Improving the accuracy of Indoor localization System using smart phones

Project Report submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology.

in

Computer Science & Engineering

under the Supervision of

Mr. Shailendra Shukla

By

Srajan
111227

to

Jaypee University of Information and Technology

Waknaghat, Solan – 173234, Himachal Pradesh

Certificate

This is to certify that project report entitled "**Improving accuracy of Indoor Localisation System using smart phones**", submitted by "**Srajan"**in partial fulfillment for the award of degree of Bachelor of Technology in Computer Science & Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

14May 2015

Mr.Shailendra Shukla

Assistant Professor

Acknowledgement

I take this opportunity to express my profound gratitude and deep regards to my guide Mr. Shailendra Shukla for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by his time to time shall carry me a long way in the journey of life on which I am about to embark.

The in-time facilities provided by the Computer Science department throughout the project development are also equally acknowledgeable.

At the end I would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work.

14May 2015 Srajan

111227

Table of contents

Abbreviationsused

S.no	Abbreviated words	Full Forms
1.	LBS	location based services
2.	LST	least sqaure trilateration
3.	AP	Access point
4.	TOA	Time of arrival
5.	RFID	Radio frequency identification
6.	TDOA	Time differece of arrival
7.	RSSI	Received signal strength
8.	IMU	Innertial measurement sensors
9.	GPS	Global positioning system
10.	MANET	Mobile Adhoc Network
11.	IEEE	Institute of Electrical and Electronic Engineering
12.	SLAM	Simultaneous Localization And Mapping
13.	A-GPS	Assisted gps
14.	LAN	Local area network
15.	LLS	Linear least square
16.	NLLS	Non linear least square

List of Figures

Abstract

Modern smartphones are a great platform for Location Based Services (LBS). While outdoor LBS for smartphones has proven to be very successful, indoor LBS for smartphones has not yet fully developed due to the lack of an accurate positioning technology. Smartphones that uses the innate ability of mobile phones to detect wifi signal strength using wifi analyzer.

This strategy usesthe results of trilateration technique which determines the position of users in indoor areas based on Wi-Fi signal strengths from access points (AP) within the indoor vicinity. In this paper, percentage of signal strengths obtained from Wi-Fi analyzer in a smartphone were converted into distance between users and each AP. But due to inaccuracy of only trilateration technique, we used a method to improve the result i.e **least square trilateration**.The method of **least squares**is a standard approach to the approximate solution of overdetermined systems. A user's indoor position could then be determined using a formula proposed based on this trilateration technique.

1. Introduction

1.1 Objective

Indoor localization as mentioned above is a research area that has attracted quite a lot of attention recently. A lot of work has been done on different innovative approaches to solve this problem. In this project I will analyze these technologies for advantages, disadvantages from a practical deployment point of view, and then implement a system combining the best and most practical technologies i.e **"the least square trilateration**" to increase the accuracy of location prediction. The goal is to bring together these technologies that currently exist and create a working implementation.

We will be implementing these technologies on the Android platform creating an Android application that can localize a user in COM1 using just the phone itself. It should also be extensible to localize in other places as well.COM1 building will be used as the test bed for testing the app. The results of this app will be compared to using the technologies used in silos to see how combining them affects the localization. I will also look into some possible simulation model to test out the app for different factors.

1.2 Motivation

Outdoor localization has already been tapped to its best potential and next natural step is moving indoors. The current outdoor localization technology (mainly GPS) does not extend to the indoors due to loss of GPS signal. This has called for a need to develop innovative indoor localization technologies. This need for indoor localization for pervasive mobile computing has led to a lot of interesting research in the past ten years. Some of the initial research included use of specialized emitters and sensors placed inside buildings to localize objects and people. This method though quite accurate is not scalable for commercial deployments and involves a certain overhead cost in installing and maintaining the additional infrastructure.

Another popular method makes use of the existing infrastructure using wireless access points to triangulate and localize using mobile devices. This method is quite accurate but it usually requires extensive surveying and training effort to build a radio frequency (RF) map of the building. There are also improvements to this method which reduce the efforts or eliminate them completely but at the cost of accuracy. With mobile technology becoming more powerful over the last few years, it is now embedded with more sensors which can improve the accuracy of the prediction by combining them with these earlier technologies. This is the motivation behind this project in trying to use fairly new methods such as accelerometer readings, acoustic background sound system and barometer readings in addition to wireless RF maps to increase the probability of predicted positions.

1.3 Methodology

Indoor localization is one of essential technologies for many applications, such as energy-efficient buildings, disaster rescue, and indoor navigation. Numbers of Indoor Positioning System (IPS) has been proposed and implemented.

Examples include WiFi Received Signal Strength Indicator (RSSI)[13] from all Access Point (AP) in the building, FM radio signal features (RSSI, SNR, multi-path, etc.), acoustic background spectrum . Another group of systems look into installing extra infrastructures as beacons in building, where users carry receiver of the signal sent by beacons to form fingerprints. The technologies used by beaconsinclude ultrasound[2] , infrared[3], magnetic induction, RFID[2]. Moreover, rich sensors available on smartphones stimulates researchers to bring the idea of dead reckoning into trajectory estimation and localization . Basically, they use Inertial Measurement Sensors (IMS)[14] embedded in commercialsmartphones (e.g. accelerometer, gyroscope, compass, magnetometers) to estimate the velocity and direction of users and in turn estimate the trajectory and the location given the known start point.

However, no agreement has been made on a general solution as Global Positioning System (GPS)[15] in outdoor localizations, which doesn't work well for indoor environments because of the signal attenuation inside buildings. Almost all indoor localization techniques have specific assumptions, which in turn become limitations that thwart their large scale real-world deployments. Firstly, there are general problems for different types of techniques. IMS performs for sporadic movements and suffers from cumulative drift errors. Fingerprint-based methods require the signature at each positionto be consistent by time, and the interference should not low as well. But at this stage there is no universal consistent signal. Moreover, beaconing techniques need extra infrastructures that are not common in buildings, which will bring lots of cost on devices, setup, and maintenance. Secondly, some IPSs are designed for specific scenarios. As an example, SurroundSense is designed for shopping mall, where there are always distinguishable features for all adjacent stores and similar types of stores in terms of light, color, and sound.

We argue that their limitations don't mean these systems are useless, because of following points. First, the limitations don't always happen at the same time, and some of them have actually been proven to be almost mutually independent.

Liu et al[13]. look at the consistency of fingerprint of WiFi signal and FM signal, and found for most locations, there are at least one consistent signal. Second, it is not hard to detect whether the context satisfies the assumptions. For example, we can regard accelerometer data with very large variance as uncommon or too complex movement from users. The signal consistency can be checked when smartphone detects little movement from accelerometer (i.e. the phone is staying at one location). Third, crowdsourcing from user inputs is also a good ways to determine the context, especially when proceeded in a motivating manner, like games or competitions.

Another reason for the absence of consensus on indoor localization solution is their heterogeneous goals. Some systems pay attentions on the physical precision of the estimated location, such as some Simultaneous Localization And Mapping(SLAM)[17] application. Some other systems only care about the specific locations relevant to the application. A building energy tracking system only cares about the zones around electronic devices. In addition, instead of accurate physical location estimation, some applications need fine-grained room level positioning, which we call as semantic localization. Semantic localization differs from physical localization in the sense that two physically nearby points could locate in two different (and maybe adjacent) rooms, imaging two locations against the same wall form each side; and two points in one room can be apart from each other at meter level, imaging two points locating at two diagonal corners. A consumer behavior analytics system needs accurate estimations of the stores the consumervisited.

The first solution used in this is to treat the unknown position as the point of intersection of the surfaces of several spheres, whose centers are known fixed positions. Alternatively, by linearizing the equations and converting the problem into one of finding the intersection of several planes, more useful solution techniques can be established .

The nonlinear least squares technique provides the most accurate results of all methods proposed in this thesis. It calculates the exact position when exact distances are used; and a reasonably accurate position when used with approximate distances. The accuracy of the calculated position is degraded when the elevation of the unknown position is above the elevationof the lowest beacon position, and when the unknown position is located outside the perimeter of the beacon positions

The key point is to assign higher credits and weights to more possibly accurate results, and combine them using appropriate data fusion methods. Each IPS is required to implement its own way of predicting the accuracy of its result, which we call as confidence. The overhead for existing IPSs to be integrated is independent of existing implementation, but depends on how accurate they want to the confidence to be. The algorithms to calculate the confidence vary between different techniques, and how well the designers and developers understand the localization methods.

For example, **Fingerprint[8]** based confidence could be more confident if the signal is consistent. Based on their confidences and the history performance (i.e. the credit), the framework assigns weight to each of them. IPSs with more accurate confidences in history gain higher credits. The idea is to recognize the bestresults by conservatively trusting their confidence.

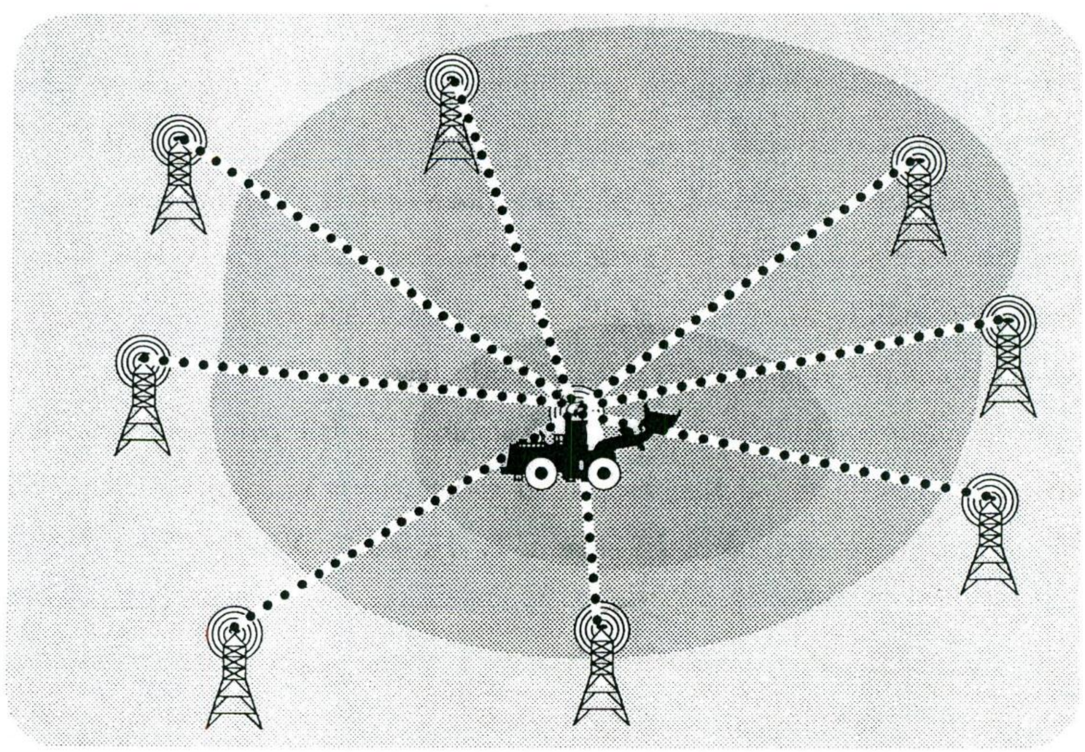


Figure 1: Illustration of positioning problem[15]

1.4 Application

The usefulness of the mathematical solution of the trilateration positioning problem presented in this thesis is not restricted to personal position detection. This solution could be used to improve the accuracy of existing trilateration applications, or it could be implemented in new electronic positioning systems. Possible applications of this solution include precision farming; underwater positioning; navigational aids for ships, aircraft, or automobiles; determining the position of aircraft, rockets, missiles, and satellites; and any of the existing trilateration systems listed below.

