Indoor Positioning System based on Bluetooth Low Energy

A Degree's Thesis

Submitted to the Faculty of the

Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona

Universitat Politècnica de Catalunya

by

Agustí Corbacho Salas

In partial fulfilment

of the requirements for the degree in

SCIENCE AND TELECOMMUNICATION TECHNOLOGIES ENGINEERING

Advisor: Josep Paradells Aspas

Barcelona, June 2014

Abstract

Bluetooth Low Energy is the new specification of Bluetooth available for all new smartphones. It is a new low power consumption technology aimed to transmit small amount of data. In addition to be able to transmit data, BLE can be used to locate things. *iBeacon* protocol, which uses BLE, is aimed at that objective with special attention in proximity location with BLE beacons. The objective of this project was to implement an indoor positioning system based on Bluetooth Low Energy beacons and the *iBeacon* protocol. This indoor positioning system uses trilateration methods to estimate the position based on distances estimated through modelling the indoor path loss. Several existing techniques and methods are studied and analysed in this project, the ones that presented best results have been implemented in the system. Through an evaluation of the system and different calibration techniques, an accuracy of 1~1.5 meters has been obtained.

Resum

Bluetooth Low Energy és la nova especificació de Bluetooth disponible en els nous models de smartphones. És una nova tecnologia de baix consum amb l'objectiu de transmetre petites quantitats de dades. A més de ser capaç de transmetre dades, BLE també es pot fer servir per localització. Per altra banda, el protocol *iBeacon*, el qual utilitza BLE, té l'objectiu de localitzar coses amb especial atenció a la localització per proximitat amb beacons BLE. L'objectiu d'aquest projecte era implementar un sistema de posicionament en interiors basat en els beacons de Bluetooth Low Energy i el protocol *iBeacon*. Aquest sistema de posicionament en interiors utilitza mètodes de trilateració per estimar la posició basat en l'estimació de distàncies a partir de models de propagació en interiors. En aquest projecte, s'han estudiat i analitzat diferents tècniques i mètodes, i s'han implementat aquelles que han presentat millors resultats. A través d'una avaluació del sistema i diferents tècniques de calibratge, s'ha obtingut una precisió al voltant de 1~1.5 metres.

Resumen

Bluetooth Low Energy es la nueva especificación de Bluetooth disponible en los nuevos modelos de smartphones. Es una nueva tecnología de bajo consume con el objetivo de transmitir pequeñas cantidades de datos. Además de ser capaz de transmitir datos, BLE puede localizar cosas. Por otra banda, el protocolo *iBeacon*, el cual utiliza BLE, tiene el objetivo de localizar cosas a partir de la proximidad con beacons BLE. El objetivo de este proyecto era implementar un sistema de posicionamiento en interiores que se basara en los beacons de Bluetooth Low Energy y el protocolo *iBeacon*. Este sistema de posicionamiento en interiores utiliza métodos de trilateración para estimar la posición basado en la estimación de distancias a partir de modelos de propagación en interiores. En este proyecto se han estudiado y analizado diferentes técnicas y métodos, y se han implementado aquellas que han presentado mejores resultados. A través de una evaluación del sistema y diferentes técnicas de calibración, se ha obtenido una precisión alrededor de 1~1.5 metros.

Acknowledgements

First of all, I would like to thank my advisor Dr. Josep Paradells Aspas for his support, guidance and advice throughout this project.

I would like to thank Joaquim Oller from Telematics Department of UPC for all of the Bluetooth Low Energy knowledge that he provided to me at the beginning of this project.

Finally, I would like to thank all of my family and friends that have shown support along the way. Especially to my girlfriend, who has given to me the enough moral support every day for the completion of this thesis.

Revision history and approval record

Revision	Date	Purpose
0	23/06/2014	Document creation
1	07/07/2014	Document revision 1
2	11/07/2014	Document revision 2

DOCUMENT DISTRIBUTION LIST

Name	e-mail
Agustí Corbacho Salas	aguscsp@gmail.com
Josep Paradells Aspas	josep.paradells@entel.upc.edu

Written by:		Reviewed and approved by:	
Date	11/07/2014	Date	11/07/2014
Name	Agustí Corbacho Salas	Name	Josep Paradells Aspas
Position	Project Author	Position	Project Supervisor

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

telecom BCN

Table of contents

List of Figures

List of Tables:

1. Introduction

Bluetooth Low Energy is the new specification 4.0 of Bluetooth, it is a new technology with low energy consumption aimed at smart home, health, sport and fitness sensor industry. Its features such as low cost and low range has led to the development of indoor positioning systems based on this technology.

1.1. Statement of purpose

The purpose of this project was to study the viability of an Indoor Positioning System based on Bluetooth Low Energy. BLE is a new technology that can be used for many applications including positioning. So, a first approximation of an indoor positioning system based on BLE needs to be studied for analysing its viability in future applications.

So, the goals of the project are described below:

- Get familiar with Bluetooth technology, especially with new specification Bluetooth Low Energy.
- Study and analyse path loss models of indoor propagation focusing on BLE characteristics and signal power behaviour.
- Study the performance of existing positioning algorithms for Wireless Positioning Systems and improve them in terms of accuracy with new methods.
- Create a user-friendly application to obtain and show the coordinates of a device placed in an indoor environment based on distances from target device to BLE devices (beacons).

1.2. Requirements and specifications

The project must satisfy the following requirements:

- The developed solution should be consistent with the low power requirements of BLE devices.
- Distances should be estimated based on BLE features such as RSSI and some extra information.
- The final application should be able to discover, identify and connect to BLE devices available in its proximity.
- The coordinates of a device in an indoor positioning system should be calculated based on estimated distances using mathematical methods.
- The final application should run in a widely developed operating system and should have a friendly-user interface.
- The indoor positioning system should reach an accuracy about 1~2 meters.
- The final application should be stable, running continuously without crashing.

The specifications of the project and the final application are the following ones:

- The application must use Bluetooth Low Energy technology for positioning.
- Application must work in a widely developed operating system that allows the implementation of an indoor positioning system such as iOS, Android or Windows Phone.

1.3. Methods and procedures

This project is an independent project of Telematics Department of UPC, not related with others at the same time, but taking profit of the previous work with BLE and location solutions developed by the research group that hosts the development of this project.

