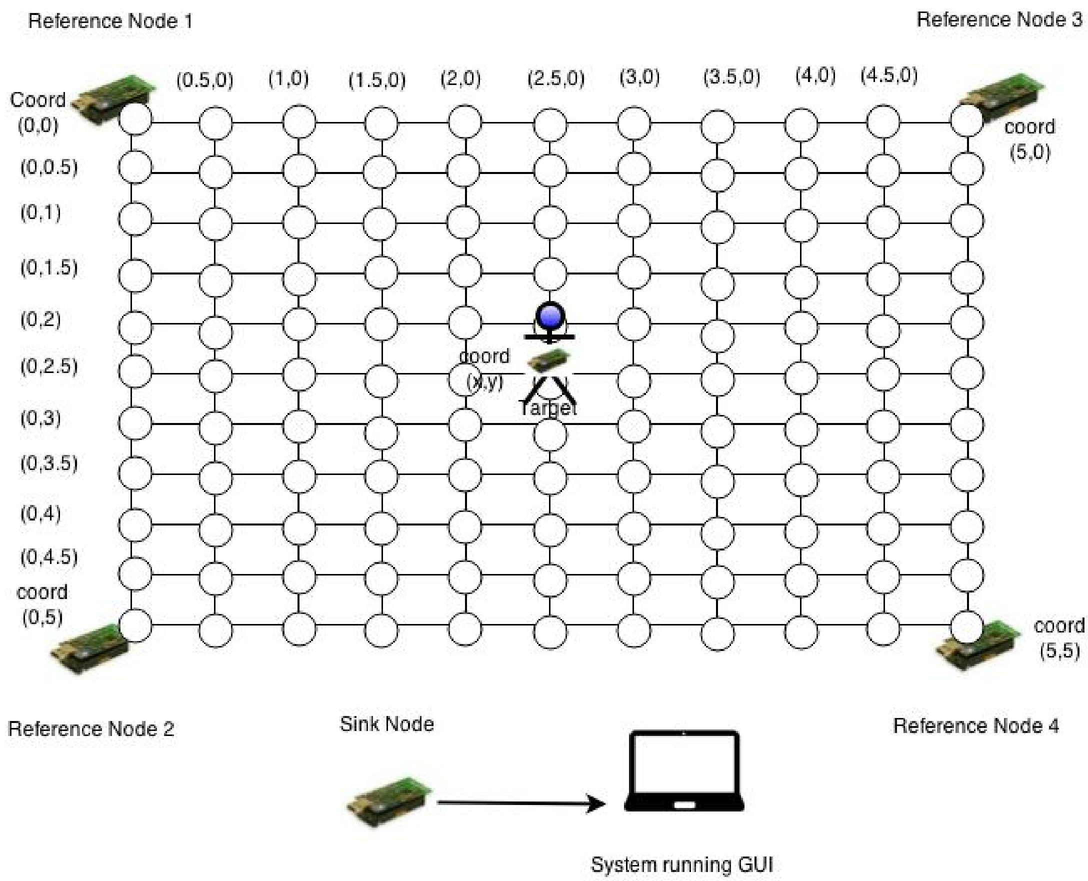


Figure 5.3: Fingerprinting based Localization System

it keeps updating target node coordinates.

5.2 Implementation of Fingerprinting based Localization System

In fingerprinting based localization system, all the calculations and readings are collected before the deployment of the system. The topology is always grid in fingerprinting and a database is maintained which stores all the a priori measurements.

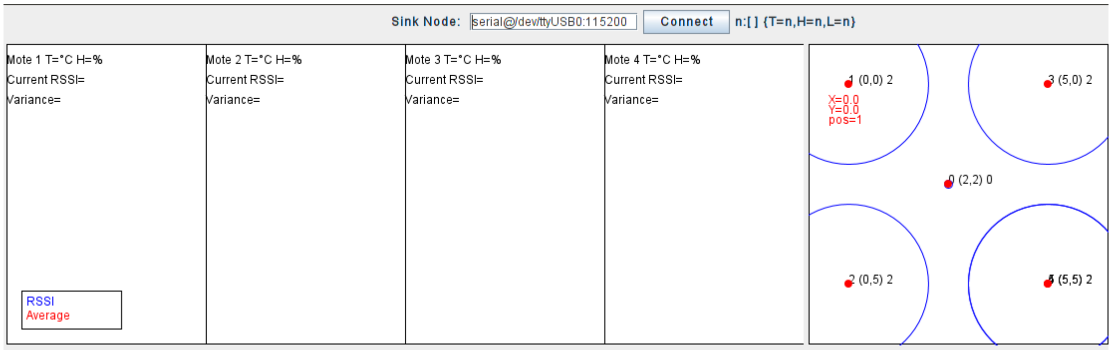


Figure 5.4: GUI for fingerprinting.

Figure 5.3 shows the grid topology for fingerprinting based localization system. In this system the grid size is 5m by 5m which is further divided into very small rectangles. The circles shown in fig 5.3 the corners of each rectangle show the points where measurements were taken a priori.

In the measurements phase, 10 RSSI values from each reference node is collected at each corner of the rectangle an their average is calculated. The average is then saved in a database along with the coordinates of the corner of a rectangle and the physical distance.

After deploying the system, when the java application receives data from the network, it first takes the average of the received RSSI values from the reference nodes. It then maps this average RSSI to RSSI values saved in the database. If it matches the RSSI value in database, it shows extract the coordinates from DB and shows them in the Graphical User Interface.

Figure 5.4 shows the GUI for fingerprinting based localization system. GUI is almost the same as the GUI of 5.1.1 except there is now one less reference node and the topology is now grid. This system has also been implemented using Telosb motes there were 6 motes used: 4 motes as reference nodes, 1 as the target, and 1 as the SINK node.

It can be seen that the node in with ID 0 is our target node and has coordinates of (2,2). This is because when we start the application it automatically assigns it these coordinates and later when application starts receiving data from the network, it updates these coordinates on the run time.

5.3 Implementation of the Time Difference of Arrival (TDOA)

This technique takes advantage of the distinct velocities of two signals transmitted from one node and calculates the time difference of arrival between these signals when transmitted at the same time. If V_{RF} and V_{US} are the velocities of the two signals: Radio frequency signal and Ultrasonic signal respectively used and Δt is the time difference of arrival for the two signals at the second reference point, then the distance d between the two reference points is calculated as

$$d = \frac{\Delta t}{(\frac{1}{V_{US}} - \frac{1}{V_{RF}})} \tag{5.5}$$

If $V_{RF} >> V_{US}$ then equation above becomes

$$d = \Delta t V_{US} \tag{5.6}$$

TDOA system has been designed using Arduino Due microcontrollers. Please refer to Hardware chapter for the hardware details.

Arduino Due Microcontrollers do not have built in ultrasonic receiver and transmitter, thus we have separately interfaced them with Arduino. Ultrasonic transducers work on 40kHz frequency and the operating voltage of these transducers is around 12-20V.

Arduino Due Microcontrollers work on just 3-5V hence we have designed a voltage multiplier to that amplifies Arduino Due signal to 12V.

In the implementation we have used 5 Arduino Due Microcontrollers: 1 target node, 3 reference nodes and 1 sink node.

5.3.1 Target Node

Target node is equipped with a MPU6050 accelerometer, and Ultrasonic transmitter, Xbee transceiver and an Arduino Due microcontroller. An MPU6050 accelerometer is used to detect if the target is still or is in motion. Target node only transmits data when the node is moving.