Various positioning systems which apply trilateration principles are currently in use worldwide. A brief search in the literature led to the following examples. There are numerous military applications of trilateration. One such application is the Global Positioning System, which was developed by the Department of Defense. This system uses a constellation of satellites which transmit radio signal timing data to mobile receivers on earth. These receivers calculate unknown locations by using the satellites known orbital ephemeris, the radio signal timing data, and a combination of the Doppler principle and trilateration. The satellites produce two signals, the Standard Positioning Service which is available for civilian use, and the more accurate Precise Positioning Service which is coded and restricted to military use.

Accurate elevationsare not available from this system on a real time basis [Remondi 1991][6]. Logan International Airport uses a radar based trilateration system for locating and identifying aircraft and other transponder equipped vehicles on the surface of the airport [Manning 1979][6]. The Geodetic Survey of Canada uses a trilateration system that involves the use of an aircraft flying between ground stations. In this application the distances between the aircraft and the ground stations are computed to calculate a position on the surface of the earth [DeLoach 1963]. The Rome Air Development Center at Griffiss Air Force Base uses a ground based system to determine the position of aircraft in flight[6] .

1.5System Design

Our Indoor Positioning consists of three parts viz. client device (which can be wifi enabled device like mobile, smartphones, PDA etc), existing network of at least 3 wifi access points (AP's) and server. The client device is capable of wifi access and receiving wifi signals from all access points in the vicinity. The key purpose of client is to record the wifi signals strength data i.e. RSSI. The network of wifi access points acts as communication channel between client server. All access points used in the system are similar. D-LINK wifi AP's are used in system. The server is a laptop which runs Windows 7 . The main task of server is of positioning and navigation system. Hence, it is also called as positioning server. The framework for indoor positioning system is depicted in figure 2.

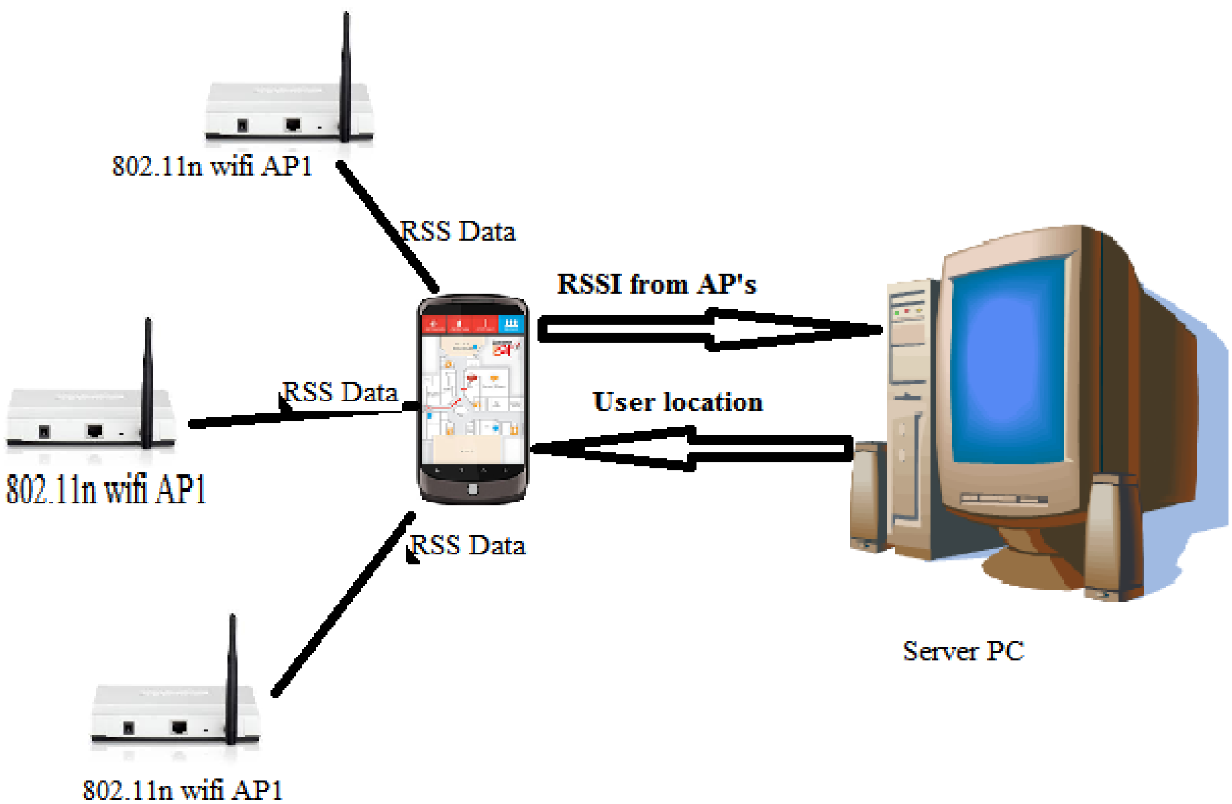


Figure 2: Framework of indoor positioning system

The Android application “WiFiScan” scans the wifi access points in the vicinity of the device. It collects the data like signal strength from the respective access point in dBm, MAC address of AP, channel frequency, SSID etc. This data is the key point in positioning of user in indoor system. Android device then sends this RSS data along with respective MAC address of AP to the positioning server. The server runs the positioning algorithms which calculate the location co-ordinates of the user. Server runs two types of algorithms a) To calculate distance of user from respective AP and b) Trilateration algorithm to find exact location of user from distance. Server then sends these location co-ordinates back to the client device. The android device has the front end application showing indoor map of the system. The location co-ordinates calculated by server are shown as the user position. This is a part of user positioning in indoor environment..

1.6Properties of Good Localisation System

1. Should be light enough to run locally on the phone.
2. Should require minimum effort in setting up (fingerprinting).
3. Should be stable with variation of time.
4. Should be robust to changes in the environment.
5. Should work equally well with any measurement device (any mobile phone).
6. Should be accurate enough to predict the room the user is in.

2. Related Works

Indoor GPS positioning system[15] is a modular system used to track and locate persons or objects inside buildings. Nowadays, Indoor GPS plays an important role in various domain including consumer’s applications, emergency services, machines or gadgets and for military purposes . GPS is now embedded into mobile phones which provide Location Based Services (LBS) . It is known as assisted GPS or A-GPS which is built to overcome the limitation of GPS. GPS is only good for outdoor environment activities and work poorly in indoor environment . One of the limitations of indoor GPS is weak signal acquisition because GPS signal is weak inside buildings and cannot penetrate building wall structures which affect performance in coordinate measurement or position detection .

On the other hand, finger printing is another alternative in position determination. It requires comparison of signals from current measurements with a pre-measured data in particular locations . There are two phases in fingerprinting which is offline training phase and online estimation phase. Wi-Fi signal strength is an example of offline training phase in finger printing method.

Another solution for position determination involves detection of proximity such as Radio-Frequency Identification (RFID) and Bluetooth technology. Most of the researcher tries to apply mobile devices in their study. In Mobile Adhoc Network (MANET), devices can randomly move in any directions. Stationary nodes broadcast the hello massages signals and the node received the signal automatically determines the location position itself based on three signals received from 3 anchor nodes and run the Kernel AODV[16] platform.Some researchers have done experiments at a shield tunnel construction site using the fingerprint method of Received Signal Strength Indication (RSSI) from each Access Point (AP). Another way to determine position of object is by using RFID.

Some former work of combing multiple indoor localization techniques resulted to significant performance improvements. Azizyan et al. take WiFi, sound, light, and color features from mobile phone sensors as signatures, and sequentially and gradually filter estimated position candidates with multiple signature-based techniques, each of which uses the output candidates set of the former one as input[7]. The improvement brought about by this cascaded methodology is proved to increase with the number of available sensors increases in general. However, they also observe sound filtersometimes rule out the correct positions, which gives more motivation for us to look into the determinability of the intermediateaccuracy before blindly combing them. Chen et al. [13] combine WiFi and FM signal indicators as one signature. They found the interferences to WiFi and FM signals causing erroneous results happen independently. Thus, using integrated signature almost remove all the errors, drastically increasing the localization accuracy from around 80% up to 98%. We believe this phenomenon is also applicable to other localization techniques, and will bring more benefits if more appropriate intelligence are introduced.

However, no comprehensive and general study has been done for combing multiple IPSs. Rai et al [13] build an fully-automated indoor localization system called Zee, which features no training phase, and the system converge pretty neatly by combining IMU-based dead-reckoning and WiFi signature-based localizations. Zee uses estimated trajectory to determine the positions on the map, and records the WiFi signatures simultaneously to build the signature database from zero. WiFi signatures in history are used to calibrate the localization methods. This is a good example of combining different techniques, but they don't have in-depthinvestigation on general combinations.

Paiidya et al[13]. introducethe idea of combining different wireless IPSs because they are not avaliable everywhere, and the coverage of collection will provide a more pervasive services. However, they only use averaging without weightsand have no discussions about the rationale.

3. Wifi Trilateration

The method used for improving the accuracy of an indoor position system is **Wi-Fi Trilateration[15]** which uses Wi-Fi signal strength relating with distance formula to determine position of a user. Based on the concept of GPS, minimum of three access points (AP) are needed to determine the position of a user in an indoor location. The Wi-Fi signals are in the form of radio wave where the movements of the signals are highly dependent on the frequency. Signals with different diameters are transmitted by APs in all direction according to the respective signal strength. Since wireless routers provide coverage of about 100 feet (30.5 meters), signal strength is used to find the collision point in order to specify the accurate position of an object.

3.1 Overview

The standard protocol of Wi-Fi is 802.11 which was introduced by Institute of Electrical and Electronic Engineering (IEEE) and it is used in wireless LAN[14]. The standards come in several flavors which are 802.11a that transmits at 5 GHz and can move up to 54 megabits of data per second. On the other hand, 802.11b is the slowest and slightly less expensive and transmits in the 2.4 GHz frequency and can carry 11 megabits of data per second. Networking standard 802.11g also transmits at 2.4 GHz like 802.11b but it is much faster and theoretically can handle up to 54 megabits of data per second. 802.11n is the newest standard to improve speed and range. These kinds of protocol standards allow communication via internet through channels of communication medium that is available in Wi-Fi. There are 14 channels available in Wi-Fi where the use of each channel can be selected to avoid interferences in the wireless transmissions.This study deploys Wi-Fi technique in conjunction with IEEE 802.11g networking standard. Here,we assume the three APsare known as AP1, AP2, and AP3.

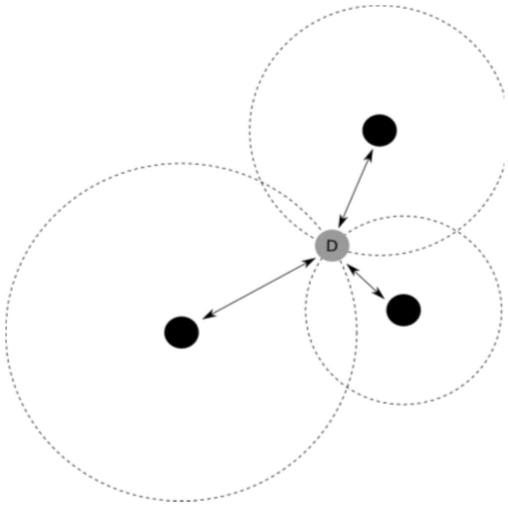


Figure 4: Three access points[15]

3.2 Mathematics Involved

Assume that the coordinates of the three APs as Figure 2:

Then, based on three coordinates of the APs, we need to find the coordinates of the user's position that is represented as *Θ*

Θ=(x, y, z) : spatial coordinates of use.

Let's assume that a user is using a smart phones that serves as a receiver of the signals transmitted from the access points. Application of Wi-Fi analyzer in the smart phone presents the signal strength in terms of percentage. The highest percentage of signal strength indicates that *Z* is closest to the AP whereas the lowest percentage implies that Z is maximum range of AP.