The procedure of the project followed next methods:

- Study BLE technology and its application to positioning.
- Study Path Loss Model existing solution and analyse them in order to choose the best one.
- Study and analyse existing mathematical techniques for positioning.
- Create a new application that locates a device with tools studied before.

1.4. Work Plan

1.4.1. Work packages and milestones

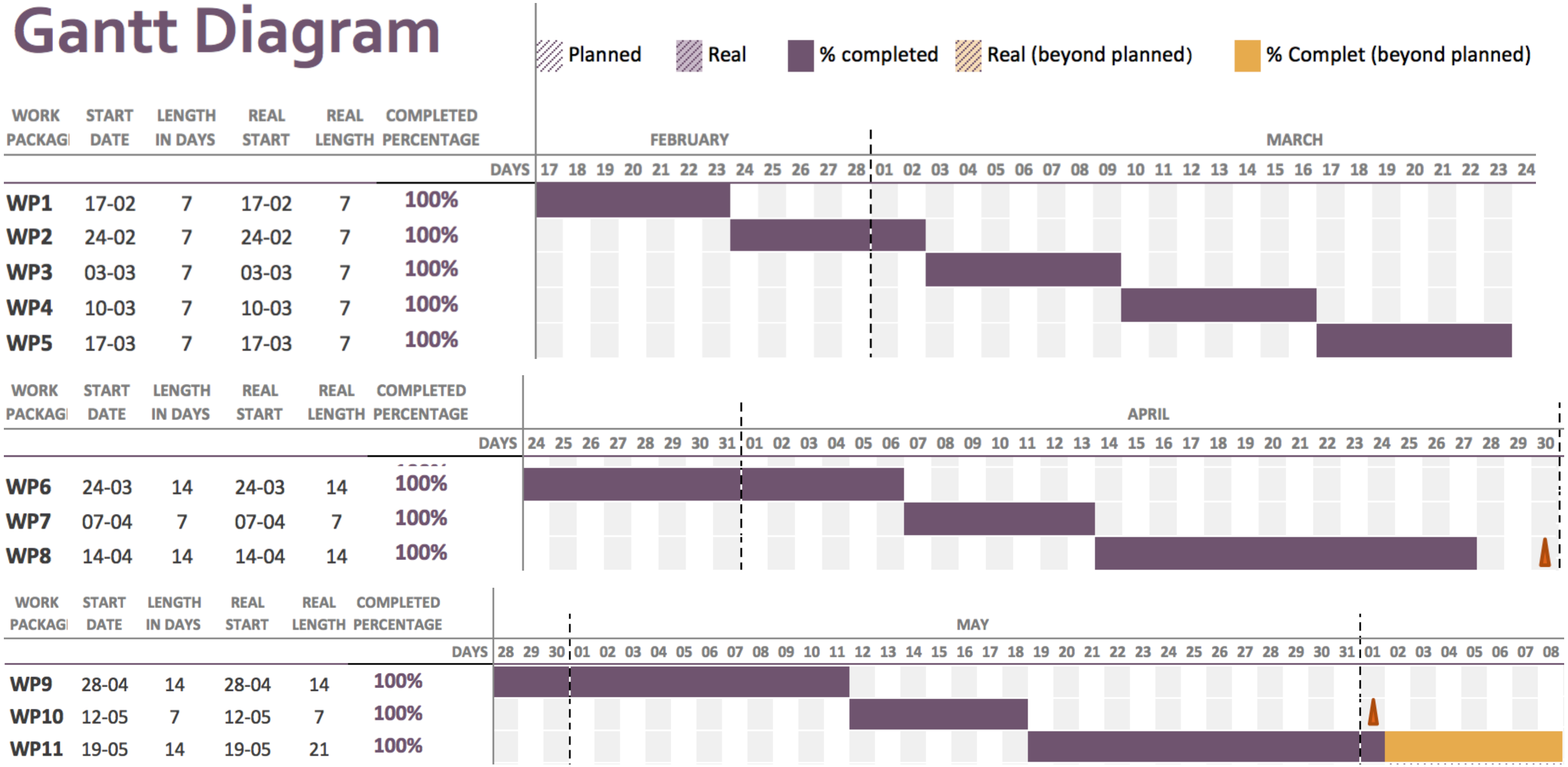
The work plan is divided in some work packages that are summarized below.

WP#	Task#	Short title	Milestone / deliverable	Date (week)
	1	Study of BLE		
1	2	Analyze CC2541DK-MINI		1
	3	Analyze software BTool from TI		
	1	API CC2541DK-MINI from TI		
2	2	API Google and B-SIG for Android		2
	3	API from another third parties		
3	1	Create an application to scan and connect BLE devices		3
	2	Application to read RSSI values	Application for reading RSSI values of BLE	
4	1	Gather information about path loss characterization		4
	2	Create path loss model		
5	1	Create an application to obtain distances from the master to BLE devices	Application for obtaining distances based on RSSI measurements.	5
	2	Test and validate the app and results.		
	1	Document BLE technology		6
6	2	Justify programming language used		
	3	Document path loss characterization		7
	4	Document the completed tests		

7	1	Scenario definition		8
	2	Gather information about existing calculation methods for location		
8	1	Analyze, verifty and improve path loss model with MATLAB.		9-10
9	1	Program location algorithms		11-12
	2	Test and validate the results of the location algorithms		
	3	Define the mathematical model used by the application	Critical Review	
10	1	Document location algorithms used by the app		13
	2	Analyze and document the completed tests		
11	1	Create the final application to obtain the coordinates inside an indoor positioning system.	Final Application	14 -15
12	1	Do some tests in an indoor environment.		16
12	1	Document the application		17
	2	Analyze and document the completed tests		18
	3	Complete the final report template.	Final Report	19

Table 1.1. Work Packages and Milestones.