Ultrasonic Transmitter is connected to the voltage multiplier that which is further

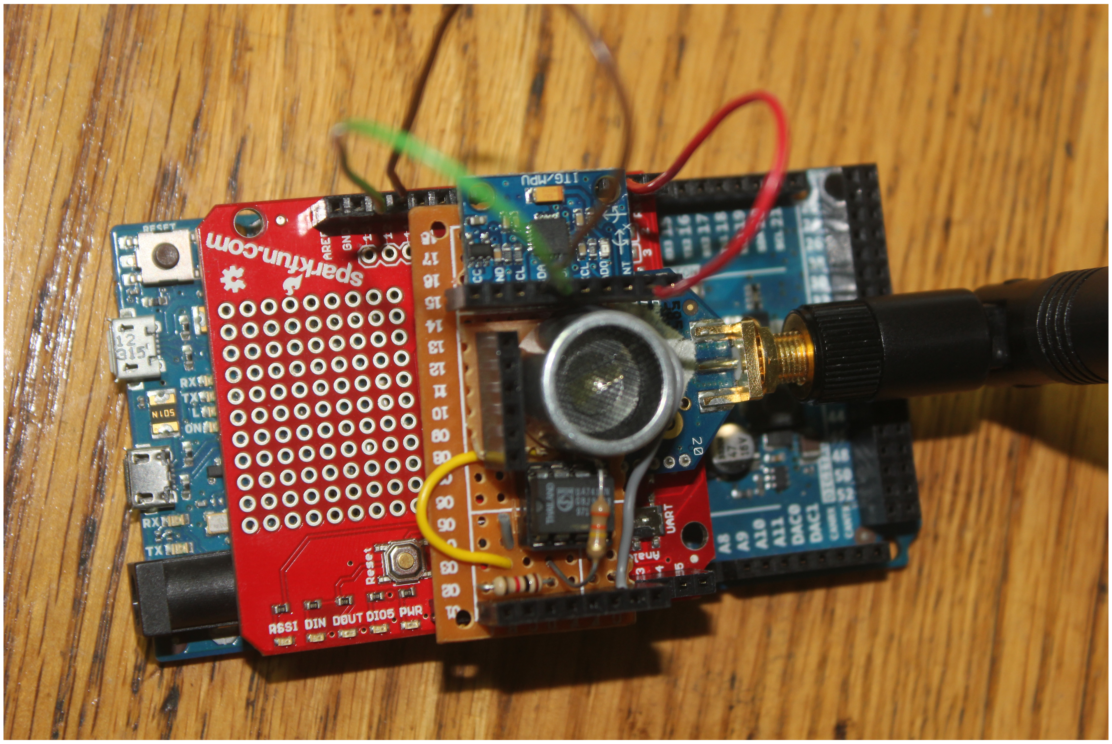


Figure 5.5: Target Node equipped with US transmitter, MPU6050 accelerometer, and XBee transceiver

connected to Arduino PWM pin which generates a square wave of 40kHz and an amplitude of 3Volts. Hence, ultrasonic transmitter now transmits a perfectly square wave of 40kHz and has an amplitude of 12V.

Target node is also equipped with XBee RF transceiver that transmits RF signals whose velocity is 2/3rd of the speed of light. Fig 5.5 shows the target node. Target node keeps communicating with MPU6050 accelerometer to detect if there is any movement, if there is a contineous movement for 3 seconds, it sends RF and US signals back to back and again communicates with MPU6050 accelerometer to detect any further movements. RF packet contains the target ID only.

5.3.2 Reference Node

We have used three reference Nodes and each of them is equipped with an Ultrasonic receiver, XBee RF transceiver and an Arduino Due microcontroller. When a target node transmits two signals: RF and US back it is apparent that RF will arrive at reference node earlier than US. When the receiver receives this RF signal signal, it saves its arrival time t_1 and starts the timer and waits for US tone detection. When a US signal arrives it saves its arrival time in variable t_2 and then takes the difference of $t_2 - t_1$.

It then creates a new RF packet that contains Target ID, reference nod ID and the difference of arrival of RF and US signals. Reference node then sends this packet to the Sink node at unicast address of the sink node.

The problem with ultrasonic receiver is that it cannot be used standalone with a microcontroller because the signal that it receives has an amplitude in Milli volts, thus we have designed a three stage amplifier followed by a comparator. The amplifier circuit diagram is shown in fig 5.7. The output of the ultrasonic receiver is connected to a three stage amplifier which is followed by LM386. The output of LM386 is then goes to the Analog input of the Arduino Microcontroller. We read the status of this analog pin to detect ultrasonic tone. Fig 5.6 shows the reference node used in TDOA system.

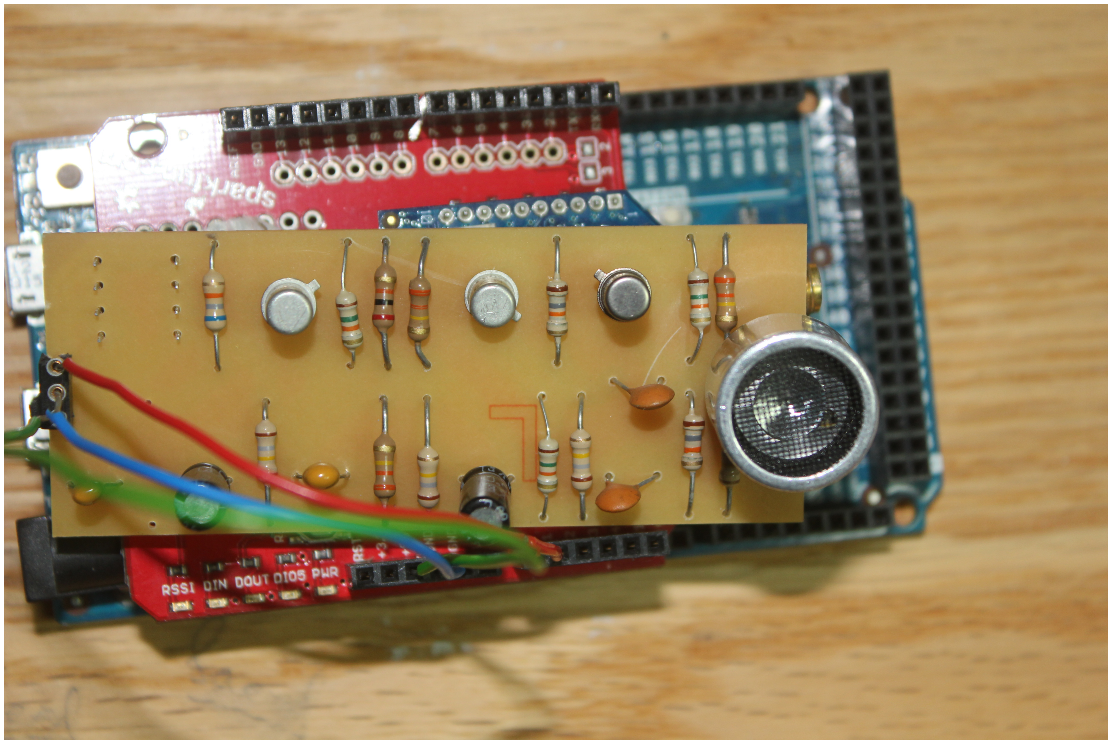


Figure 5.6: Reference Node equipped with US receiver amplifier and XBee RF transceiver