The percentage of signal strength obtained from the Wi-Fi analyzer can be converted to distance between a user's to each AP using this equation (Equation 1):

***Distance, di = p* (1– *mi*)**…………….(**3.1**)

Where;

m = is the percentage of signal strength

p = is the maximum coverage of signal strength

i = 1,2,3……n

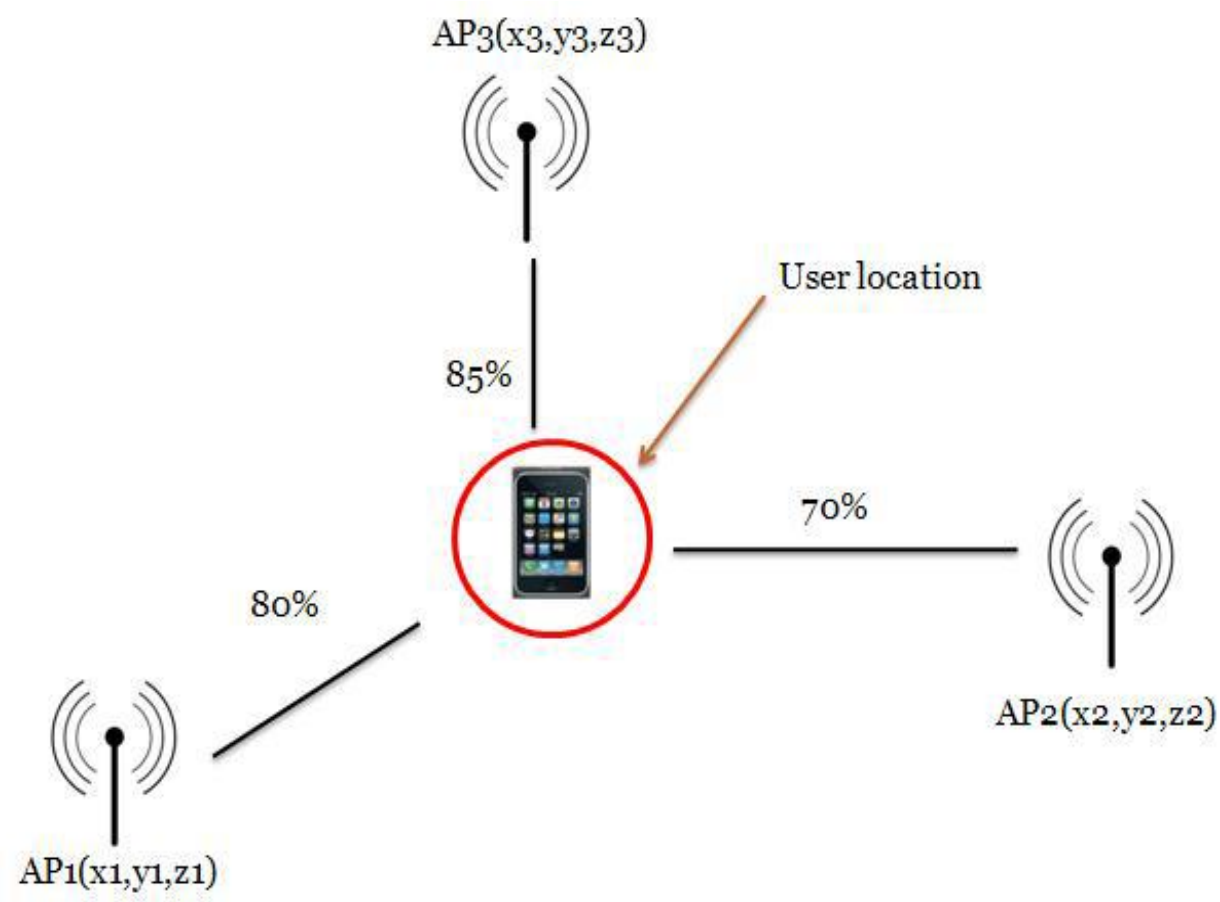


Figure 5: Illustration WiFi signal strenght from three access points[15]

From Figure 3, let each AP be placed at the center. Assume a scenario where a student who uses a smart phone, is looking for a book in a library. Then, we assuming that signal strength for each AP will spread the signal in wave forms. The signal strength will form 3 circles and intersect each other. The intersection of 3 circles is the position of user and we want to determine the location of user who is labeled by *B* (*x*, *y*). To simplify the calculations, the equations are formulated so intersection of circle is occurred at Cartesian plane (see Figure 2). The equation for any of these circle is as follow (assuming $z = 0$):

$$(x - x_i)^2 + (y - y_i)^2 = r_i^2 \qquad \text{..................(3.2)}$$

The intersection of 3 circles is obtained by solving systems of linear equations for 2 variables simultaneously. The linear systems are solved in order to determine the coordinates *x* and *y*.

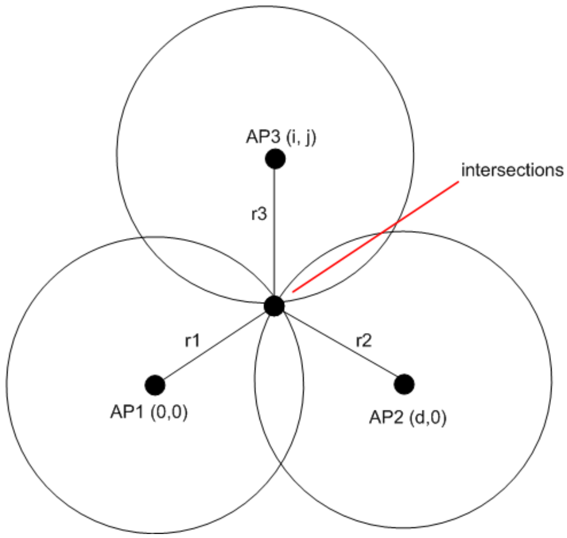


Figure 6:Intersections of 3 circles[15]

Based on Figure 4, we start with the equations for three circles:

$$r_1^2 = x^2 + y^2 + z^2 \qquad \text{..................}(\mathbf{3.3})$$

$$r_2^2 = (x-d)^2 + y^2 + z^2 \qquad \text{................}(\mathbf{3.4})$$

$$r_3^2 = (z-i)^2 + (y-j)^2 + z^2 \qquad \text{.................}(\mathbf{3.5})$$

To determine the location of B, we have to solve for (*x*, *y*, *z*).

The method to do it is by using systems of linear equations for 2 variables and solve these eaquation of linear system $\tilde{A}\ x = b$. By using this method, the *j*th constraints is used as a linearizing tool. Adding and subtracting *xj*, *yj* and *zj* in (3.3),(3.4) ,(3.5), gives:

$$(x - x_j + x_j - x_i)^2 + (y - y_j + y_j - y_i) + (z - z_j + z_j - z_i)^2 = r_i^2$$

..............**(3.6)**

With (*i* = *1, 2,...* , *j+1* , ... , *n*).

Linear system is easily written in matrix form $\tilde{\boldsymbol{A}}\boldsymbol{x} = \boldsymbol{b}$**,**

$$A = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ \vdots & \ddots & \vdots \\ x_n - x_1 & y_n - y_1 & z_n - z_1 \end{bmatrix}$$

$$\vec{x} = \begin{bmatrix} x - x_1 \\ y - y_1 \\ z - z_1 \end{bmatrix}$$

$$\vec{b} = \begin{bmatrix} b_{21} \\ \vdots \\ b_{n1} \end{bmatrix}$$

……………(3.7)

Based on the calculation by using (3.7), the position of *B* is given by (*x*, *y*, *z*).

4. Linearization

The procedures which were used to linearize and solve the equations 3.2 for the intersection of several spheres are based on geometry, linear algebra, and analysis. This linearization process reduces the degree, and converts the problem into one of finding the point of intersection of several planes. The solution of the system of linearized equations is completely determined when the exact distances from four beacons are known.

4.1Development of the Linear System

The following mathematical notation was introduced in Section 3
The constraints are the equations of the spheres with radii ri,

$$(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 = {r_i}^2 \qquad (i = 1, 2, ..., n). \quad \text{…………………} \quad (4.1)$$

The jth constraint is used as a linearizing tool. Adding and subtracting xj , yj and zj in (4.1) gives

$$(x - x_j + x_j - x_i)^2 + (y - y_j + y_j - y_i)^2 + (z - z_j + z_j - z_i)^2 = {r_i}^2 \quad \text{................}(4.2)$$

with (i = 1, 2, ..., j − 1, j + 1, ..., n).
Expanding and regrouping the terms, leads to

$$\begin{aligned}
&(x - x_j)(x_i - x_j) + (y - y_j)(y_i - y_j) + (z - z_j)(z_i - z_j) \\
= \; &\frac{1}{2}[(x - x_j)^2 + (y - y_j)^2 + (z - z_j)^2 \\
&-{r_i}^2 + (x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2] \\
= \; &\frac{1}{2}[{r_j}^2 - {r_i}^2 + d_{ij}^2] = b_{ij},
\end{aligned}$$

where $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$ ….(4.3)is the distance between beacons Bi and Bj.
Since it does not matter which constraint is used as a linearizing tool, arbitrarily select the first constraint (j = 1). This is analogous to selecting the first beacon.

Since i = 2, 3, ..., n, this leads to a linear system of (n − 1) equations in 3 unknowns:

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

………………(4.4)

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

……………….(4.5)

.
.
.

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

…………………(4.5)

This linear system is easily written in matrix form

$$Ax = b,$$ … … … … …(4.6)

With

$$\mathbf{A} = \begin{pmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ \vdots & \vdots & \vdots \\ x_n - x_1 & y_n - y_1 & z_n - z_1 \end{pmatrix}$$

$$\vec{x} = \begin{pmatrix} x - x_1 \\ y - y_1 \\ z - z_1 \end{pmatrix}, \quad \vec{b} = \begin{pmatrix} b_{21} \\ b_{31} \\ \vdots \\ b_{n1} \end{pmatrix}.$$

……………….(4.7)

The linear system (4.6) has (n−1) equations in three unknowns. Therefore, theoretically only four beacons (n = 4) are needed to determine the unique position of a piece of person in the mine; provided no more than two beacons are co-linear.

4.2 Geometrical Interpretation of linear system

In this section I digress by showing two alternative geometrical techniques to derive the linear system (4.6). The two alternative methods involve the use of analytic geometry, and trigonometry. In turn, they provide a nice geometrical interpretation of the equations in (4.6).