1.4.2. Time Plan (Gantt Diagram)



WORK PACKAGE	START DATE	LENGTH IN DAYS	REAL START	REAL LENGTH	COMPLETED PERCENTAGE
WP1	17-02	7	17-02	7	100%
WP2	24-02	7	24-02	7	100%
WP3	03-03	7	03-03	7	100%
WP4	10-03	7	10-03	7	100%
WP5	17-03	7	17-03	7	100%

WORK PACKAGE	START DATE	LENGTH IN DAYS	REAL START	REAL LENGTH	COMPLETED PERCENTAGE
WP6	24-03	14	24-03	14	100%
WP7	07-04	7	07-04	7	100%
WP8	14-04	14	14-04	14	100%

WORK PACKAGE	START DATE	LENGTH IN DAYS	REAL START	REAL LENGTH	COMPLETED PERCENTAGE
WP9	28-04	14	28-04	14	100%
WP10	12-05	7	12-05	7	100%
WP11	19-05	14	19-05	21	100%

WORK PACKAGI	START DATE	LENGTH IN DAYS	REAL START	REAL LENGTH	COMPLETED PERCENTAGE
WP12	02-06	7	09-06	14	100%
WP13	09-06	28	23-06	18	100%

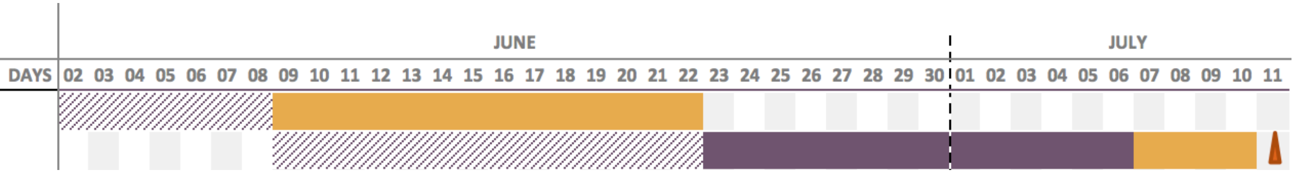


Fig 1.1. Gantt Diagram of the project.

2. State of the art

2.1. Bluetooth Low Energy

Bluetooth wireless technology is the global wireless standard which enables short range wireless communication between devices.

Bluetooth Low Energy (BLE) is the new specification [1], [2] of Bluetooth technology developed by Bluetooth Special Interest Group (SIG). It has been designed as a complementary technology to classic Bluetooth in order to guarantee low power consumption. Although it uses the Bluetooth brand and uses similar technology, BLE should be considered a different technology with different goals and applications.

BLE is designed for transmission of small amount of data and for ultra-low power consumption. It is not thought for maintaining a connection for a long time between devices. It uses the idea of 'Get what you want when you want' and this allows the devices to be awoken only when are asked for data. In this way, devices consume less power.

2.1.1. Architecture

Bluetooth Low Energy devices can be in different operating states and roles depending on its function. Therefore, the possible states are the following:

- **Standby:** Does not transmit or receive packets
- **Advertising:** Broadcasts advertisements in advertising channels
- **Scanning:** Looks for advertisers
- **Initiating:** Initiates connection to advertiser
- **Connection:**
    - Master Role: Communicates with device in the Slave role.
    - Slave Role: Communicates with single device in Master Role.

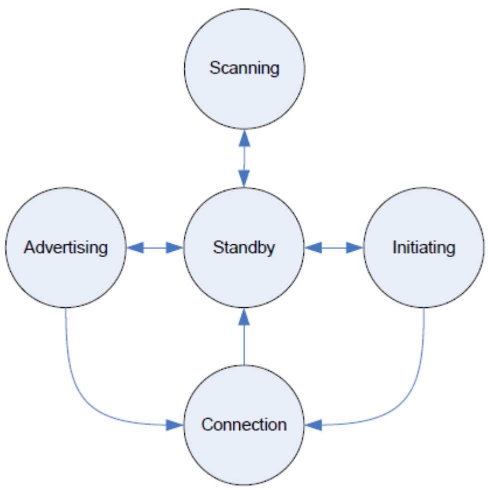


Fig 2.1. Link Layer state machine of BLE

The network topology of BLE is the star type. Master devices can have multiple link layer connections to peripherals (slaves) and simultaneously scan for another devices. On the other hand, a slave can have only one link layer connection to one Master.

Moreover, a peripheral can send advertising events without expecting a connection; it is used to show data to the scanners without the need to maintain a long time connection.

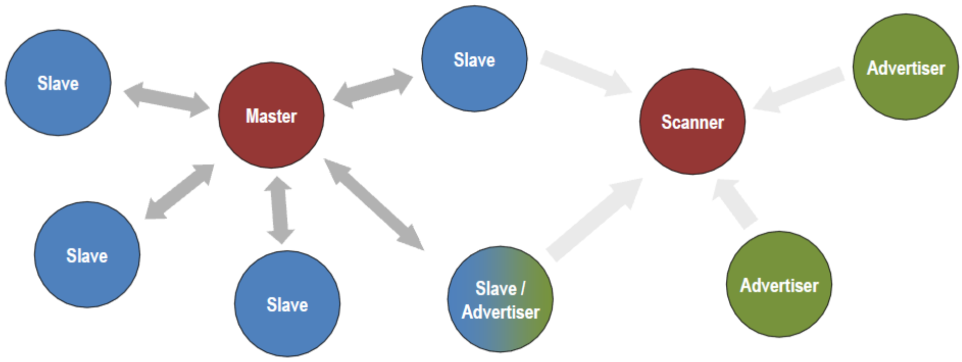


Fig 2.2. BLE network topology.

The BLE protocol stack is split into three basic parts: controller, host and applications. The controller is a physical device that can transmit and receive radio signals and interprets these as packets with information. It comprises Physical Layer, Direct Test Mode, Link Layer and Host Controller Interface.

The Host is a software stack that manages how two or more devices communicate between them. There is no defined upper interface for the host, each operating system or environment has different way of exposing host APIs for developers. It contains the Logical Link Control and Adaptation Protocol, the Security Manager, Attribute protocol (ATT), Generic Attribute Profile (GATT) and Generic Access Profile (GAP).

The Applications use the software stack and therefore the controller, to enable a use-case. The Application Layer defines three types of specifications: characteristic, service and profile.