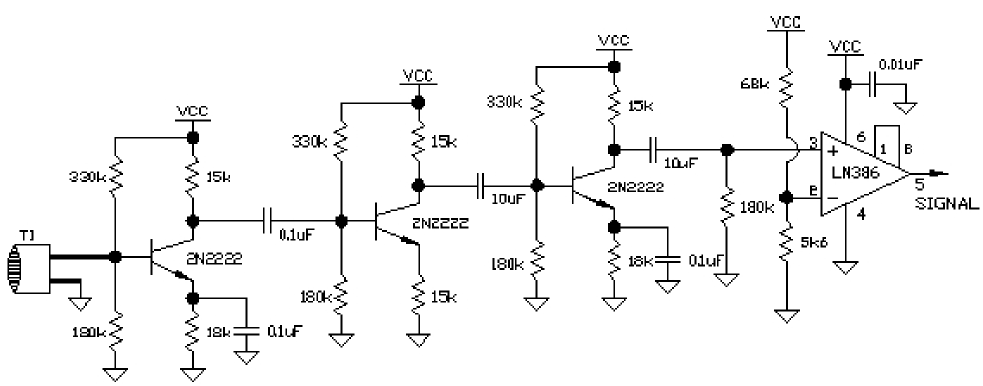


Figure 5.7: Ultrasonic Receiver Amplifier

5.3.3 Movement Detection

An accelerometer works on the principle of piezo electric effect. Here, imagine a cuboidal box, having a small ball inside it, like in the picture above. The walls of this box are made with piezo electric crystals. Whenever you tilt the box, the ball is forced to move in the direction of the inclination, due to gravity. The wall with which the ball collides, creates tiny piezo electric currents. There are totally, three pairs of opposite walls in a cuboid. Each pair corresponds to an axis in 3D space: X, Y and Z axes. Depending on the current produced from the piezo electric walls, we can determine the direction of inclination and its magnitude. The MPU 6050 communicates with the Arduino through the I2C protocol. The MPU 6050 is connected to Arduino as shown in the following diagram. Here, if your MPU 6050 module has a 5V pin, then you can connect it to your arduino's 5V pin. Else, we will have to connect it to the 3.3V pin. Next, the GND of the arduino is connected to the GND of the MPU 6050.

The program we will be running here, also takes advantage of the arduino's interrupt pin. Therefore, connect arduino's digital pin 2 (interrupt pin 0) to the pin labeled as INT on the MPU 6050. Next, we need to set up the I2C lines. For this connect the pin labeled as SDA on the MPU 6050 to the arduino's analog pin 4 (SDA). And the pin labeled as SCL on the MPU 6050 to the arduino's analog pin 5 (SCL). And that's it, we have finished wiring up the Arduino MPU 6050.

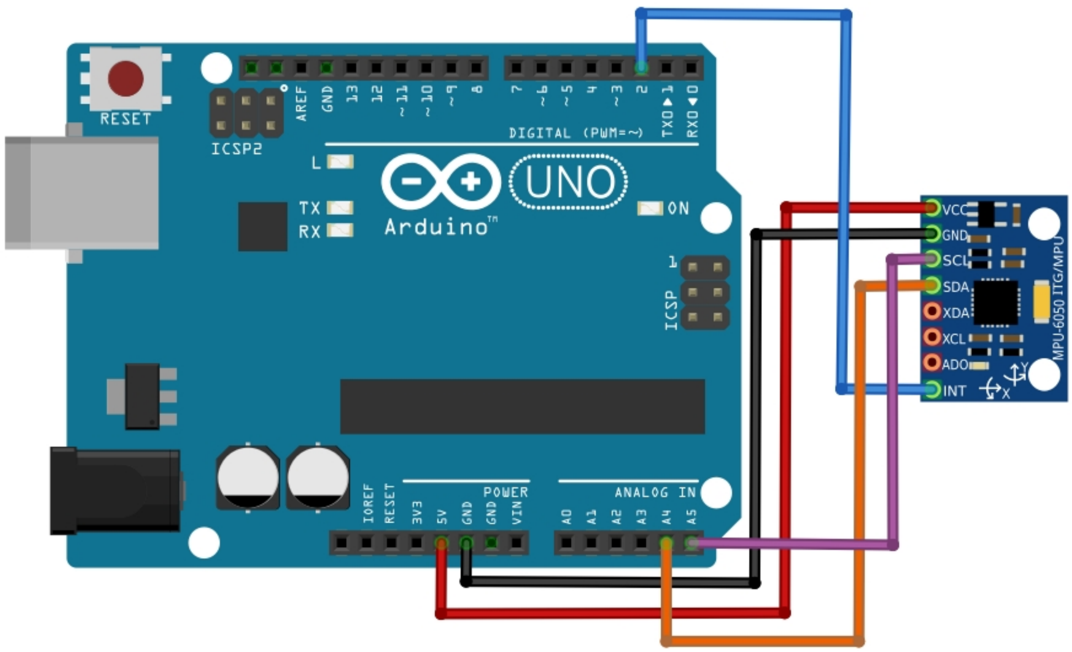


Figure 5.8: MPU6050 interfacing with Arduino microcontroller

5.3.4 Graphical User Interface

The GUI is almost the same as the GUI for RSSI based localization system, however the only difference is TDOA based localization system is based on time, sink node receives Δt from reference nodes and calculate the distance based on Δt.

It can be seen in fig **??** that there were only 4 reference nodes used and simple trilateration technique is used to measure the the coordinates of the target node.

Just like RSSI based localization system, in TDOA sink node is connected to the java application through serial interface and sends updated readings to the application. In case of TDOA all the nodes in this system are based on Arduino instead of Telosb motes.

CHAPTER 6

RESULTS AND DISCUSSION

In this chapter we discuss the results of three localization systems that we implemented: Fingerprinting, Self calibration Localization, and Time Difference of Arrival (TDOA) based localization system.

We first talk about the head count application we built using Arduino and Java. Headcount application only detects

6.1 Implementation of Headcount Application

The implementation consists of two architectures – hardware and software. Hardware architecture consists of Arduino Microcontrollers and Series 2 Xbee modules. 6 Arduino and 6 XBEE modules have been used to develop the prototype.

Xbee modules have been configured using Xbee Explorer and XCTU software. PAN ID and XBEE module type, API mode were two parameter that have been configured, while other parameters remain the same. The configuration for each XBEE module is shown in table 6.1. After successful configuration of XBEE

Device	PAN ID	Scan Channels	API Enable	Function Set
Target	789	80	2	Zigbee Router
Reference Node	789	80	2	Zigbee Router
SINK node	789	80	2	Zigbee Coordinator

Table 6.1: Xbee Modules Configuration

modules, Arduino Microcontrollers have been programmed. Each target node consists of an Arduino Microcontroller and XBEE module. Each target creates a new packet every 1.7 seconds which contains its ID for example and broadcasts it.

Broadcast messages from target node are received by fix reference nodes. Each reference node also consists of an Arduino Microcontroller and XBEE module. Reference nodes are fixed and have been programmed in such a way that they receive packets from target nodes, extract target ID from incoming packets, creates a new packet and append their IDs along with the target ID and send it to the SINK. Target ID and reference node IDs are separated by a ';' split character.

Fixed reference nodes send their packets to the SINK node using unicast. SINK node is connected to a dedicated computer where C# application is running, when SINK node receives a packet from reference node it extracts the data which is in this format (targetID;ReferenceNodeID) and separates the target and reference node id. It then shows this information on the map in C# application.

6.1.1 Discussion of Results

After successful implementation of the prototype described in above section, we tested it on a network as shown in fig 6.1. It can be seen that the prototype consists of three fixed reference nodes, three target nodes and one coordinator. It

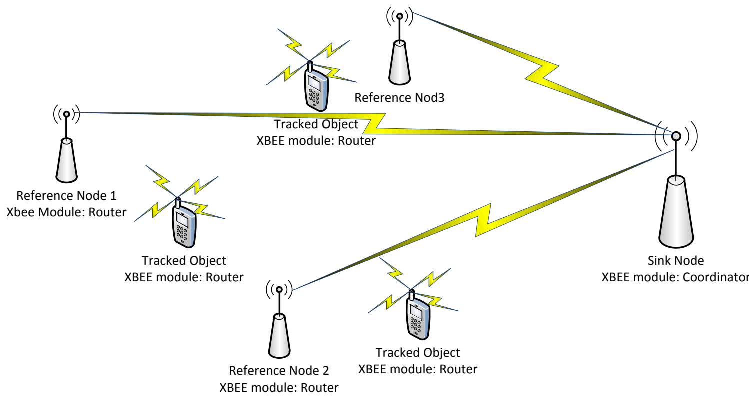


Figure 6.1: Sample Network

can be observed that target nodes broadcast their ID while reference nodes send their information to the SINK node using Unicast. When SINK node receieves information from reference nodes, it extracts it and pass this information to C# application through communication layer. Communication layer is the interface between C# application and the SINK node.