4.2.1 Analytic Geometry

The following analysis uses analytic geometry to show that each of the equations in the linear system (4.6) represents a plane.
Select equation (4.4), the first equation of the system (4.6), for analysis:

$$(x-x_1)(x_2-x_1)+(y-y_1)(y_2-y_1)+(z-z_1)(z_2-z_1)$$
$$=\frac{1}{2}[{r_1}^2-{r_2}^2+d_{21}^2]=b_{21}. \quad \ldots\ldots\ldots\ldots\ldots\ldots(4.8)$$

In Figure 5 , the points B1 (x1, y1, z1) and B2 (x2, y2, z2) refer to two beacon locations and point P (x, y, z) represents the unknown point. The point O represents the (arbitrary) origin of the cartesian coordinate system. Further denote the plane PB1B2 by α.
Straightforward algebra allows us to rewrite (4.8) in the form

$$(x-x_0)(x_2-x_1)+(y-y_0)(y_2-y_1)+(z-z_0)(z_2-z_1)=0 \quad \ldots\ldots\ldots\ldots\ldots(4.9)$$

With

$$x_0 = x_1+\frac{b_{12}}{d_{12}^2}(x_2-x_1)=\frac{1}{2d_{12}^2}[({r_2}^2-{r_1}^2)(x_1-x_2)+d_{12}^2(x_1+x_2)]$$

$$y_0 = y_1+\frac{b_{12}}{d_{12}^2}(y_2-y_1)=\frac{1}{2d_{12}^2}[({r_2}^2-{r_1}^2)(y_1-y_2)+d_{12}^2(y_1+y_2)]$$

$$z_0 = z_1+\frac{b_{12}}{d_{12}^2}(z_2-z_1)=\frac{1}{2d_{12}^2}[({r_2}^2-{r_1}^2)(z_1-z_2)+d_{12}^2(z_1+z_2)] \quad \ldots\ldots\ldots\ldots(4.10)$$

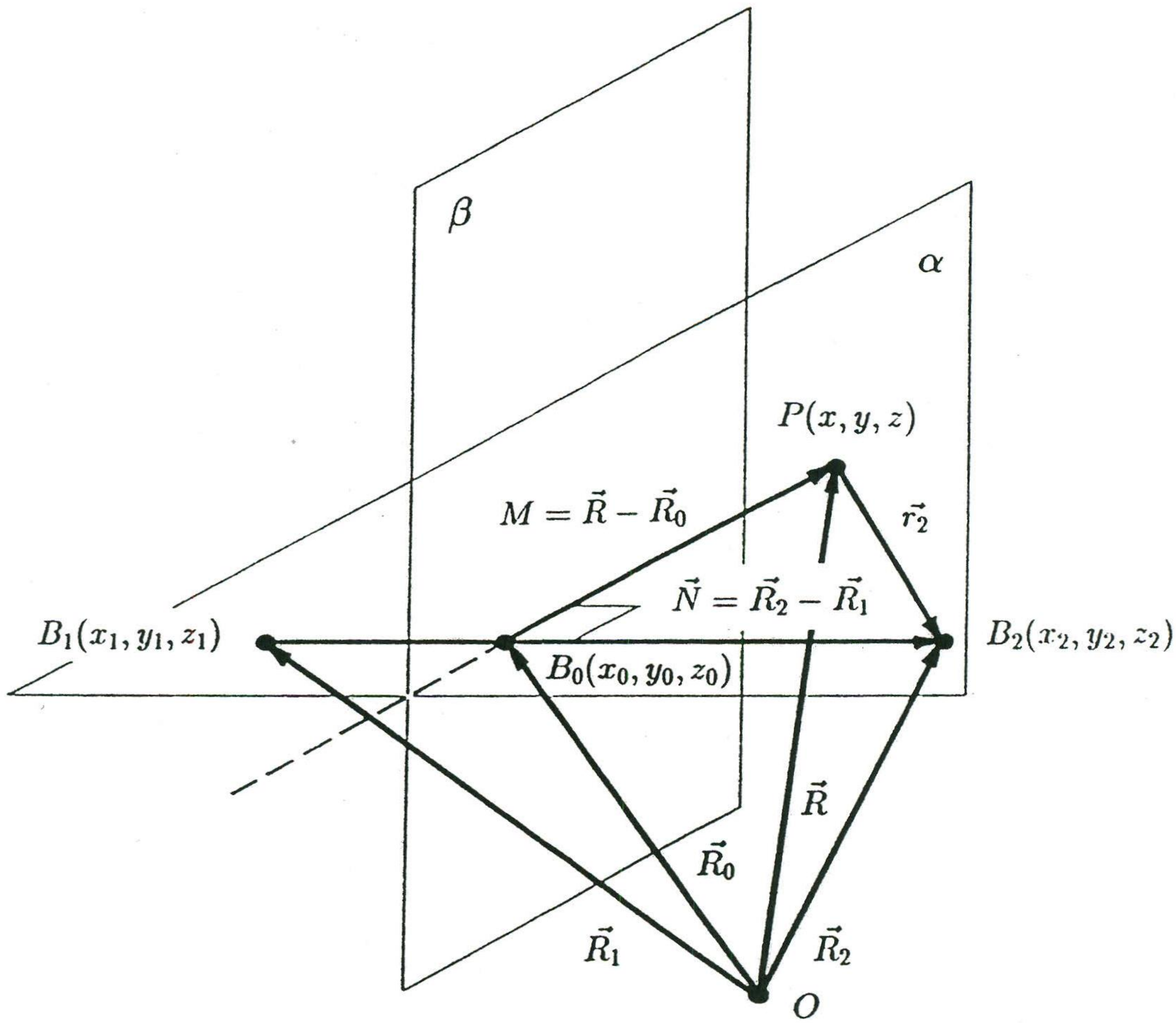


Figure 7: Analytical geometrical interpretation[6]

Using analytic geometry, it is obvious that (4.9) is the normal form of the plane β containing the points P(x, y, z) and B0 (x0, y0, z0), with normal vector $\vec{N} = \vec{R_2} - \vec{R_1}$. Vector N has components (x2−x1, y2−y1, z2−z1). Vector M is defined as $\vec{M} = \overrightarrow{B_0P} = \vec{R} - \vec{R_0}$, with components (x−x0, y−y0, z−z0). Equation (4.9) is obtained by expressing that vector M is orthogonal to vector N . In mathematical notation this is

$\vec{M} \cdot \vec{N} = 0$.

The planes α and β are thus orthogonal.

They intersect along the line carrying the vector $\vec{M} = \overrightarrow{B_0P}$.

A plane β is obtained for each of the equations in the linear system (4.6). The coordinatesof the point of intersection of these β-planes is the location P(x, y, z) of the equipment in the mine. Three of these β-planes are needed to uniquely determine this position.

4.2.2Trigonometry

The following analysis uses trigonometry to derive the linear system (4.6). Consider the triangle B1B2P in Figure 6, and let O be the arbitrary origin of the coordinate system. Recall that $\vec{r_1} = \overrightarrow{B_1P} = \vec{R} - \vec{R_1}$ has components (x − x1, y − y1, z − z1) and

$\vec{r_2} = \overrightarrow{B_2P}$.

Furthermore, $\vec{N} = \vec{R_2} - \vec{R_1} = \vec{r_2} - \vec{r_1}$ has components (x2 − x1, y2 − y1, z2 − z1). Applying the cosine rule in the triangle B1B2P, and taking into account that $r_1 = \|\vec{r_1}\|, r_2 = \|\vec{r_2}\|$

and $d_{12} = \|\vec{N}\|,$, one obtains

$$\begin{aligned} {r_2}^2 &= {r_1}^2 + d_{12}^2 - 2\vec{r_1} \cdot \vec{N} \\ &= {r_1}^2 + d_{12}^2 - 2[(x - x_1)(x_2 - x_1) + (y - y_1)(y_2 - y_1) + (z - z_1)(z_2 - z_1)] \end{aligned} \quad ..(4.12)$$

which is nothing else than equation(4.4) the first equation of the linear system(4.6)

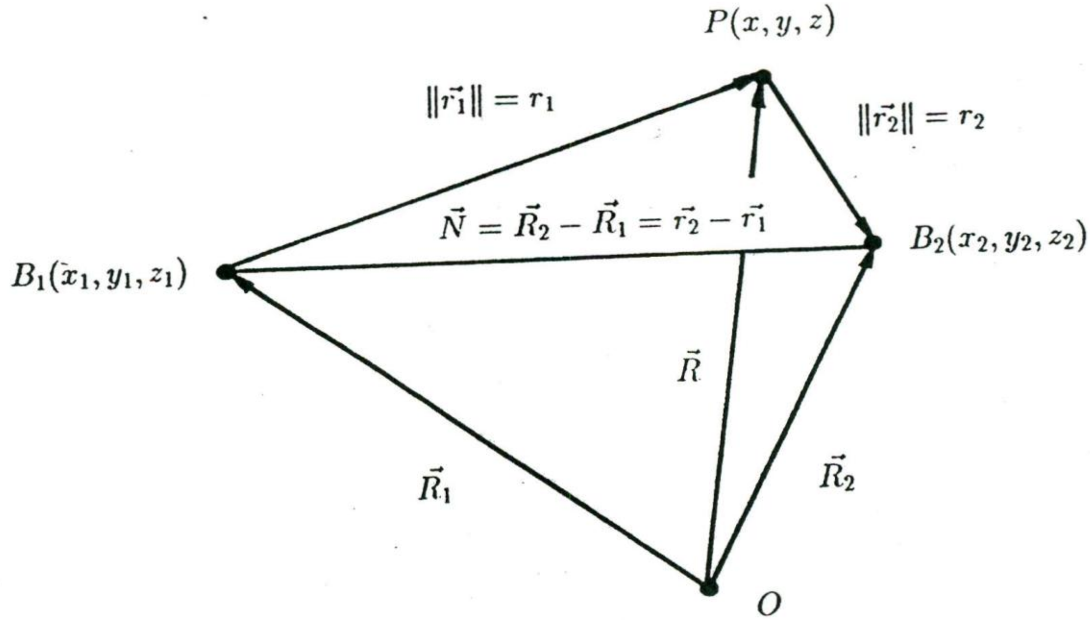


Figure 6: Trigonometric Interpretation[6]

5. Linear Least Square Method

In this chapter we will show that applying the linear least squares method to the linear system (4.6) is an unacceptable solution technique because it does not calculate the locations within a tolerance of five feet when used with approximate distances. The equipment locations obtained by the entire linear least squares method are generally more accurate than the locations obtained by solving four equations of the linear system (4.6) directly.

5.1 Development of the Linear Least Squares Method

In practice, the distances ri are only approximate. Thus 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,

$$S = \vec{r}^{\,T}\vec{r} = (\vec{b} - \mathbf{A}\vec{x})^T(\vec{b} - \mathbf{A}\vec{x}) \quad \text{.....................(5.1)}$$

leads to the normal equation,

$$\mathbf{A}^T\mathbf{A}\vec{x} = \mathbf{A}^T\vec{b} \quad \text{...................(5.2)}$$

There are several methods to solve (5.2) for $\vec{x}$.
The condition number of A^TA determines which method is best.
If A^TA is non-singular and well-conditioned then

$$\mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b} \quad \text{....................................(5.3)}$$

is used.
If A^TA is singular or badly conditioned then the normalized QR-decomposition of A is generally used. In this method A = QR, where Q is an orthonormal matrix and R is upper-triangular matrix. The solution for $\vec{x}$ in the normalizedQR-decomposition is then found from

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

by back substitution when A is full rank.
It may happen that the matrix A^TA is close to singular even when the original matrix A was not close to singular. For situations like that, QR decomposition may overcome the problem.If not, singular value decomposition (SVD) can be used to solve the least squares problem fairly accurately.

5.2 Singular Value Decomposition (SVD)

The optimal solution $\vec{x}_0$ is then given by $\vec{x}_0 = \mathbf{A}^+\vec{b}$. The pseudo-inverse $\mathbf{A}^+ = \mathbf{V}\mathbf{\Sigma}^+\mathbf{U}^H$ involves the unitary matrices U,Voccurring in the SVD of A, this is A = U_VH. The matrix ∑+ is obtained from the "diagonal" matrix ∑ asfollows: The p×q matrix ∑ has entries <∑>ij= 0 if i ≠ j and <∑>ij= $\sigma_i \geq 0$ for 1 ≤ i ≤k and k + 1 ≤ i ≤min{p, q}. The numbers σ_i are called the singular values. The matrix ∑+ is then the q × p matrix whose nonzero entries are
<∑+ >ii= 1/σ_i, for 1 ≤i ≤ k.
To detect degeneracy of the matrix A one computes the ratio σ_1/σ_n, where $\sigma_1$is the largest singular value and $\sigma_n$is the smallest singular value when A is full rank. The ratio σ_1/σ_n may be regarded as a condition number of the matrix A.
The smallest singular value, σ_n is the distance in the 2-norm from A to the nearest singular matrix. The fact that $\sigma_1/\sigma_n$is small may be considered as a condition of near-singularity of A.

6. Non Linear Least Square-Trilateration

The nonlinear least squares method developed in this chapter is acceptable for use. The accuracy of the z coordinate calculated from approximate distances is within a tolerance of 5.0 m. This accuracy is attainable if the equipment is inside the perimeter of the beacons at an elevation that is not close to or above the elevation of the lowest beacon.