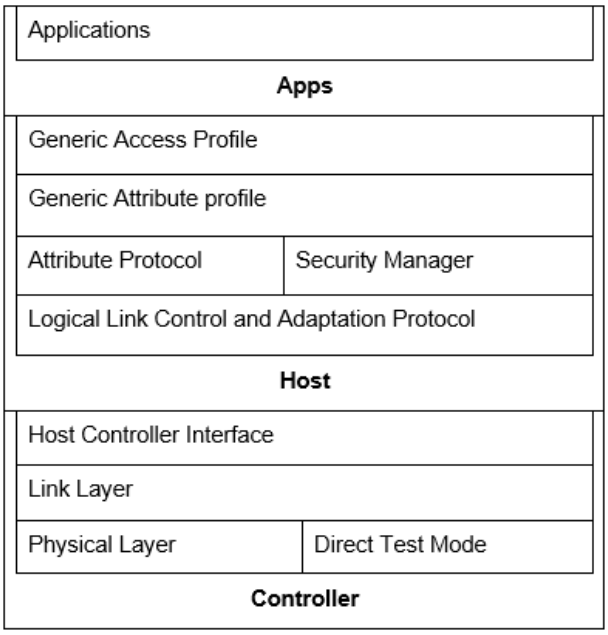


Fig 2.3. BLE protocol stack.

2.1.1.1. Physical Layer

The Physical Layer does the hard work of transmitting and receiving bits using radio waves. The frequency band of BLE is within 2.4 GHz Industrial Scientific Medical (ISM) which is license-free. The BLE band is split up in 40 separate RF channels, separated 2MHz from each other. The centre frequency f_c for each k channel can be calculated by:

$$f_c = 2402 + 2k \tag{2.1}$$

There are 3 channels for advertisement and 37 for transmitting data.

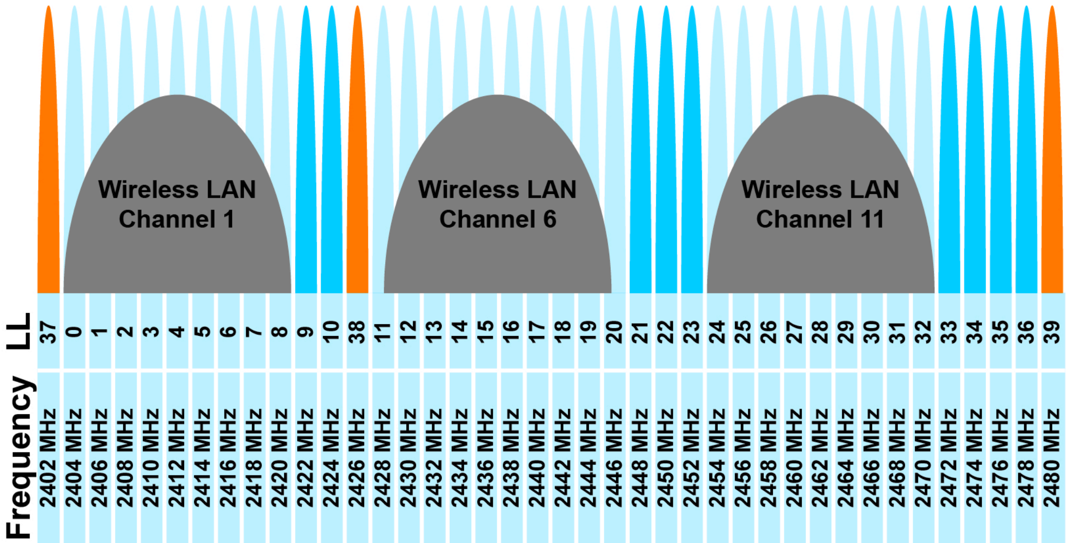


Fig 2.4. BLE radio channels.

Due to the coexistence with other technologies as IEEE 802.11 and ZigBee, the three advertising channels are strategically placed to avoid interferences with these technologies. Moreover, in a connection state mode, BLE uses Frequency Hopping Spread Spectrum (FHSS) technique to reduce interferences and the spectral power density.

BLE radio uses Gaussian frequency-shift keying. The modulation uses two frequencies to identify bit '1' or '0'.

$$\begin{cases} f = fc + \Delta \rightarrow '1' \\ f = fc - \Delta \rightarrow '0' \end{cases} ; \quad where\ \Delta \geq 185KHz \tag{2.2}$$

A Gaussian filter is used to slow the transitions between frequencies and reduces the spectral spreading caused by ISI.

The BLE specification limits the maximum transmit power to +10 dBm and minimum transmit power of -20 dBm. The minimum required receiver sensitivity for BLE is -70 dBm.

2.1.1.2. Link Layer

The Link Layer is responsible for advertising, scanning, and creating and maintaining connections. It is also responsible for packets structure.

A BLE device uses the advertising channels to broadcast data, advertise that they are connectable and discoverable (**Advertiser**), and to scan (**Scanner**) and initiate connections (**Initiator**).

The data channels are used once the connection between the devices is done.

Scanner devices look at advertising channels for advertising packets from other devices. There are four types of advertising: general, directed, non connectable and discoverable.

- **General Advertising:** is the most common. The scanner device can get the advertisement packet or go into a connection.
- **Direct Advertising:** it is used when a device needs to connect quickly to another device. This advertising event must be repeated every 3.75 milliseconds
- **Non connectable Advertising:** it is used for devices that want to broadcast data. The device cannot be discoverable nor in a connection
- **Discoverable Advertising:** it cannot be used to initiate a connection, but it can be scanned for other devices. Scanner device can obtain the advertising data in the scan response. It is used to broadcast data too.

Therefore, a broadcasting device will include data in advertisement packets that are not direct connectable. Broadcasted data can be received by passive and active scanning devices. The problem is that a broadcasting device does not know when a device could need its data.

In order to get the devices connected, a device must use a connectable advertising event. When the initiator receives the advertisement packet, it sends a connect request to the advertiser. Once the advertiser receives the connect request, the devices are connected and can exchange data packets.

The packet structure is the same for advertising and data packets. The bytes are transmitted with the least significant bit first. For example, 0x80 is transmitted as 00000001.

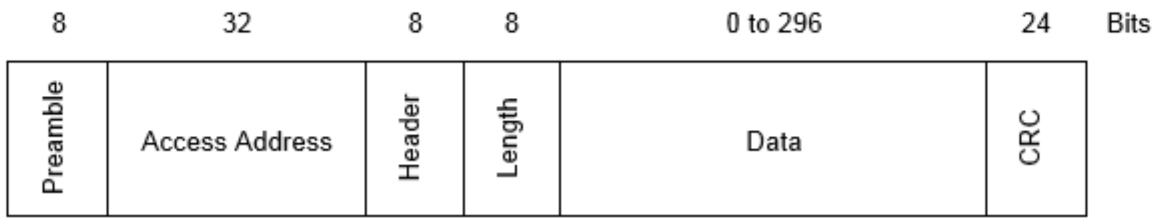


Fig 2.5. BLE packet structure

- **Preamble:** The first eight bits are 01010101 or 10101010.
- **Access Address:** there are two types: advertising access address and data access address. The first one is used for broadcasting data, scanning or initiating connections and is always 0x8E89BED6. The second one is used in a connection and changes in each link layer connection.