Application show the location of the tracked target on the map as it is shown in fig. 4 It is already mentioned that in this prototype 3 ref nodes, 3 targets, and

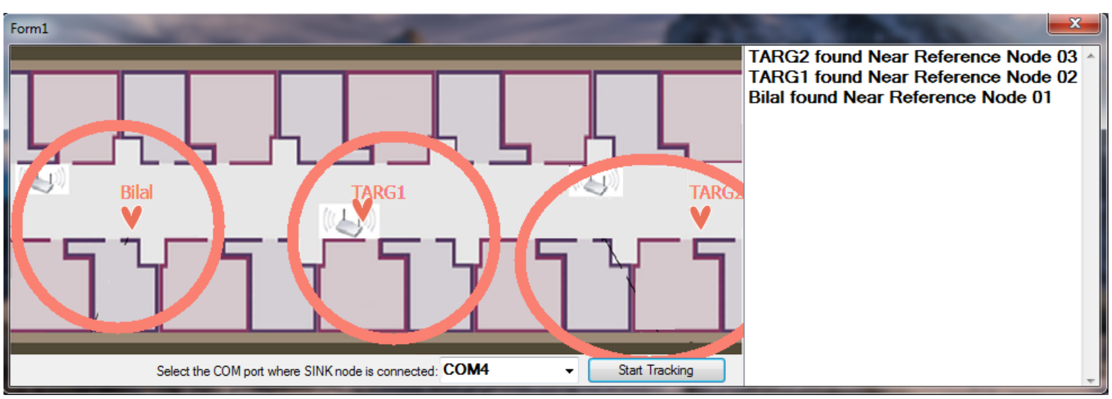


Figure 6.2: Application

1 sink node have been used. The red circles on the map show the coverage area

of each reference node. It can be seen that target Bilal is detected under the coverage area of reference node 1, therefore it is showing up in its vicinity.

6.1.2 Conclusion

In headcount we developed a prototype for tracking and localizing objects using Arduino and Xbee RF transceivers. Headcount only tells us the presence of object. Reference nodes receive messages from target and forward them to the SINK. SINK node first checks which reference node is sending the message and extract the target ID. Upon receiving the message, SINK pass this information to the application where target and reference nodes are shown on the GUI.

6.2 Results of Fingerprinting Based Localization System

Figure 6.3, 6.4, and 6.5 show the topology used for Fingerprinting based localization. The nodes in the black circles are reference nodes and the node in the red circle is the target node. As it can be seen that in this system the reference nodes are placed on the corners of square. We have already mentioned in the implementation chapter that a database is created in Microsoft Access that which is connected to our Java application. As Java application receives data from the

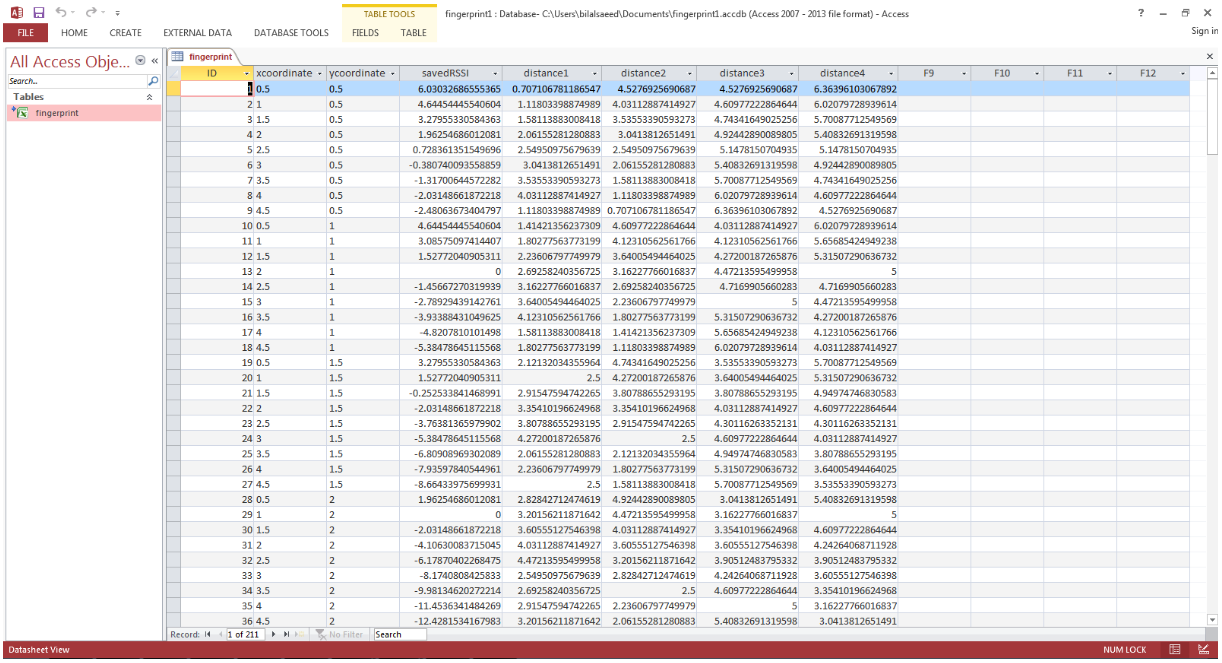
Figure 6.3: Fingerprinting based localization system

Figure 6.4: Fingerprinting based localization system

Figure 6.5: Fingerprinting based localization system

network, it maps the received RSSI values to the values stored in database and extract the corresponding coordinates and distances from the database and plot them on the GUI.

Fig **??** shows the MS access database which is connected to our java application. The first column of the database is the ID number of the row, 2nd column shows the x-coordinate of the the grid, 2nd column shows the the y-coordinate of the grid , 3rd column contains RSSI values that were collected in measurement collection phase, columns 5-8 show the distances1-4 respectively. Distances1-4 are the physical distances at any grid point from target node to the reference nodes1-4. When application receives RSSI values from four reference nodes, it calculates the average of these four RSSI values and maps this average to the RSSI values stored in savedRSSI column of the database. When application successfully maps the average RSSI to the value stored in savedRSSI column, it fetches the coordinates