6.1 Development of the Nonlinear Least Squares Method

The sum of the squares of the errors on the distances is minimized in this least squares method. Recall that r_i denotes the approximate distance between the equipment , and the ith beacon; and that ˆri stands for the exact distance,

i.e

$$(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 = \hat{r_i}^2. \quad \ldots\ldots\ldots(6.1)$$

To minimize the sum of the squares of the errors on the distances, one must minimize the function

$$F(x, y, z) = \sum_{i=1}^{n} (\hat{r_i} - r_i)^2 = \sum_{i=1}^{n} f_i(x, y, z)^2, \quad \ldots(6.2)$$

With

$$f_i(x, y, z) = \hat{r}_i - r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} - r_i. \quad \ldots(6.3)$$

Minimizing the sum of the square errors is a fairly common problem in applied mathematics for which various algorithms are available. Numerous different approaches can be taken, from simple to very complicated [Mikhail 1976]. The Newton iteration was selected from among those available to find the 'optimal' solution P(x, y, z).

A 'good' initial guess for (˜x, ˜y, ˜z) is obtained from the linear least squares method. A 'better guess' for z can be obtained by solving the constraint equations .

The only case considered is the case for which Fmin > 0 and therefore n > 3. Differentiating (6.2) with respect to x yields

$$\frac{\partial F}{\partial x} = 2\sum_{i=1}^{n} f_i \frac{\partial f_i}{\partial x}.$$ ………….(6.4)

The formulae for the partials with respect to y and z are similar. Introducing the vectors $\vec{f}, \vec{g}$ and the Jacobian matrix J, leads to

$$\vec{g} = 2\mathbf{J}^T \vec{f},$$ ………….(6.5)

Where,

$$\mathbf{J} = \begin{pmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial z} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial z} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_n}{\partial x} & \frac{\partial f_n}{\partial y} & \frac{\partial f_n}{\partial z} \end{pmatrix}, \quad \vec{f} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix}, \quad \vec{g} = \begin{pmatrix} \frac{\partial F}{\partial x} \\ \frac{\partial F}{\partial y} \\ \frac{\partial F}{\partial z} \end{pmatrix}.$$

…………….. (6.6)

Using the vector $\vec{R}$

$$\vec{R} = \begin{pmatrix} x \\ y \\ z \end{pmatrix},$$ …………….(6.7)

Newton iteration gives

$$\vec{R}_{\{k+1\}} = \vec{R}_{\{k\}} - \left(\mathbf{J}_{\{k\}}^T \mathbf{J}_{\{k\}}\right)^{-1} \mathbf{J}_{\{k\}}^T \vec{f}_{\{k\}},$$ …….(6.8)

where $\vec{R_{\{k\}}}$ denotes the kth approximate solution. The subscript {k} in J and $\vec{f}$ means that these quantities are evaluated at $\vec{R_{\{k\}}}$.

Obviously $\vec{R}_{\{1\}} = (\tilde{x}, \tilde{y}, \tilde{z})^T$.

Using the explicit form of the function fi(x, y, z) leads to

$$\mathbf{J}^T\mathbf{J} = \begin{pmatrix} \sum_{i=1}^{n} \frac{(x-x_i)^2}{(f_i+r_i)^2} & \sum_{i=1}^{n} \frac{(x-x_i)(y-y_i)}{(f_i+r_i)^2} & \sum_{i=1}^{n} \frac{(x-x_i)(z-z_i)}{(f_i+r_i)^2} \\ \sum_{i=1}^{n} \frac{(x-x_i)(y-y_i)}{(f_i+r_i)^2} & \sum_{i=1}^{n} \frac{(y-y_i)^2}{(f_i+r_i)^2} & \sum_{i=1}^{n} \frac{(y-y_i)(z-z_i)}{(f_i+r_i)^2} \\ \sum_{i=1}^{n} \frac{(x-x_i)(z-z_i)}{(f_i+r_i)^2} & \sum_{i=1}^{n} \frac{(y-y_i)(z-z_i)}{(f_i+r_i)^2} & \sum_{i=1}^{n} \frac{(z-z_i)^2}{(f_i+r_i)^2} \end{pmatrix}, \quad \text{…..(6.9)}$$

And

$$\mathbf{J}^T\vec{f} = \begin{pmatrix} \sum_{i=1}^{n} \frac{(x-x_i)f_i}{(f_i+r_i)} \\ \sum_{i=1}^{n} \frac{(y-y_i)f_i}{(f_i+r_i)} \\ \sum_{i=1}^{n} \frac{(z-z_i)f_i}{(f_i+r_i)} \end{pmatrix}. \quad \text{…………(6.10)}$$

In practice this type of iteration works fast, in particular when the matrix $\mathbf{J}^T\mathbf{J}$is augmented by a diagonal matrix which effectively biases the search direction towards that of steepest decent.
Levenberg and Marquardt [Lawson, and Hanson 1974] developed this improvement.

As the solution is approached such modifications can be expected to have a decreasing effect.

7. Working and Implementation

7.1 Technology used

Wi-Fi is a local area wireless computer networking technology that allows electronic devices to network, mainly using the 2.4 gigahertz (12 cm) UHF and 5 gigahertz (6 cm) SHF ISM radio bands.

The Wi-Fi Alliance defines Wi-Fi as any "wireless local area network" (WLAN) product based on the Institute of Electrical and Electronics Engineers' (IEEE) 802.11 standards". However, the term "Wi-Fi" is used in general English as a synonym for "WLAN" since most modern WLANs are based on these standards. "Wi-Fi" is a trademark of the Wi-Fi Alliance. The "Wi-Fi CERTIFIED" trademark can only be used by Wi-Fi products that successfully complete Wi-Fi Alliance interoperability certification testing.

Many devices can use Wi-Fi, e.g. personal computers, video-game consoles, smartphones, digital cameras, tablet computers and digital audio players. These can connect to a network resource such as the Internet via a wireless network access point. Such an access point (or hotspot) has a range of about 20 meters (66 feet) indoors and a greater range outdoors. Hotspot coverage can comprise an area as small as a single room with walls that block radio waves, or as large as many square kilometres achieved by using multiple overlapping access points.

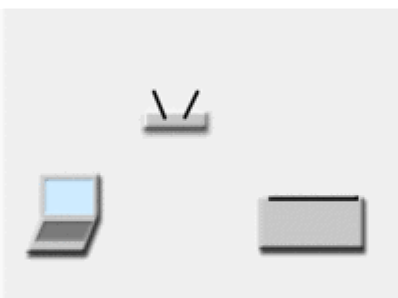
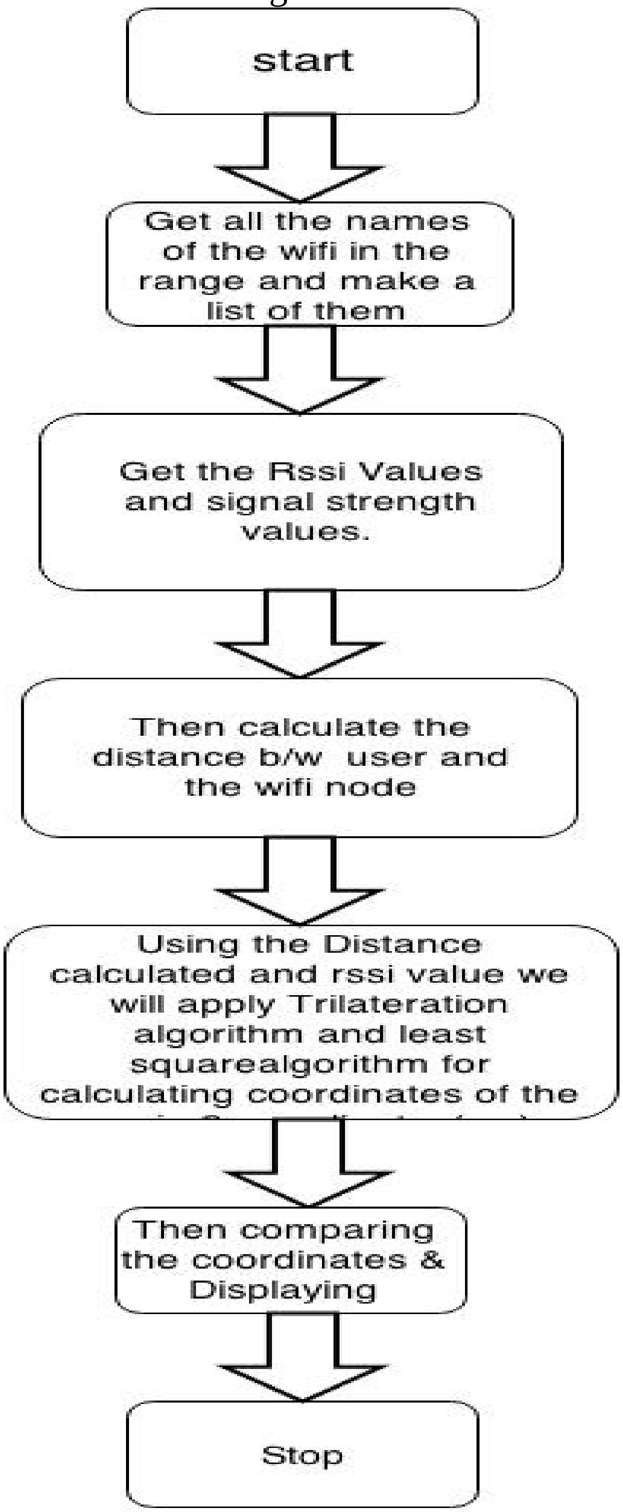


Figure 7: ***Depiction of a device sending information wirelessly to another device, both connected to the local network, in order to print a document***

7.2 Flowchart

This is how the algorithm works



7.3Code

This code is for an app which calculates the coordinates of the user using least square algorithm. The code we have implemented looks for the available wifi networks in the sorroundings and by selecting any of the network it gives us the signal strength of that wifi network. Using the obtained signal strength it then calculates the distance of the wifi points. For our algorithm to work, we need only three wifi access points. So it takes in only top three available wifi points and distance is calculated.

Firstly in this we take our position to be origin. So the signal strength and distance is calculated of all the three wifi points. The signal strength at position 1 i.e. at origin is calculated. So a circle's equation is formed with origin as the centre. Now taking y direction as north we move 2 m in y direction. The signal strength there will be different and hence the distance. So the distance at position 2 is now calculated and a new equation will be formed with that point as the centre of that circle whose radius is the distance between the mobile device and the wifi point. Then with the same axis, we move 2 m in x direction. Again the signal strength and the distance will be different. So third equation is formed with that as the center. This process is done for all the three wifi points and the coordinates of are calculated.