  Note: The access address is not the device address. Device addresses are public or private. The public addresses consists in 48-bit address obtained from IEEE Registration authority and has 24 bits assigned by the company and 24 bits of the company id. The private ones are optional for BLE devices and changes frequently.

- **Header:** The content depends on whether the packet is an advertising packet or data packet. For advertising packet the header indicates the advertising packet types.
- **Length:** The number of bits of length is variable depending on the type of the packet.

- **Data (Payload):** It is the real data that is transmitted. It could be advertising data from a broadcasting device, a scan response with some information or simply application data from devices that are connected.
- **CRC:** It is 3-byte cyclic redundancy check. It is computed over the header, length and data. It can detect all odd numbers of errors, and 2-bit and 4-bit errors.

Link Layer can also encrypt data within the payload. That means that nobody can decipher the data because they do not have the shared key used by the devices connected.

Low power consumption is accomplished because of two main characteristics of BLE. The first one is low peak-power consumption, having a high peak-power consumption will burn out the batteries too fast. The second one is based on characteristics of the protocol of BLE:

- **Short packets** avoids constant calibration of the radios maintaining the frequency stable during the transmission and reception.
- **High bit rates** allow the devices to use the radio in short time periods transmitting the information fast. So, the higher the bit rate is, the more efficient the radio will be.
- **Low Duty cycle** keeps the small consumption as small as possible. The BLE device only transmits when is asked for it. Otherwise it is 'sleeping' consuming low power.
- **Low Overhead** of BLE affects also on consumption.

2.1.1.3. The Host/Controller Interface

The Host Controller Interface (HCI) defines the physical connection between the host and the controller through HCI commands. The specification defines several physical interfaces: UART, 3-Wire UART, USB and SDIO.

Some commands are related to device setup, device discovery, connection setup and state, physical links, link information and other events.

2.1.1.4. Logical Link Control and Adaptation Protocol

The L2CAP in BLE defines L2CAP channels and L2CAP signalling commands. The L2CAP provides a protocol multiplexing layer that multiplex BLE in three different fixed channels (generated when there is a connection): Attribute Protocol, Signalling and Security Manager Protocol. The L2CAP signalling is used basically for updating Connection Parameters and for rejecting them. The Connection Parameters are usually modified by the slave. The slave tells the master how often it wants to transmit. The master could also modify these parameters and both can reject them.

The Security Manager Protocol is used for pairing and key distribution. These concepts are related as pairing typically encrypts the link, pairing two devices. And the distributing keys allow faster reconnections between the devices (bonding).

2.1.1.5. The Attribute Protocol

The ATT follows the client-server architecture. The server has data stored in attributes that the client can read and write. The client sends requests to the server and gets responses back with data or notifications. The Attribute Protocol defines four fields of data:

- **Handle:** address of an individual attribute
- **Type:** UUID, defines what the value means.
- **Value:** The actual value of the data attribute.

- **Permissions:** Whether the attribute can be read or write.

The ATT protocol defines six types of messages depending on its action:

- **Request:** Client requests something from the server.
- **Response:** Server sends response to a request form the client.
- **Command:** Clients send a command to the server without expecting a response.
- **Notification:** Server notifies the client of a new attribute data value. (No confirmation)
- **Indication:** Server indicates to the client a new value. Usually expecting confirmation.
- **Confirmation:** Client sends a confirmation to an indication from the server.

2.1.1.6. The Generic Attribute Profile

The GAP is above ATT and defines the type of attributes and how they are used. The attribute data value is exposed in Characteristics that are encapsulated by Services. A Service is an encapsulation of characteristics that defines how these are used. A characteristic is a set of attributes that describe the state of the device. For instance, a BLE device might have a lot of services, each one describing a functionality of the device. Each service might have some characteristics that expose the data using attributes.

Services and Characteristics can be discovered by a master device within a connection with a slave.

2.1.1.7. The Generic Access Profile

The GAP defines the profile roles of a device: broadcaster, observer, peripheral and central. It also defines procedures for discovering, creating bonds, exchange of security information, establishing connections and resolvable private addresses. Moreover, it defines advertising and Scan Response Data formats.

A profile is defined based on the application use-case and includes roles for the client and the server, and the services required by the application.

For example, Proximity Profile has the next three services: Immediate Alert, Tx Power and Link Loss Alert. And each service has its characteristics with attribute definitions.

handle (hex)	handle (dec)	Type (hex)	Type (#DEFINE)	Value (default)	GATT Server Permissions	Notes
0x23	35	0x2800	GATT_PRIMARY_SERVICE_UUID	0x1803 (LINK_LOSS_SERVICE_UUID)	GATT_PERMIT_READ	Start of Link Loss Service
0x24	36	0x2803	GATT_CHARACTER_UUID	0A (read/write permissions) 25 00 (handle 0x0025) 06 2A (UUID 0x2A06)	GATT_PERMIT_READ	Alert Level characteristic declaration
0x25	37	0x2A06	PROXIMITY_ALERT_LEVEL_UUID	0 (PP_ALERT_LEVEL_NO)	GATT_PERMIT_READ | GATT_PERMIT_WRITE	Link Loss Alert Level characteristic value (see defined values in proxperiph.h)
0x26	38	0x2800	GATT_PRIMARY_SERVICE_UUID	0x1802 (IMMEDIATE_ALERT_SERVICE_UUID)	GATT_PERMIT_READ	Start of Immediate Alert Service
0x27	39	0x2803	GATT_CHARACTER_UUID	04 (write permission) 28 00 (handle 0x0028) 06 2A (UUID 0x2A06)	GATT_PERMIT_READ	Alert Level characteristic declaration
0x28	40	0x2A06	PROXIMITY_ALERT_LEVEL_UUID	0 (PP_ALERT_LEVEL_NO)	GATT_PERMIT_WRITE	Path Loss Alert Level characteristic value (see defined values in proxperiph.h)
0x29	41	0x2800	GATT_PRIMARY_SERVICE_UUID	0x1804(TX_PWR_LEVEL_SERVICE_UUID)	GATT_PERMIT_READ	Start of Tx Power Level Service
0x2A	42	0x2803	GATT_CHARACTER_UUID	12 (read permission) 2B 00 (handle 0x002B) 07 2A (UUID 0x2A07)	GATT_PERMIT_READ	Tx Power Level characteristic declaration
0x2B	43	0x2A07	PROXIMITY_TX_PWR_LEVEL_UUID	0	GATT_PERMIT_READ	Tx Power Level characteristic value (dBm)
0x2C	44	0x2902	GATT_CLIENT_CHAR_CFG_UUID	0x0000	GATT_PERMIT_READ | GATT_PERMIT_WRITE	Tx Power Level characteristic configuration