ID	xcoordinate	ycoordinate	savedRSSI	distance1	distance2	distance3	distance4	F9	F10	F11	F12
1	0.5	0.5	6.03032686555365	0.707106781186547	4.5276925690687	4.5276925690687	6.36396103067892				
2	1	0.5	4.64454445540604	1.11803398874989	4.03112887414927	4.60977222864644	6.02079728939614				
3	1.5	0.5	3.27955330584363	1.58113883008418	3.53553390593273	4.74341649025256	5.70087712549569				
4	2	0.5	1.96254686012081	2.06155281280883	3.0413812651491	4.92442890089805	5.40832691319598				
5	2.5	0.5	0.728361351549696	2.54950975679639	2.54950975679639	5.1478150704935	5.1478150704935				
6	3	0.5	-0.380740093558859	3.0413812651491	2.06155281280883	5.40832691319598	4.92442890089805				
7	3.5	0.5	-1.31700644572282	3.53553390593273	1.58113883008418	5.70087712549569	4.74341649025256				
8	4	0.5	-2.03148661872218	4.03112887414927	1.11803398874989	6.02079728939614	4.60977222864644				
9	4.5	0.5	-2.48063673404797	1.11803398874989	0.707106781186547	6.36396103067892	4.5276925690687				
10	0.5	1	4.64454445540604	1.41421356237309	4.60977222864644	4.03112887414927	6.02079728939614				
11	1	1	3.08575097414407	1.80277563773199	4.12310562561766	4.12310562561766	5.65685424949238				
12	1.5	1	1.52772040905311	2.23606797749979	3.64005494464025	4.27200187265876	5.31507290636732				
13	2	1	0	2.69258240356725	3.16227766016837	4.47213595499958	5				
14	2.5	1	-1.45667270319939	3.16227766016837	2.69258240356725	4.7169905660283	4.7169905660283				
15	3	1	-2.78929439142761	3.64005494464025	2.23606797749979	5	4.47213595499958				
16	3.5	1	-3.93388431049625	4.12310562561766	1.80277563773199	5.31507290636732	4.27200187265876				
17	4	1	-4.8207810101498	1.58113883008418	1.41421356237309	5.65685424949238	4.12310562561766				
18	4.5	1	-5.38478645115568	1.80277563773199	1.11803398874989	6.02079728939614	4.03112887414927				
19	0.5	1.5	3.27955330584363	2.12132034355964	4.74341649025256	3.53553390593273	5.70087712549569				
20	1	1.5	1.52772040905311	2.5	4.27200187265876	3.64005494464025	5.31507290636732				
21	1.5	1.5	-0.252533841468991	2.91547594742265	3.80788655293195	3.80788655293195	4.94974746830583				
22	2	1.5	-2.03148661872218	3.35410196624968	3.35410196624968	4.03112887414927	4.60977222864644				
23	2.5	1.5	-3.76381365979902	3.80788655293195	2.91547594742265	4.30116263352131	4.30116263352131				
24	3	1.5	-5.38478645115568	4.27200187265876	2.5	4.60977222864644	4.03112887414927				
25	3.5	1.5	-6.80908969302089	2.06155281280883	2.12132034355964	4.94974746830583	3.80788655293195				
26	4	1.5	-7.93597840544961	2.23606797749979	1.80277563773199	5.31507290636732	3.64005494464025				
27	4.5	1.5	-8.66433975699931	2.5	1.58113883008418	5.70087712549569	3.53553390593273				
28	0.5	2	1.96254686012081	2.82842712474619	4.92442890089805	3.0413812651491	5.40832691319598				
29	1	2	0	3.20156211871642	4.47213595499958	3.16227766016837	5				
30	1.5	2	-2.03148661872218	3.60555127546398	4.03112887414927	3.35410196624968	4.60977222864644				
31	2	2	-4.10630083715045	4.03112887414927	3.60555127546398	3.60555127546398	4.24264068711928				
32	2.5	2	-6.17870402268475	4.47213595499958	3.20156211871642	3.90512483795332	3.90512483795332				
33	3	2	-8.1740808425833	2.54950975679639	2.82842712474619	4.24264068711928	3.60555127546398				
34	3.5	2	-9.98134620272214	2.69258240356725	2.5	4.60977222864644	3.35410196624968				
35	4	2	-11.4536341484269	2.91547594742265	2.23606797749979	5	3.16227766016837				
36	4.5	2	-12.4281534167983	3.20156211871642	2.06155281280883	5.40832691319598	3.0413812651491				

Figure 6.6: Fingerprinting Database in MS access

and 4 distances from the database and show this data on the GUI. The fetched distances from database are now again used in trilateration algorithm and new coordinates are calculated.

The accuracy of fingerprinting based localization system is not that accurate as compared with other localization systems we implemented. The main reason is RSSI is the most disturbing parameter thus for the received RSSI value it might not match the RSSI values stored in the database. Hence it is more likely that we get wrong coordinates of the target node.

Figure ?? shows the case when the target node was places outside the grid. In this case when application receives RSSI measurements from reference nodes it calculates the average but node is places outside it would not find any match for the average RSSI in the database thus no coordinates and distance measurements

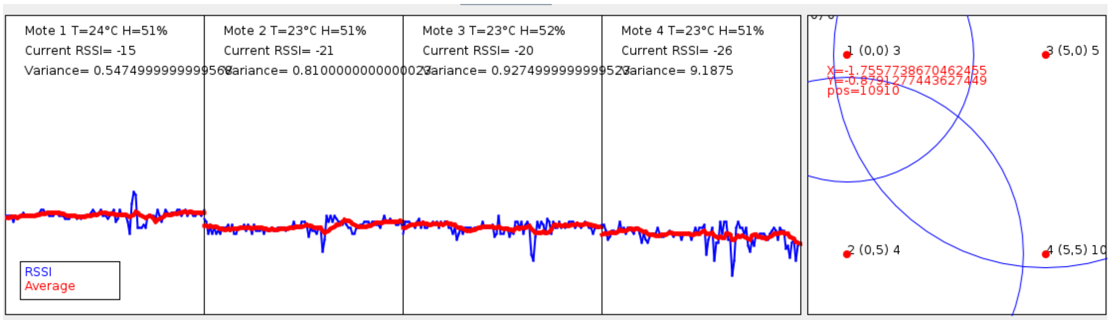


Figure 6.7: Case 1: Target node is outside the grid

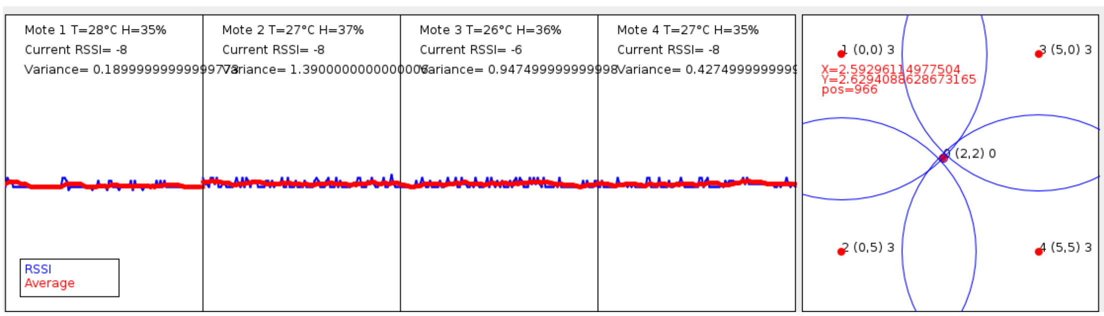


Figure 6.8: Case 2: Target node places in the grid

are being fetched. Since we have four reference nodes in the system, application plots the graphs of RSSI values received from each reference nodes. The red line shows the running average of the received RSSI values and blue line shows the current RSSI values. Reference nodes also send their temperature and humidity readings to the the sink node and application fits them in GUI according to the reference node IDs.

Now let's consider the case when the the target node was placed in the grid. As it can be seen in fig

We placed the target node at (3,1.5) in the grid. Reference nodes receive RSSI values of -8, -8, -6, and -8 at reference nodes 1,2,3, and 4 respectively.

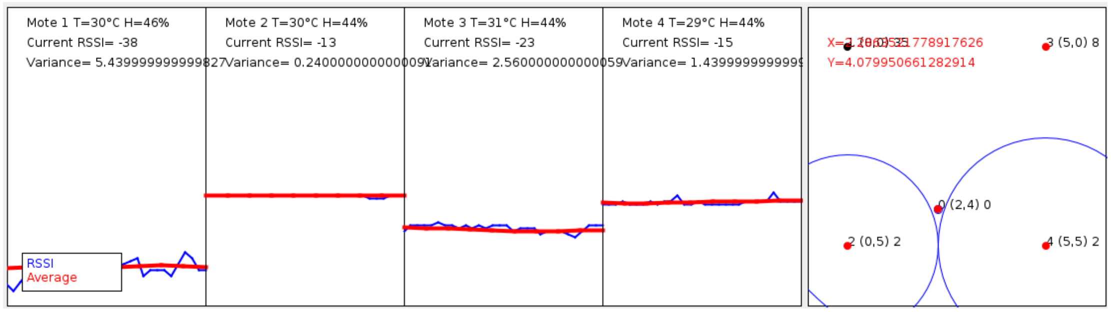


Figure 6.9: Node is placed at some other coordinates

The average RSSI from reference nodes is -5.5, now application matches this average of -5.5 in the database and fetches the corresponding coordinates and distance measurements. Application now uses recalculate the coordinates using these fetched distance measurements. Hence the actual coordinates of the target nodes were (3,1.5), but after applying trilateration based on database distance measurements the target node coordinates turn out to be (2,2).