Now having the coordinates of the wifi points the user's location is calculated using least square trilateration method.

```
public class MainActivity extends Activity {

   WifiManager mainWifiObj;
   WifiScanReceiver wifiReciever;
   ListView list;
   String wifis[];
   List<ScanResult> wifiScanList;
public void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
list = (ListView)findViewById(R.id.list);
```

```
mainWifiObj = (WifiManager) getSystemService(Context.WIFI_SERVICE);
wifiReciever = new WifiScanReceiver();
mainWifiObj.startScan();

   }
```

The above module of the code creates a list view for the wifi names and creates an object of wifi manager to get the names and values of the wifi in the range.

```
int pos=0;
        String selectedFromList =(String) (list.getItemAtPosition(position));
for(int i = 0; i < wifiScanList.size(); i++){
if(((wifiScanList.get(i)).SSID.toString()).matches(selectedFromList))
            {
pos=i;
            }

        }
int strength1 = ((wifiScanList.get(0)).level);
int strength2 = ((wifiScanList.get(1)).level);
int strength3 = ((wifiScanList.get(2)).level);
int value1=WifiManager.calculateSignalLevel(strength1,11);
int value2=WifiManager.calculateSignalLevel(strength2,11);
```

```
int value3=WifiManager.calculateSignalLevel(strength3,11);
                String val1=Integer.toString(value1);
                String val2=Integer.toString(value2);
                String val3=Integer.toString(value3);
                String rssi=String.valueOf(strength1);
                //                Toast.makeText(MainActivity.this,val                ,
Toast.LENGTH_SHORT).show();
                Intent intent = new Intent(MainActivity.this,Activity2.class);
intent.putExtra("Name", selectedFromList);
intent.putExtra("Strength1", val1);
intent.putExtra("Strength2", val2);
intent.putExtra("Strength3", val3);
intent.putExtra("Rss",rssi);
startActivity(intent);
```

The above module gets the strength and Rssid of the wifi clicked. And then passing it to the next page of the app using the intent function. We are passing the name on which the user click's,strength of the top 3 wifi's and rssi value of the wifi on which the user has clichéd.

```
public class WiFiScanReceiver extends BroadcastReceiver {
        private static final String TAG = "WiFiScanReceiver";
          WiFiDemo wifiDemo;

        public WiFiScanReceiver(WiFiDemo wifiDemo) {
        super();
            this.wifiDemo = wifiDemo;
          }
```

```
  @Override
public void onReceive(Context c, Intent intent) {
   List<ScanResult> results = wifiDemo.wifi.getScanResults();
   ScanResult bestSignal = null;
for (ScanResult result : results) {
if (bestSignal == null
        ||          WifiManager.compareSignalLevel(bestSignal.level,
result.level) < 0)
bestSignal = result;
   }

   String message = String.format("%s networks found. %s is the
strongest.",
results.size(), bestSignal.SSID);
Toast.makeText(wifiDemo,                                   message,
Toast.LENGTH_LONG).show();

Log.d(TAG, "onReceive() message: " + message);
  }
```

The above module is giving us the wifi scan results and listing them .

```
public boolean onOptionsItemSelected(MenuItem item) {
		// Handle action bar item clicks here. The action bar will
		// automatically handle clicks on the Home/Up button, so
long
		// as you specify a parent activity in AndroidManifest.xml.
		int id = item.getItemId();
		if (id == R.id.action_settings) {
			return true;
		}
		return super.onOptionsItemSelected(item);
	}
```

The above module will pass the id of the element selected so that rssid and wifi strength can be shown of the selected item.

```
	intent=getIntent();
level1  = intent.getStringExtra("Strength1");
	strength1 =Float.parseFloat(level1);
level2  = intent.getStringExtra("Strength2");
	strength2 =Float.parseFloat(level2);
level3  = intent.getStringExtra("Strength3");
	strength3 =Float.parseFloat(level3);
```

The above code gets the value’s passed from the previous page using the get string extra function.

```
s1=(TextView)findViewById(R.id.t5);
	s2=(TextView)findViewById(R.id.t2);
	s3=(TextView)findViewById(R.id.t3);
```

```
        s4=(TextView)findViewById(R.id.t7);
        s5=(TextView)findViewById(R.id.t6);
        s6=(TextView)findViewById(R.id.t8);
        s7=(TextView)findViewById(R.id.t9);
        String n;
        n=s2.getText().toString();
        n=n+" "+rss;
s2.setText(n);
        n=s3.getText().toString();
        n=n+" "+level1;
s3.setText(n);
        d1=10*(1-(strength1/10));
        d2=10*(1-(strength2/10));
        d3=10*(1-(strength3/10));

        n=s4.getText().toString()+d1;
s4.setText(n);
        n=s6.getText().toString()+d2;
s6.setText(n);
        n=s7.getText().toString()+d3;
s7.setText(n);
initializeViews();
```

In the above line of codes we are making objects of the text views we have created in the app layout (activity_activity2.xml). Then we calculate the distance using the strength and then initializing all the views.

```
sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
if
(sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)
!= null) {
        // success! we have an accelerometer

accelerometer = sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
sensorManager.registerListener(this, accelerometer, SensorManager.SENSOR_DELAY_NORMAL);
    } else {
        // fai! we dont have an accelerometer!
    }
```

The above line of codes gets an object of the sensor listener which will get the changes in the accelerometer values.

```
void calculate()
  {
float r1,r2,r3;//radius of the circles. distance caluclated from the strength
    float x1,x2,x3,y2,y3,y1;
x1=(int )(Math.random() * 50 + 1);
x2=(int )(Math.random() * 50 + 1);
x3=(int )(Math.random() * 50 + 1);
    y1=(int )(Math.random() * 50 + 1);
    y2=(int )(Math.random() * 50 + 1);
    y3=(int )(Math.random() * 50 + 1);
double d21,d31;
double b2,b3,x,y,z=0;
```

```
x3=(x1+x2)/2;
        y3=5;
        r1=d1;
        r2=d2;
        r3=d3;//distance using strength
                /*double rad1=(double)(x1*x1 + y1*y1 + z1*z1);
                r1=(float)Math.sqrt(rad1);
                double          rad2=(double)(Math.pow(x-x2,2)+Math.pow(y-
y2,2)+Math.pow(z-z2,2));
                r2=(float)Math.sqrt(rad2);
                double          rad3=(double)(Math.pow(x-x3,2)+Math.pow(z-
z3,2)+Math.pow(z-z3,2));
                r3=(float)Math.sqrt(rad3);*/
        d21=Math.sqrt(Math.pow((x2-x1), 2)+Math.pow((y2-y1), 2));
        d31=Math.sqrt(Math.pow((x3-x1), 2)+Math.pow((y3-y1), 2));
        b2=0.5*(Math.pow(r1, 2)- Math.pow(r2, 2)+ Math.pow(d21, 2));
        b3=0.5*(Math.pow(r1, 2)- Math.pow(r3, 2)+ Math.pow(d31, 2));
        x= b2/x2;
        y= (b3-(x*x3))/y3;
        //display x,y,z

        String n=s1.getText().toString()+x;
s1.setText(n);
        n=s5.getText().toString()+y;
s5.setText(n);


    }
```

The above function will calculate the coordinates using the trilateration algorithm and then changing the values of some text views where we want to show that.

```
public void onSensorChanged(SensorEvent event) {


displayCurrentValues();
        // get the change of the x,y,z values of the accelerometer
deltaX = Math.abs(lastX - event.values[0]);
deltaY = Math.abs(lastY - event.values[1]);
deltaZ = Math.abs(lastZ - event.values[2]);



    }
```

The above function will call the display current values function when the accelerometer values changes.

```
public void displayCurrentValues() {
currentX.setText(Float.toString(deltaX));
currentY.setText(Float.toString(deltaY));
currentZ.setText(Float.toString(deltaZ));
    }
```

The above function will show the current values of the x, y, z coordinates provided by the accelerometer.

```
Intent intent;
String level1,level2,level3;
String rss1;String rss2;String rss3;
float  xx1,xx2,xx3;
float d1,d2,d3;
float x11,x12,x13;
float y11,y12,y13;
String name;
```

```
Button b,b2,b3;
float x0=0,  y0=0,  z0=0;
float x2_ ;
float y2_  ;
float z2_ ;

float x3_  ;
float y3_  ;
float z3_ ;

TextView s3,s4,s5;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_activity2);
        intent=getIntent();
        level1  = intent.getStringExtra("Strength1");
          xx1 =Integer.parseInt(level1);
        level2  = intent.getStringExtra("Strength2");
          xx2 =Integer.parseInt(level2);
        level3  = intent.getStringExtra("Strength3");
          xx3 =Integer.parseInt(level3);
          b=(Button)findViewById(R.id.button1);
          b2=(Button)findViewById(R.id.button2);

          rss1=intent.getStringExtra("Rss1");
          rss2=intent.getStringExtra("Rss2");
          rss3=intent.getStringExtra("Rss3");
        name=intent.getStringExtra("Name");
         // s1=(TextView)findViewById(R.id.t1);
         // s2=(TextView)findViewById(R.id.t2);
          s3=(TextView)findViewById(R.id.t3);
          s4=(TextView)findViewById(R.id.t4);
          s5=(TextView)findViewById(R.id.s5);
            d1=10*(1-(xx1/1000));
```

```
d2=10*(1-(xx2/1000));
d3=10*(1-(xx3/1000));
s4.setText("Distance for first wifi" + d1+"\n"+
        "Distance for second wifi" + d2+"\n"+
        "Distance for third wifi" + d3+"\n"+
        "Rss for first wifi" + rss1+"\n"+
        "Rss for second wifi" + rss2+"\n"+
        "Rss for third wifi" + rss3+"\n"
            );


b2.setOnClickListener(new OnClickListener(){

        @Override
        public void onClick(View arg0) {
            int strength1 = ((GlobalClass.GetInstance(
).getWifiList().get(0)).level);
        int strength2 = ((GlobalClass.GetInstance(
).getWifiList().get(1)).level);
        int strength3 = ((GlobalClass.GetInstance(
).getWifiList().get(2)).level);

        int
value1=WifiManager.calculateSignalLevel(strength1,1001);
        int
value2=WifiManager.calculateSignalLevel(strength2,1001);
        int
value3=WifiManager.calculateSignalLevel(strength3,1001);

          y11=10*(1-(value1/1000));
          y12=10*(1-(value2/1000));
          y13=10*(1-(value3/1000));

        calculate();
```

```
                    s4.setText("our coordinates with respect to the 1st
wifi" + x0 +","+ y0 +","+z0 +","+"\n"+
                              "coordinates of 2nd wifi " + x2_ +","+
y2_ +","+z2_ +","+"\n"+
                              "coordinates of 3rd wifi " + x3_ +","+
y3_ +","+z3_ +",");
                    }});

          b.setOnClickListener(new OnClickListener(){

               @Override
               public void onClick(View arg0) {
                    int strength1 = ((GlobalClass.GetInstance(
).getWifiList().get(0)).level);
               int strength2 = ((GlobalClass.GetInstance(
).getWifiList().get(1)).level);
               int strength3 = ((GlobalClass.GetInstance(
).getWifiList().get(2)).level);

               int
value1=WifiManager.calculateSignalLevel(strength1,1001);
               int
value2=WifiManager.calculateSignalLevel(strength2,1001);
               int
value3=WifiManager.calculateSignalLevel(strength3,1001);

               x11=10*(1-(value1/1000));
               x12=10*(1-(value2/1000));
               x13=10*(1-(value3/1000));


               s4.setText("Distance for first wifi" + x11+"\n"+
                         "Distance for second wifi" + x12+"\n"+
                         "Distance for third wifi" + x13+"\n"+
```

```
                    "Rss for first wifi" + strength1+"\n"+
                    "Rss for second wifi" + strength2+"\n"+
                    "Rss for third wifi" + strength3+"\n"
                        );

            }});



    }
    void calculate()
    {

        float d1d = d1;
        float  d2d=d2;
        float d3d=d3;
        float   x1   =   (float)   ((Math.pow(d1d,2)-Math.pow(x11,2))/4
+1);
        float y1=(float) ((Math.pow(d1d,2)-Math.pow(y11,2))/4 +1);
        float             z1=(float)             Math.sqrt((Math.pow(d1d,2)-
Math.pow(x1,2)-Math.pow(y1,2)));
        float x2=(float) ((Math.pow(d2d,2)-Math.pow(x12,2))/4 +1);
        float y2=(float) ((Math.pow(d2d,2)-Math.pow(y12,2))/4 +1);
        float z2=(float) Math.sqrt(Math.pow(d2d,2)-Math.pow(x2,2)-
Math.pow(y2,2));
        float x3=(float) ((Math.pow(d3d,2)-Math.pow(x13,2))/4 +1);
        float y3=(float) ((Math.pow(d3d,2)-Math.pow(y13,2))/4 +1);
        float z3=(float) Math.sqrt(Math.pow(d3d,2)-Math.pow(x3,2)-
Math.pow(y3,2));

        //DISPLAY -"SHIFTING WIFI 1 AS ORIGIN
        //      position of user is "

        x0=-x1;
          y0=-y1;
```

```
        z0=-z1;

        x2_=x2-x1;
        y2_=y2-y1;
        z2_=z2-z1;

        x3_=x3-x1;
        y3_=y3-y1;
        z3_=z3-z1;
}
```

7.4 Analysis

The linear least squares method gives the most accurate results of both methods developed and examined, when approximate distances are involved in the calculations.