Fig 2.6. Proximity Profile (Services, Characteristics and Attributes)

2.2. System architecture

In this section, a description of the system architectures commonly adopted by some indoor positioning systems is given. In order to locate a device inside an indoor environment, a network of BLE devices is needed. Since BLE devices acting as slaves can only be connected to one master, it makes no sense to maintain a connection between the master device and the slaves. So, the BLE devices are placed at a fixed point advertising all the time, this is what is known as beacon. Beacons are inexpensive small devices and are easy to deploy.

Moreover, if no connections are established between master mobile device and peripherals, no data is interchanged and allows the privacy of the mobile device to be maintained.

Once the devices are defined, the architecture is discussed depending on how the positioning is done. Since the BLE devices are thought of for low consumption, all computations for location are done on a master device or on a server, depending on the architecture.

An example of client/server architecture is studied in [3]. The hierarchal architecture consists of a mobile device which collects the RSSI of the signal of Access Points placed strategically in an indoor environment. This information is given to the server, which does the necessary computation to locate the device and save its location in a database in order to track the movements of the device. The position is passed on to the mobile, which offers the corresponding services or actions to the user.

Another purpose of client/server architecture besides tracking is for positioning technique fingerprinting. In [4], the deployment of the beacons is also strategically placed. A radio map of the indoor environment is done in an offline phase and it is saved in the server (knowing the location of the beacons). It means that the indoor environment is divided in some areas and the RSSI of these areas are stored in the database.

Then, in an online phase, the mobile devices will send the real RSSI values to the server in order to compare with the previous samples and to obtain the estimated position which is sent back to the mobile in order to do the necessary actions.

The deployment of the beacons in fingerprinting is critically since better accuracies can be obtained depending on how the deployment is done. In [5] a deployment algorithm has been developed to improve the position accuracy in the fingerprinting method. They reduce the distance error by approximately 40%. So, accuracy is dependent on beacons density. The higher the number of beacons, the higher the accuracy will be.

The client/server architecture is more complex and expensive, and in case of fingerprinting, always needs an offline phase which complicates the implementation of the system in different distributions of indoor environment.

A common architecture consists only in a mobile device and beacons. It is usually based on triangulation positioning technique. In [6] system, the beacons are also placed strategically knowing its positions. Then, the device calculates the distances from the mobile device to the beacons and also the position based on some positioning technique (usually triangulation).

In this architecture, the mobile device is responsible for all the computations in order to obtain its position and also for offer all the services depending on the estimated position.

2.3. Positioning techniques

This section describes the different positioning techniques, taking into account the characteristics of BLE technology and its useful features for positioning. Therefore, only solutions based on proximity and RSSI measurements are considered. The author has found no studies about BLE positioning, so positioning techniques based on Bluetooth are described instead of BLE. Both use the same parameter for location: Radio Signal Strength Indicator (RSSI).

2.3.1. Trilateration

Trilateration is a positioning technique which uses measured distances to determine the position of a point. The system measures distances from a point to, at least, three references, and forms circles where the intersection of these three will give the location of the point.

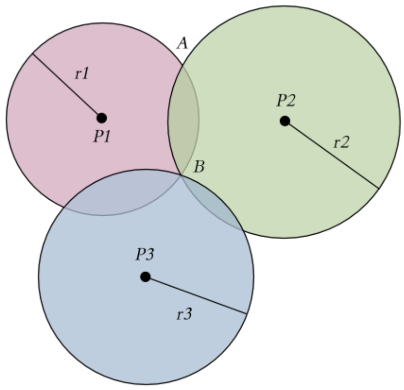


Fig 2.7. Trilateration representation example

There are several mathematical methods to solve the equations of trilateration. In [6] three methods are discussed: least square estimation (LSE), three-border positioning and centroid positioning. The results of the study yields that LSE method gives better results than the others.

In addition, [7] uses the gradient method, minimizing the error between the Euclidean distance and the distance measured. So, there are some mathematical methods to resolve trilateration.

However, resolving equations is not the critical part of positioning. The accuracy of the system is highly determined by the measurement of the distances.

2.3.1.1. Distance measurement

The manner that the BLE uses to measure distances is through RSSI measurements. The RSSI value is included in every advertisement packet in BLE. This allows using different path loss models in order to obtain an accurate distance.

Several studies about Bluetooth v2.1 [8] and [9], have pointed out that RSSI is not a good parameter for location because it depends on the device. Moreover, if the device is not far enough, the obtained RSSI value is 0. So, it is not a trusting parameter for location. Since Bluetooth Low Energy is a new specification of Bluetooth SIG thought for location, these studies have no impact in our project.

In addition, [16] has done a study of power-accuracy in BLE comparing Bluetooth and Bluetooth Low Energy. The result shows that BLE is better for localization since its accuracy is higher. This is because Bluetooth needs a connection between the devices to read the RSSI values. Besides the disadvantages commented in [8] and [9], they say that RSSI values obtained from Bluetooth are not very accurate. Using BLE, we can adjust the advertisement interval to control the number of samples of RSSI. Otherwise, Bluetooth reads several RSSI every time period (which is greater than BLE interval).

The signal power decreases mostly because of the distances, but there are also other effects that more or less affect the signal power depending on whether the devices are in line-of-sight (LOS) or not.

- **Fading:** is the time variation of the received signal. It depends on the environment and the movement of the devices.
- **Multipath:** it causes the signal to arrive in different ways to the device due to reflections, diffraction and scattering, depending on the indoor distribution such as furniture, walls, objects, etc.

So, we can say that RSSI is a random process and its path loss needs to be modelled. Moreover, in cases where there is no LOS, the distance measurement is more complicated and needs reconfiguration algorithms to get good estimations of distance.