6.3 Self Calibration Localization System

In this localization system all the reference nodes send their neighbors calibration packets and calculate the reference RSSI at 5m. Unlike fingerprinting based localization system, no a priori measurements are needed and no database is required. All the distance estimation and coordinates calculations are done in run time.

Fig 6.10 shows the placement of nodes in indoor. All the sensors nodes are placed in a way that any three consecutive reference nodes make a triangle. When

Figure 6.10: Deployment of Nodes in indoor

application receives packets from the sink node, it extracts reference node id, target id, reference RSSI and target RSSI. It then create separate lists for each reference node and saves reference node id, target id, reference node rssi and target rssi in the list. In our case application maintains 5 lists because we have 5 reference nodes. To calculate the distance, it uses path loss exponential model as described in chapter 5. Application calculates the running average of reference and target RSSI values of each reference node and uses these averages to measure the distances.

After calculating distances it then check if the distance measurements are valid. In our system any three consecutive reference nodes make a perfect equilateral triangle and each side of all the triangles is 5m. Hence, application first checks if the distance is below 5m. If distance measurements are more than 5m which means our target is not in the triangle boundary.

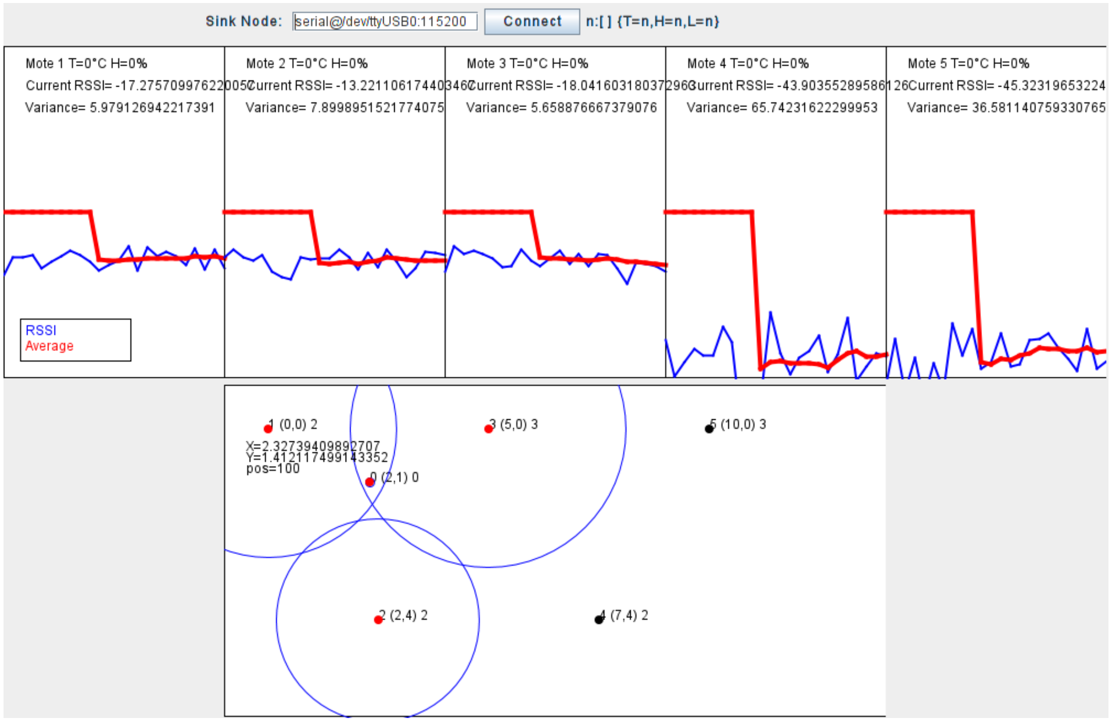


Figure 6.11: Self Calibration Localization System

After verify distance measurements, application now decides which triangle to use. In our case we have three triangles: 1,2, and 3. The decision is made on the basis of the sum of the RSSI values of each triangle. It can be seen that the triangle 1 is used to calculate the coordinates of the target node, because the sum of RSSI values of this triangle is -48.47 which is greater than the sum of RSSI of triangle 2 and triangle 3 (Sum of RSSI of triangle 2 = -75.1645 and Sum of RSSI of triangle 3 = -76.584).

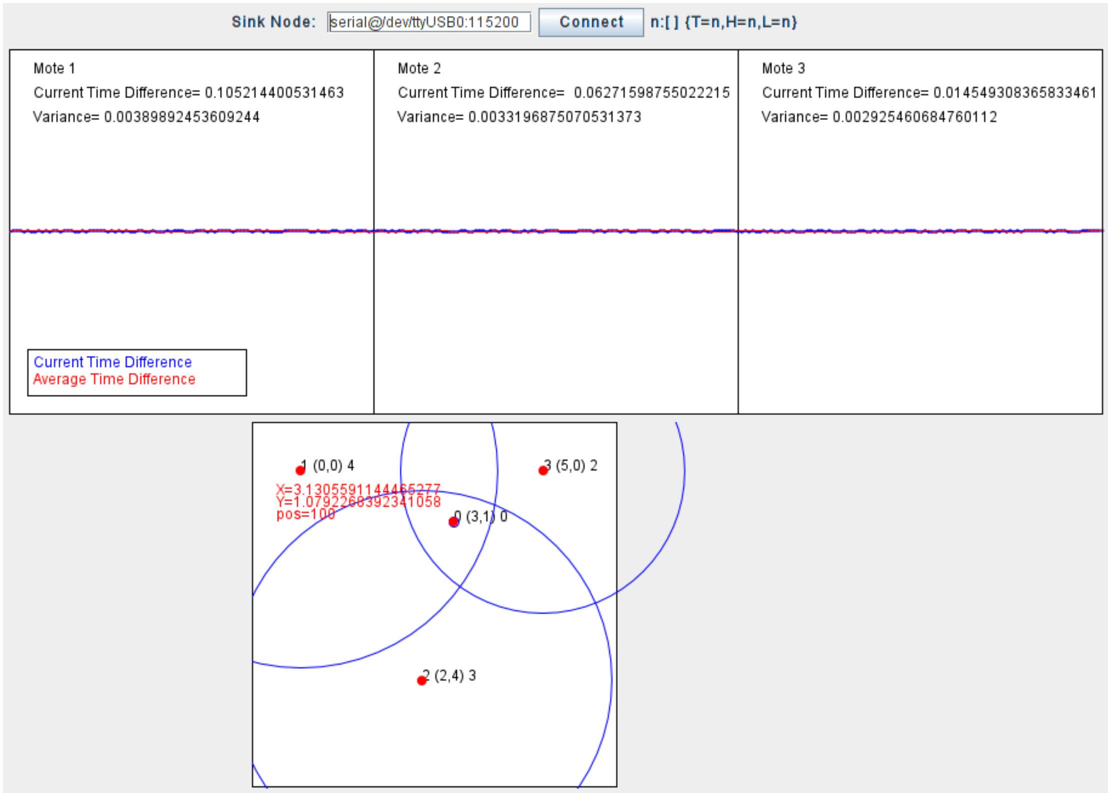


Figure 6.12: Time Difference of Arrival GUI

6.4 Time Difference of Arrival

For TDOA localization System we have used 3 reference nodes, 1 target node and 1 sink node. The GUI is the same as the GUI of previous localization systems. fig [ref] shows the deployment of Sensors in indoor. Unlike previous systems, TDOA uses time measurements to estimate the distance to a target node.