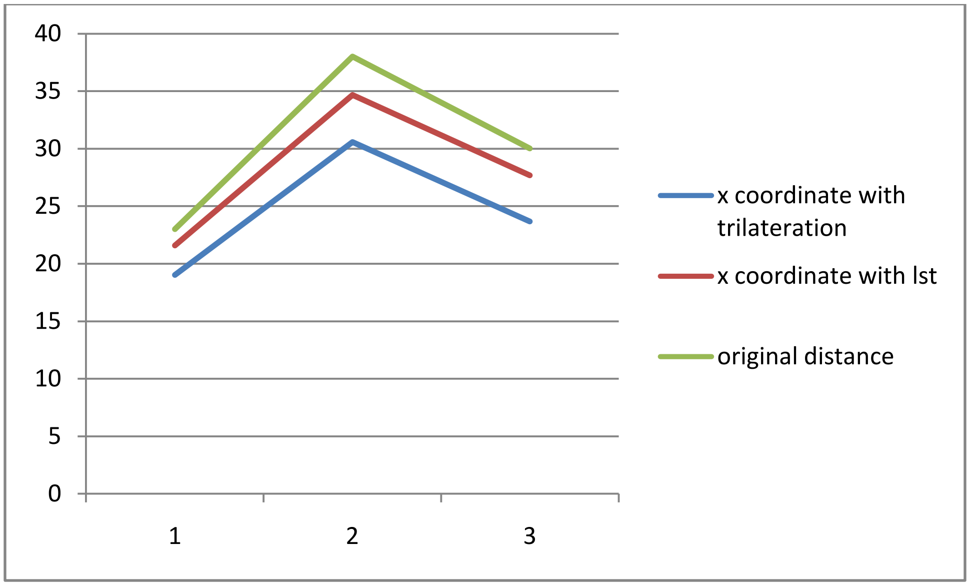
The least squares solution procedure calculates the coordinates of the equipment within the required tolerance of 5.0 m for both exact and approximate distances. The use of this method should be restricted to situations where equipment in the is inside the perimeter of the beacons, and below the more or less common plane of the beacons. This method will provide results if these constraints are violated. The accuracy of the solution decreases as the elevation of the equipment increases, and as the equipment moves farther outside the perimeter of the beacons.

When exact distances were used, the least squares solution technique calculated the x and y coordinates within a tolerance of 0.0 m. When approximate distances were used, the calculated x and y coordinates were within a tolerance of 0.375 m. The accuracy of the z coordinate, calculated with approximate distances, increases as the constraints are imposed. The z tolerance was found to be m feet.

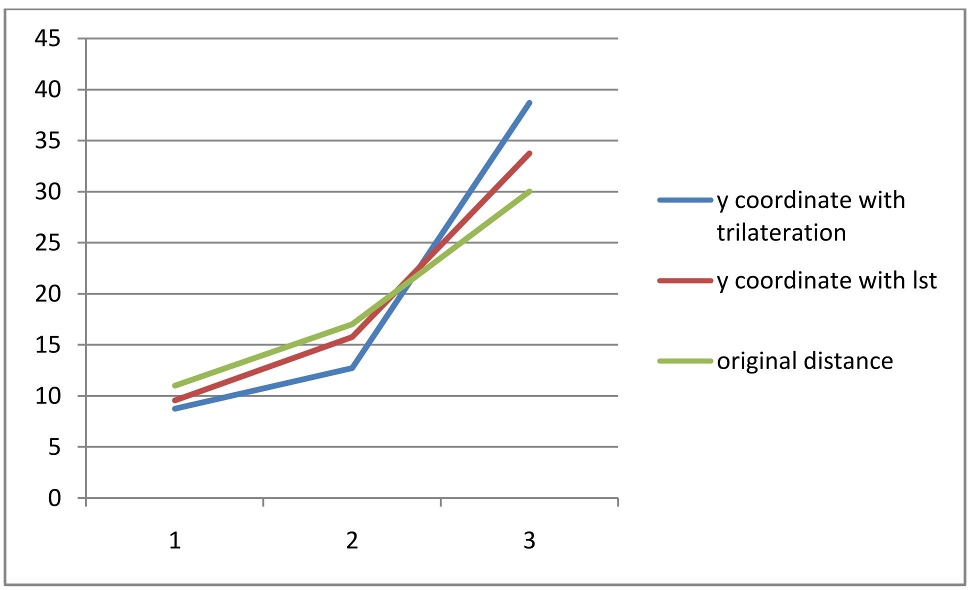
S.no	Coordinates obtained from trilateration.	Distance from 1st wifi.	Distance from 2nd wifi.	Distance from 3rd wifi.	Corrected coordinates.

1.	19.02385,8.76048	7.76334	10.52854	16.08504	21.58302,9.57832
2.	30.58275,12.74628	10.84730	17.83726	24.46302	34.67382,20.78372
3.	23.67183,38.6932	18.78345	20.87352	29.74831	27.68325,33.73846

Table 1: Table showing the coordinates.



Graph 1: showing variations of x coordinate



Graph 2 : showing variations in y coordinate

7.5 ScreenShots

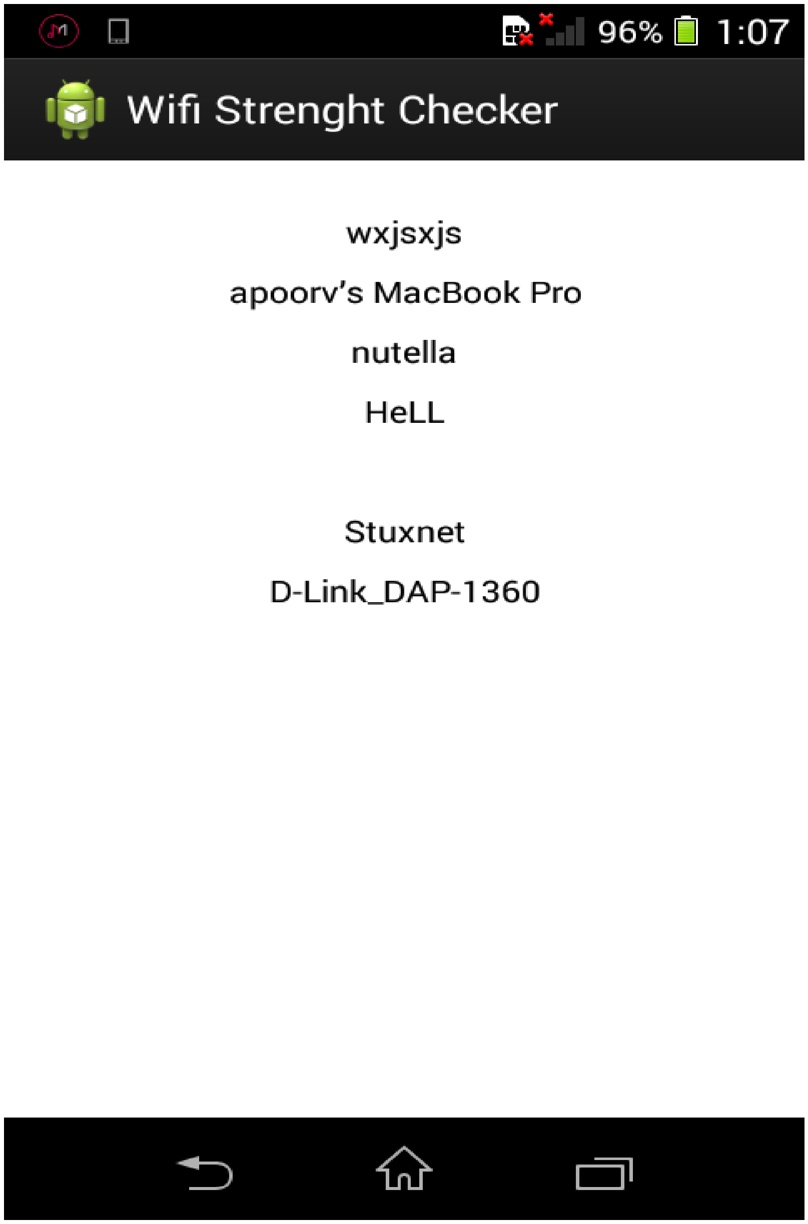


Figure 8: App showing all the wifi's in the region.