In [10], [12], [11], [13], the log-normal model is adopted to measure distances, introducing some improvements. The log-normal model is the best model which fits the indoor propagation since it has two parameters to model the problems of multipath (n) and fading which is a random variable (X_g):

$$RSSI = RSSI_0 + 10 \cdot n \cdot log_{10}\left(\frac{d}{d_0}\right) + X_g \tag{2.3}$$

Where $RSSI_0$ value is the signal power measured at distance d_0.

A study on the distribution of RSSI is done in [10] in order to characterize the path loss model. Once they have the path loss modelled, a tuning on the parameters of log-normal model is done in order to adapt the RSSI-distance ratio. Then, a combination of two models is used for distance estimation depending on a RSSI threshold. An accuracy of 2.3 meters in LOS and 2.9 meters in non-LOS are obtained.

The n parameter is really important in distance estimation; depending on its value we can improve the accuracy. In [11], a filter algorithm has been presented in order to update and estimate the n parameter. However, the authors point out that log-normal model is suitable when there are not many multipath components. So, in some indoor environments this model cannot be applied.

In [12], there are also methods to estimate the n parameter in order to improve accuracy. They have used two methods for n estimation. The first one was to calculate the path loss exponent using a number of received powers and corresponding distances. This method needs an offline phase to tune the parameter. The second one was to directly update the parameters using the gradient decent technique. This method can be done in online phase directly.

On the other hand, [13] uses a Recursive Least Square based on Received Signal Strength (RLS-RSS) algorithm to reduce the error in distance estimation. A high improvement in accuracy is obtained with this method. According to the authors, its method reduces the multipath effect in distance estimation.

There is also an indoor propagation model [14] recommended by the ITU that follows the following equation:

$$L = 20\log f + N\log d + P_f(n) - 28 \quad (2.4)$$

Where in this equation L refers to the total path loss, f the frequency of transmission (MHz), d the distance in meters, N the distance power loss coefficient (given by the ITU depending on the environment), n the number of floors between transmitter and receiver and $P_f(n)$ the floor loss penetration factor also calculated by expressions from ITU Propagation Model. No studies using the ITU model have been found in the research.

2.3.2. Fingerprinting

Fingerprinting is a positioning method that uses RSSI values in order to estimate the position of a device. Firstly, the offline phase of the method consists in divide the whole indoor room into small areas with respective coordinate positions (X,Y). Then, RSSI samples in each small area are stored in a database. Secondly, in the online phase, gathering some RSSI samples at unknown positions does the positioning. These RSSI values are compared to the values stored in the database and an algorithm matches the actual value with the fingerprints obtained in the offline phase and locate the device with the coordinates that best fit the fingerprints.

The implementation of fingerprinting with Bluetooth is useful since trilateration is not a good technique due to problems with reading RSSI ([8], [9]). However, with BLE seems to be possible and it is the scope of this project.

Some projects like [15] have obtained accuracy of around 2 meters using fingerprinting based on Bluetooth.

2.3.3. Proximity based method

Proximity based method consists in locating a device depending on its distance from another device or beacon.

Proximity Profile [1] is a new profile introduced by BLE that enables proximity monitoring between two devices. It consists in causing an immediate alert whether the connection is dropped or the path loss increases so that it is bigger than a threshold.

Commercial solutions as *Estimote* or *PayPal* also use this technique to locate a user. When the user is closer to the beacon, they push notifications in order to offer some actions to the client. For example, in a museum, when you are closer to a picture or sculpture, a notification with information about this sculpture is pushed to your mobile phone. On the other hand, *PayPal* uses proximity to guarantee that the user is near the point of sale (POS) and can proceed with the purchase.

This method is not useful for the scope of the project since it does not give the exact position of the device.

2.4. Commercial Solutions

In this section several commercial systems and hardware are commented. Since only a few indoor positioning systems based on BLE has been commercialized, beacon hardware solutions are commented too.

2.4.1. iBeacon

iBeacon is an Apple Trademark for location and proximity detection technology. *iBeacon* works on BLE. A mobile device can detect beacons nearby. Applications can know when the device proximity to one *iBeacon* is near or far, and push notifications depending on the use-case. Even though *iBeacon* is originally for iOS, SDK for Android has recently released to bring *iBeacon* technology to Android.

The main advantage of *iBeacon* is that the application can run in the background and only shows up when another *iBeacon* is detected. This gives a good user experience.

2.4.2. Estimote

An *Estimote* is a beacon that broadcast radio signals based on Bluetooth Low Energy. *Estimote* is based on *iBeacon* technology and so are also thought to work with proximities. However, there are some methods in the API that allow the location of mobile devices and measure distances indoors, but the accuracy is still poor because of the changes in indoor environment.

So far, only development kits are being sold, this means that they are not totally commercializing the *estimotes*, because they are in a development phase.

Estimote Inc. provides a Demo app for doing some tests with *estimotes* such as proximity estimation, characteristics modification and beacon localization.

Moreover, some projects about positioning have been developed with *estimotes*. [18] presents its trilateration project in *YouTube* based on *Estimotes*. The results are acceptable when target device is completely in line of sight with the beacons.

A room management system is presented in [19], they based its solution on *estimotes* and it works both with proximity and positioning. They locate people inside or outside a room (proximity) or tells the user where it is located and where the room is that he is looking for (positioning).

2.4.3. *PayPal* Beacon

PayPal Beacion is a BLE device that requires a connection with a smartphone in order to authenticate it by a remote *PayPal* server. *Paypal* Beacon is more complex since the transmitter also needs Wi-Fi technology. It is not thought of for location, but for making purchases automatically in the point of sales without any need of cards, cash, and signatures…

2.4.4. Indoo.rs

Indoo.rs is an indoor positioning system commercial solution based on BLE. The company offers a software solution called Measurement Tool in order to map the target building (floors, distribution) for indoor location, do fingerprints and add additional information like walls, zones or departments in order to improve the accuracy of the system. Then, they gives you an SDK in order to modify them according with your necessities.

3. Project development

This section analyses theoretical outline and introduces the methods followed during project development of each topic.

3.1. Bluetooth Low Energy

Bluetooth Low Energy (also known as Bluetooth 4.0 or Bluetooth Smart) is the new specification of Bluetooth technology developed by Bluetooth Special Interest Group (SIG).