Figure **??** shows the Graphical User Interface for Time Difference of Arrival based localization system. This system by far is the most accurate system as compared to the previous systems. we can see that the variations in the time are very minute. In TDOA based localization system we have used 3 three reference nodes. Reference nodes receives two difference signals from the target node and send sink node the difference of arrival of these two different signals. Sink nodes forwards this time difference to the java application and it simply multiplies this difference with the speed of Ultrasonic signals to get distance measurements.

After getting distance measurements application checks if the measurements are correct by checking if distances are below 5m which assures that the target is in the boundary covered by triangle. In our case we have just one equilateral triangle and each side has a length of 5m. After verifying distance measurements application uses trilateration technique to calculate the coordinates. Distance and coordinates estimation for this system are more accurate than previous techniques.

6.5 Comparison of Three System

In this section we talk about the comparison of three localization we implemented in this theses. Table 6.2 shows the comparison of three localization systems. It can be seen that localization system based on Time Difference of Arrival (TDOA) has proven to be accurate than other two systems. It can also be deduced that fingerprinting based localization system performs very poorly. The reason is all the measurements are matched with the data saved in database and we are not updating database measurements. On the other hand Self Calibration Localization based system provides accuracy better than fingerprinting. It is however not that accurate, we have not considered factors such fading and shadowing.

For the future work we intend to extend this work to a whole new level by designing our own microcontroller that can give us timeprecision in nano-seconds.

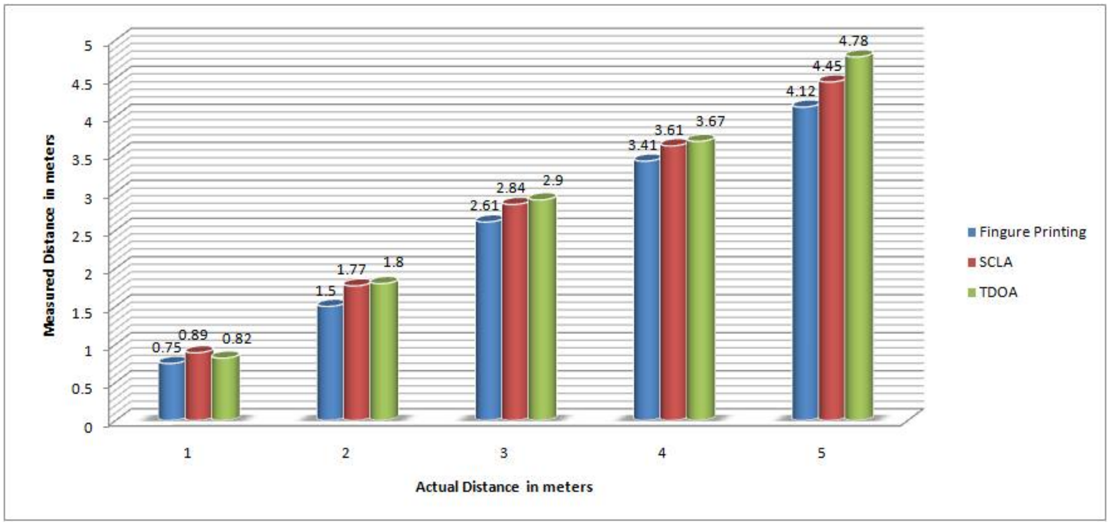


Figure 6.13: Time Difference of Arrival GUI

It will make localization much easier, because that microcontroller will only use RF signals to localize targets. Besides, for RSSI based localization system we intend to consider factors such as long/short term fading, shadowing and human movement in indoor environment. If we can successfully tackle all aforementioned factors, then RSSI based localization system can outrun time-based localization systems. Figure 6.13 shows the barchart representation of the comparison of three localization systems.

Actual Distance	**FingerPrinting**	**SCLA**	**TDOA**
1m	0.75m	0.89m	0.82m
2m	1.5m	1.77m	1.8m
3m	2.61m	2.84m	2.9m
4m	3.41m	3.61m	3.67m
5m	4.12m	4.45m	4.78m

Table 6.2: Comparison of localization systems.

CHAPTER 7

CONCLUSION

In this theses we have carried out an extensive research to address the localization problem in indoor environment. Besides research we have addressed this localization problem with pragmatic approaches. We have implemented three different localization systems on different hardware platform. Fingerprinting and RSSI based localization systems have been implemented on Telosb motes while TDOA based system was implemented on Arduino based platform.

Accuracy of TDOA system has outrun both Fingerprinting and Self calibration reference nodes based system. But the problem with TDOA system is that it requires more energy as compared to other two systems.

We have also used Accelerometer in TDOA system which prevents target node to send unnecessary messages to the network. Target node only sends data when it receives accelerometer readings. By adding this separate module to the node, no unnecessary data is sent to the network.

For the future work we plan to implement TDOA on Telosb motes as Telosb motes require a very little power to run.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, Mar. 2002. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1389128601003024

[2] Y. Sun, S. Huang, J. J. Oresko, and A. C. Cheng, "Programmable neural processing on a smartdust for brain-computer interfaces," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 4, no. 5, pp. 265–273, 2010.

[3] A. Mainwaring, J. Polastre, R. Szewczyk, and D. Culler, "Wireless Sensor Networks for Habitat Monitoring," *Computer and Information Science*, p. 88, 2002.

[4] M. P. M. Srivastava, R. Muntz, "Smart Kindergarten : Sensor-based Wireless Networks for Smart Developmental Problem-solving Environments," *Proceedings o f the Seventh Annual International Conference on Mobile Computing and Networking*, pp. 132–138, 2001.

[5] L. Schwiebert and J. Weinmann, "Research Challenges in Wireless Networks of Biomedical Sensors," *Proceedings of the Seventh Annual International Conference on Mobile Computing and Networking*, pp. 151–165, 2001.

[6] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," *IEEE Communications*, no. August, pp. 102–114, 2002.

[7] D. Anthony and A. John, "Denial of Service in Sensor Networks," *IEEE Computer Society*, 2002.

[8] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS : Security Protocols for Sensor Networks SPINS : Security Protocols for Sensor Networks," 2001.

[9] C. Feature, "Location Systems for Ubiquitous Computing," *Computer and Information Science*, no. August, 2001.

[10] K. Pahlavan and X. Li, "Indoor Geolocation Science and Technology," *IEEE communications magazine*, no. February, pp. 112–118, 2002.

[11] M. Rudafshani and S. Datta, "Localization in wireless sensor networks," in *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*, April 2007, pp. 51–60.

[12] I. Stojmenović, *Handbook of Sensor Networks: Algorithms and Architectures*. John Wiley and Sons, 2005.

[13] P. Kułakowski, J. Vales-Alonso, E. Egea-López, W. Ludwin, and J. García-Haro, "Angle-of-arrival localization based on antenna arrays for wireless sensor networks," *Computers & Electrical Engineering*, vol. 36, no. 6, pp. 1181–1186, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0045790610000352

[14] K. Yu and Y. Guo, "Anchor-free localisation algorithm and performance analysis in wireless sensor networks," *Communications, IET*, vol. 3, no. 4, pp. 549–560, April 2009.

[15] F. Liu, X. Cheng, D. Hua, and D. Chen, *TPSS: A Time-based Positioning Scheme for Sensor Networks with Short Range Beacons*, ser. Signals and Communication Technology, Y. Li, M. Thai, and W. Wu, Eds. Springer US, 2008.

[16] M. Boushaba, A. Hafid, and A. Benslimane, "High accuracy localization method using AoA in sensor networks," *Computer Networks*, vol. 53, no. 18, pp. 3076–3088, 2009.