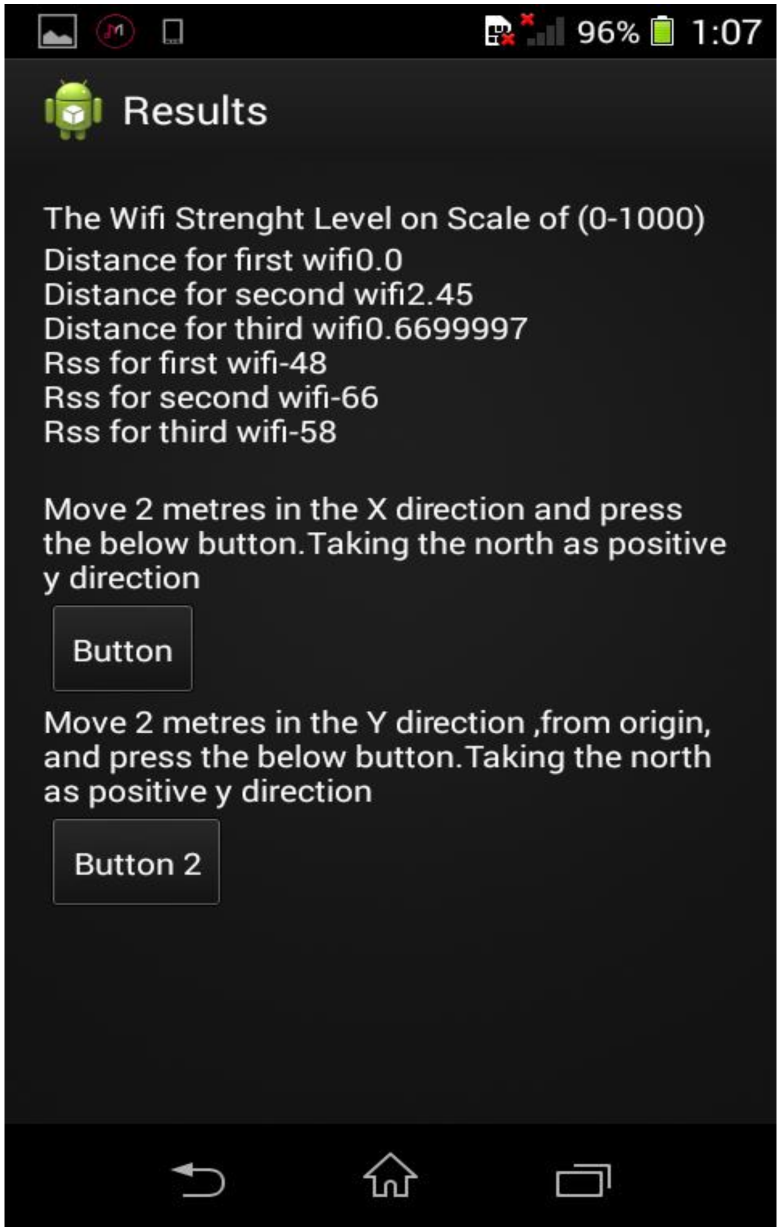


Figure 9: App showing the RSSID and wifi Strength of a wifi

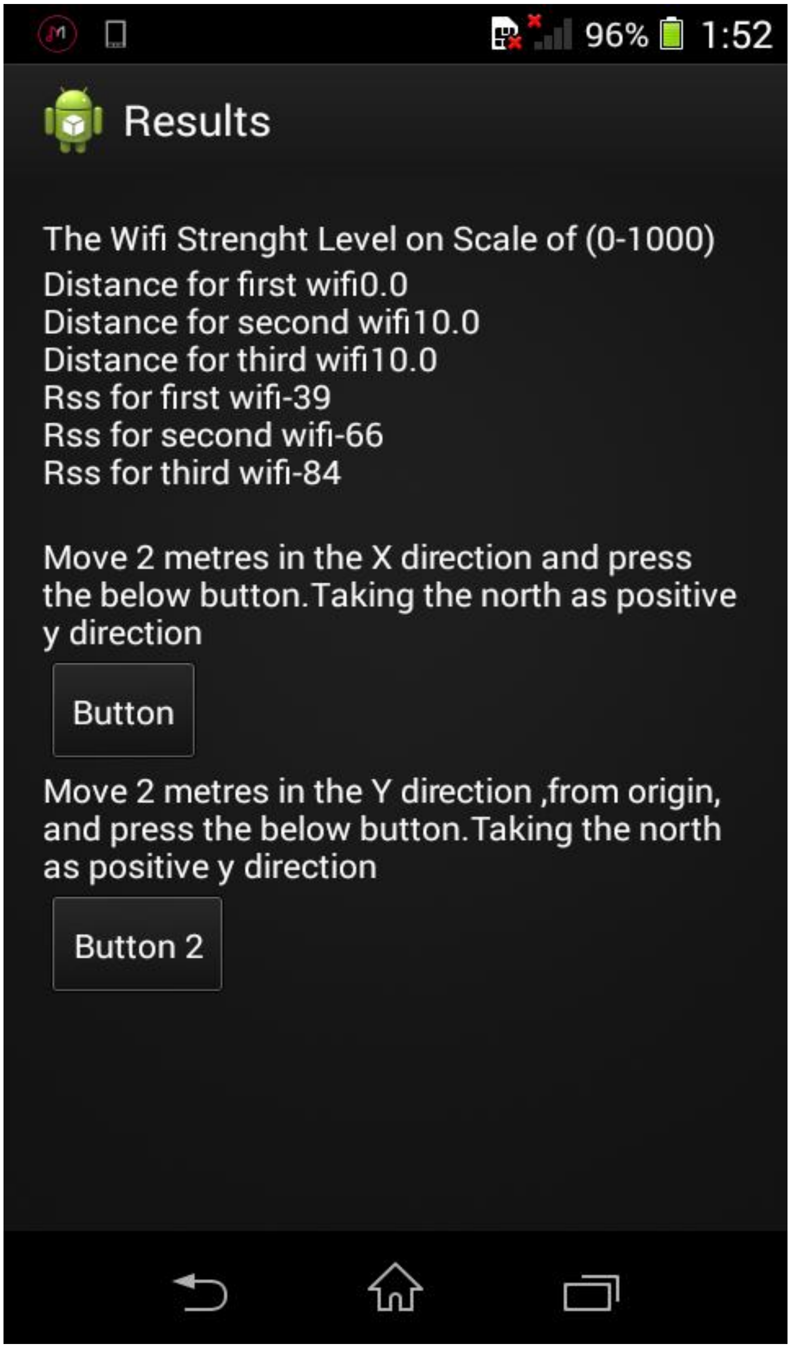


Figure 10: distance showing when moved 2 m

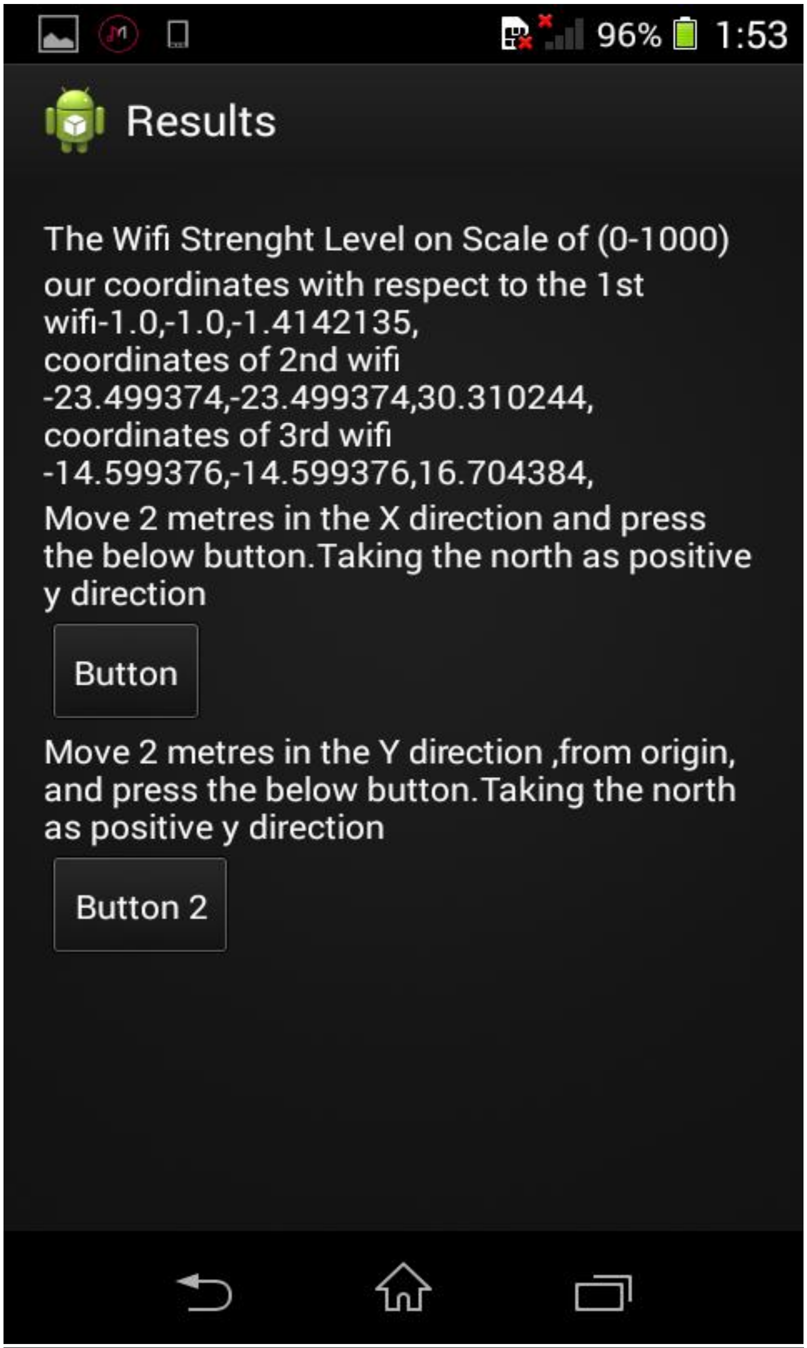


Figure 11:final coordinates when moved 2 m in y direction

8 Limitations

1. Keeping wifi and GPS on all the time might be draining on the battery.
2. Location of Wifi access points may not be known in places like offices and malls where the deployment is done by external parties.
3. Floor plans might not be publically available
4. Security in case of privacy breach might be an issue.
5. The strength of Wi-Fi signal tends to fluctuate easily due to interference and especially so when the receiver is some distance away from the access point(AP).

9 Conclusion

This report presents an efficient trilateration algorithm which estimates the position of a target object, e.g. a mobile device, based on the simultaneous distance measurements from multiple reference points. Solving the nonlinear least squares formulation of trilateration, the proposed algorithm provides an optimal position estimate of the intersection point of N≥n spheres in R^n (n=2 for 2D environments and n=3 for 3D environments), not limited to solving for the intersection points of exact n spheres in R^n. Using standard linear algebra techniques, the used algorithm, though not in the closed form, has low computational complexity and is highly applicable to real-time applications. Without depending on the techniques which tend to be affected by algebraic singularities, such as matrix inversion, the proposed algorithm has high operational robustness.

By introducing more reference points and corresponding distance measurements into the trilateration process will in general reduce the estimation uncertainty. Though targeting the applications in mobile robotics, it is our beliefthat the proposed trilateration algorithm is applicable to any ranging-based object localization tasks in various environments and scenarios.

The final conclusion is that indoor location based on Signal Quality or Signal Strength can work really well in an actual work environment. The mapping of the environment with the Trilateration feature values really established a good working base for the tracking methods to work with. When looking at the precision of the tracking methods they show great potential to be implemented in for example a smart house.

10 Future works

We have completed the calculation of the users location and basic layout of the app. Also we would like to work upon the shortcomings that we came across while dealing this method.after working upon the shortcomings I would like to sell my app to google play store.

11 References

1. “Object tracking through RSSI measurements in wireless sensor networks”, F. Viani, L. Lizzi, P. Rocca, M. Benedetti, M. Donelli, A. Massa. 8th May 2008.

2. “An Indoor Localization Mechanism Using Active RFID Tag”, Guang-yao Jin, Xiao-yiLu, Myong-Soon Park.

3. M. Hedley, D. Humphrey, and P. Ho, “System and Algorithms for Accurate Indoor Tracking Using Low-Cost Hardware,” Proc. IEEE/IOA Position, Location and Navigation Symp., pp. 633-640,May 2008.

4. George Dedes and Andrew G Dempster (2005),“Indoor GPS Positioning: Chal-lenges and Opportunities”.

5. Fluerasu, A., Boiero, G., Ghinamo, G., Lovisolo, P., & Samama, N. (2010). Indoor Positioning Using GPS transmitters : Experimental results. *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, (September), 15-17.

6. Inside.mines.edu/~whereman/papers/Murphy-MS-Thesis-Complete-1992.pdf

7. Merchant, D.C. 1981. Passive Trilateration Positioning Procedure Error Analysis. Griffis Air Force Base, NY: Rome Air Development Center. RADC-TR-81-225 (August).

8. Error Analysis of a Trilateration Network With and Without the Laplace Condition. Ohio State University: Masters Thesis.

9. Mikhail E.M. 1976. Observations and Least Squares. New York: IEP-Dun-Donnelley, Harper and Row.

10. Lawson, C.L., and R.I. Hanson 1974. Solving Least Squares Problems. Englewood Cliffs, NJ: Prentice Hall.

11. M. Hedley, D. Humphrey, and P. Ho, "System and Algorithms for Accurate Indoor Tracking Using Low-Cost Hardware," Proc. IEEE/IOA Position, Location and Navigation Symp., pp. 633-640,May 2008.

12. R. Want, A. Hopper, V. Falc˜ao, and J. Gibbons. The active badge location ACM Transactions on Information Systems (TOIS), 10(1):91–102, 1992.

13. Y.Chen, D. Lymberopoulos, J. Liu, and B. Priyantha. Fm-based indoorlocalization. In Proceedings of the 10th international conference on Mobile systems, applications, and services, pages 169–182. ACM, 2012.

14. Fluerasu, A., Boiero, G., Ghinamo, G., Lovisolo, P., & Samama, N. (2010). Indoor Positioning Using GPS transmitters : Experimental results. *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, (September), 15-17.

15. http://sdiwc.net/digital-library/download.php?id=00000223.pdf.

16. Latiff, L. A., Ali, A., Chia-ching, O., & Fisal, N. (2005). Development of an Indoor GPS-free Self-Positioning System for Mobile Ad Hoe Network (MANET). *Access* (pp. 1062-1067). Ieee.

17. http://www.cs.berkeley.edu/~pabbeel/cs287-fa09/readings/Durrant-Whyte_Bailey_SLAM-tutorial-I.pdf