[17] N. Patwari, A. Hero III, M. Perkins, N. S. Correal, R. J. O'dea, and A. O. Hero, "Relative location estimation in wireless sensor networks," *Signal Processing, IEEE Transactions on*, vol. 51, no. 8, pp. 2137–2148, 2003. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1212671\nciteulike-article-id:3157930\nhttp://dx.doi.org/10.1109/TSP.2003.814469

[18] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 37, no. 6, pp. 1067–1080, Nov 2007.

[19] J. Arias, J. Lázaro, A. Zuloaga, J. Jiménez, and A. Astarloa, "GPS-less location algorithm for wireless sensor networks," *Computer Communications*, vol. 30, no. 14–15, pp. 2904–2916, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366407002174

[20] A. Beck, P. Stoica, and J. Li, "Exact and Approximate Solutions of Source Localization Problems," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1770–1778, 2008. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4472183\npapers3://public

[21] R. Stoleru and J. Stankovic, "Probability grid: a location estimation scheme for wireless sensor networks," in *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, Oct 2004, pp. 430–438.

[22] G. Mao, "Localization algorithms and strategies for wireless sensor networks," *IGI Global*, 2009.

[23] K. W. Cheung, H. C. So, W. K. Ma, and Y. T. Chan, "Least Squares Algorithms for Time-of-Arrival-Based Mobile Location," *IEEE Transactions on Signal Processing*, vol. 52, pp. 1121–1128, 2004.

[24] F. Franceschini, M. Galetto, D. Maisano, and L. Mastrogiacomo, "A review of localization algorithms for distributed wireless sensor networks in manufacturing," pp. 698–716, 2009.

[25] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: a survey," *IEEE Network*, vol. 18, no. 4, 2004.

[26] "A survey on sensor localization," *Journal of Control Theory and Applications*, vol. 8, pp. 2–11, 2010.

[27] Y. Weng, "Source localization using TDOA measurements with sensor location uncertainty," *Proceedings of the 30th Chinese Control Conference*, pp. 5068–5072, 2011.

[28] Rodney G. Vaughan, "Compact Multiport Antennas for High Spectral Efficiency Motivation from Energy Considerations, Lessons from Early Wireless History, and Design," *European Conference on Antennas and Propagation (Eucap 2013)*, 2013.

[29] R. Stoleru, T. He, J. A. Stankovic, and D. Luebke, "A high-accuracy, low-cost localization system for wireless sensor networks," in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, ser. SenSys '05. New York, NY, USA: ACM, 2005, pp. 13–26. [Online]. Available: http://doi.acm.org/10.1145/1098918.1098921

[30] R. Peng and M. Sichitiu, "Angle of arrival localization for wireless sensor networks," in *Sensor and Ad Hoc Communications and Networks, 2006. SECON*

'06. 2006 3rd Annual IEEE Communications Society on, vol. 1, Sept 2006, pp. 374–382.

[31] W. Wang and Q. Zhu, "High accuracy geometric localization scheme for wireless sensor networks," in *Communications, Circuits and Systems Proceedings, 2006 International Conference on*, vol. 3, June 2006, pp. 1507–1512.

[32] F. Belkhouche, B. Belkhouche, and P. Rastgoufard, "Line of sight robot navigation toward a moving goal," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 36, no. 2, pp. 255–267, 2006.

Bilal Saeed

Contact No.+966598426200

Computer Engineering Department, King Fahd University of Petroleum and Minerals, Dhahran

Email: iambilalsaeed@gmail.com

Education

2013- May 2015 (Expected):MS Computer Networks CGPA 3.2 (Coursework Completed)
King Fahd University Of Petroleum and Minerals, KFUPM , Saudi Arabia

2007-2011: BS in Telecommunication Engineering with CGPA 3.51
National University of Computer & Emerging Sciences, FAST, Pakistan

Thesis Title: *Human Tracking and Localization in Indoor using Wireless Sensor Networks*
Thesis Supervisor: ***Dr Tarek R. Sheltami***

Publications

1. Muhammad Musaddiq, Shehryar Khan, Uthman Baroudi, and **Bilal Saeed**. 2014. Performance Evaluation of IEC 61850 under Wireless Communication Networks. In *Proceedings of the 6th International Conference on Management of Emergent Digital EcoSystems* (MEDES '14). ACM, New York, NY, USA, pp 90-94

2. Basem Almadani, Shehryar Khan, Tarek R Sheltami, Elhadi M Shakshuki, Muhammad Musaddiq, **Bilal Saeed**, "Automatic Vehicle Location and Monitoring System based on Data Distribution Service,"Elsevier Procedia Computer Science Vol. 37, pp. 127-134, 2014.

3. Basem Madani, **Bilal Saeed**, Muhammad Musaddiq, and Shehryar Khan , “Healthcare Systems Integration Using Real Time Publish Subscribe (RTPS) Middleware”, Journal of Computer Networks and Communications, Feb, 2015 “Under Review”

Research Experience

Human Tracking and Localization in Indoor using Wireless Sensor Networks

- Working on the development of Human tracking and localization algorithms for indoor using wireless sensor networks. In this project we carry out extensive research on existing localization algorithms and based on the performance of these algorithms we propose our own algorithm.
  ***Supervisor : Dr Tarek R. Sheltami*.**

Automatic Vehicle Location and Monitoring System based on DDS

- Proposed a Real-Time Automatic Vehicle Location (AVL) and Monitoring system for traffic control of pilgrims coming towards the holy city of Makkah in Saudi Arabia based on data distribution service (DDS) specified by the Object

Management Group (OMG), which is a Real-Time Publish/Subscribe Middleware.

Supervisor : Dr Basem Almadani

Performance Evaluation of IEC 61850 under Wireless Communication Networks

- The standard IEC 61850 is considered as a candidate for communication standard for smart grid applications. The delay performance is one of the critical issues that were specified by IEC 61850. This project involved the modeling and simulation of IEC 61850 substation automation system (SAS) under wireless and hybrid networks. OPNET is used to model and simulate the Intelligent Electronic Device (IEDs).

  Supervisor: Dr Uthman Baroudi

Funded Projects

1. **Intelligent Job Site (NSTIP Project) – 2013-2014**

I worked on the part of this project where I developed an Android Application. The Purpose of the application was to collect readings from job site ambient sensors – Temperature, Pressure and Humidity connected in Wireless Sensor Network. Based on the readings it shows how critical the site is. The overall cost of the project was 1.2m SAR.

2. **Worked on Several Projects on Odesk.**

Programming and Simulation Skills

1. SQL
2. PHP
3. MySQL
4. C
5. C++
6. Visual Basic
7. Assembly Language
8. Linux
9. Programmable Intelligent Controllers (PIC)
10. Digital Electronic Circuit Design and Development
11. FORTRAN
12. Javascript

13. Asynchronous Java And XML (AJAX)
14. Cascaded Style Sheet (CSS)
15. 3D animation using Eovia Carrara
16. Flash
17. Fireworks

Graduate Courses

1. Mobile Computing
2. Computer Networks
3. Computer Networks Design
4. Computer and Network Security
5. Network Management
6. Heterogeneous Computing
7. Client Server Programming

Professional Certifications:

Cisco Certified Network Associate (CCNA)
Dated (15th May 2013)

SAP training (May, 2014)

Honors and Awards

- Awarded fully funded Scholarship for Master's studies at KFUPM Saudi Arabia
- Awarded fully funded Scholarship for Bachelor's studies at NU-FAST Islamabad, Pakistan by the Ministry of ICT&RD. The overall price was 1.3m PKR.
- Enlisted in Dean's List of Honours' at NU-FAST for 3 times.
- Runner UP in Speed Programming at NU-FAST - 2009.
- Winner IEEE robo cup at NU-FAST – 2010.

References

Dr. Tarek R Sheltami (*tarek@kfupm.edu.sa)*
Dr. Ahraf Mahmoud (ashraf@*kfupm.edu.sa)*
Dr. Mohsin Siddiqui (mohsin@kfupm.edu.sa)