BLE was the technology chosen to do this project, so as to create an indoor positioning system (IPS). Other technologies such as ZigBee, Wi-Fi or Bluetooth are present in some indoor positioning systems:

- **Bluetooth Low Energy:** is a wireless personal area network (PAN) defined in the new specification of Bluetooth technology. It is the low-cost and low-power solution of Bluetooth aimed at fitness, healthcare, security and home entertainment industries.
- **ZigBee:** is a specification of high level communication protocols based on an IEEE 802.15 standard, used to create personal area networks (PAN) built from small, low-power digital radios. Its network topology is mesh and permits the transmission of data through nodes of a network, reaching long distances but with a small data rate. ZigBee is a low-cost technology.
- **Wi-Fi:** is a local area network (LAN) technology that allows communication between electronic devices over a wireless signal. The IEEE 802.11 standard defines Wi-Fi technology.
- **Bluetooth:** is a wireless technology standard which enables short range wireless communication between fixed and mobile devices and builds wireless personal areas (PAN)

These technologies use logos to validate and certify the features of each technology in a specific product.

Fig 3.1. Zigbee, Bluetooth, Bluetooth 4.0 and Wi-Fi logos

A comparison of the technologies mentioned are done in the following table from a point of view of indoor positioning:

	ZigBee	Wi-Fi	Bluetooth	BLE
Network topology	Star, cluster, or mesh	Ad-hoc, or Star	Scatternet	Star-bus
Frequency Band	868 MHz (Europe) 915 MHz (North America) 2.4 GHz	2.4/5 GHz	2.4 GHz	2.4 GHz
Data Rate	250 Kbps	11/54 Mbps	1 to 3 Mbps	1 Mbps
Range	10 to 100 m	Up to 100 m	Up to 10 m	Up to 40 m
Power Consumption	Very low	High	Low	Very low
Battery Life	Multiple years	Multiple hours	Multiple weeks	Multiple months
Cost	Low	High	Medium	Low
Infrastructure	To be deployed	Existing Wi-Fi nodes	To be deployed	To be deployed
Smartphones	Not supported	Supported	Supported	Supported
Developed for positioning	No	No	No	Yes
Accuracy	3-5 m	5-10 m	2-5 m	1-2 m
Typical applications	Industrial control and monitoring, sensor networks.	WLAN, broadband connections.	Inter-device data transfer (i.e. cable replacement).	Sensors, positioning, peripherals.

Table 3.1. Comparison of Zigbee, Bluetooth, BLE and Wi-Fi technologies ([20] and [21])

The technology to be used in this project should guarantee:

- **Low-power consumption:** Nowadays, it is a very important requirement in all projects, low-power consumption means having cheaper infrastructures.
- **Easy deployment:** the indoor positioning system of this project is thought to be deployed in an easy and fast manner.
- **Indoor Positioning:** The technology to be chosen must be specially designed for positioning
- **Supported by smartphones:** Smartphones are present in our lives everyday. So, the solution of this project should work perfectly with a smartphone.

- **Long battery life:** It would be desirable that the battery life of devices of the technology chosen was so long that the battery replacement was every long periods of time (e.g. several months).

All these features and requirements fit very well in Bluetooth Low Energy technology as can be seen in the table. ZigBee has the inconvenience of not being supported by smartphones nowadays. Moreover, the technology has been there for some years and so far, it has not been highly adopted for commercial solutions. On the other hand, even though Wi-Fi has a deployed infrastructure over a lot of indoor places, it has a high-power consumption that makes it difficult to maintain.

Bluetooth Low Energy is the best solution for indoor positioning, improving Bluetooth features in this field. BLE provides the optimal balance between infrastructure, reliability, robustness and accuracy.

The frequency band where BLE works is 2.4 GHz. Bluetooth has always used frequency hopping spread spectrum (FHSS) for avoid interferences of other Bluetooth devices or other technologies. Moreover, BLE advertising channels avoids Wi-Fi channels. So, there is no interference with Wi-Fi using advertising mode.

A complete study of BLE from a point of view of positioning has been carried before to start designing the positioning system. So, some commercial solutions have been analysed and are discussed in the next sections.

3.1.1. CC2541 DK-MINI (TI)

Bluetooth Low Energy has been studied through CC2541DK-MINI Texas Instrument's development kit. A study of the characteristics of BLE was necessary before starting on project development, because it decided how the application would work.

The CC2541DK-MINI Texas Instrument's development kit provides a working reference design for software development of BLE applications. The kit contains a CC2540 USB Dongle that acts as BLE master. It can be connected to a Windows PC's USB port acting as a serial port. The kit also has CC2540 "Keyfob", which is a device that acts as BLE slave. The chip is inside a plastic case operating on a single coin cell battery. It also contains two LED, an accelerometer, a buzzer and two buttons. Finally, the kit has a CC Debugger with mini USB cable, converter board, and a 10-pin connector cables used to flash the software onto both the USB dongle and the "Keyfob". It can be used for debugging software using IAR Embedded Workbench.

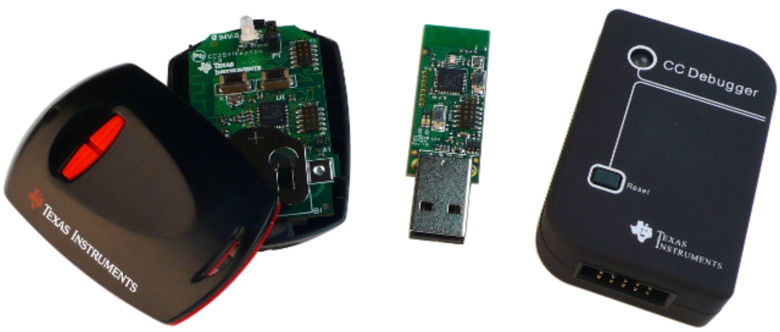


Fig 3.2. CC2541DK-MINI hardware

CC2540 system-on-chip has a high-performance and low-power 8051 microcontroller with a in-system-programmable flash memory of 128/256 KBytes, a SRMA memory of 8 Kbytes, 21 GPIO ports, and other powerful features.

Texas Instruments provides “BTool” (Windows PC application), which is a tool for testing and for verifying BLE stack, and developed custom applications. It has been used in this project to see the features of BLE and study the opportunities of this technology in indoor positioning field.

The next conclusions have been obtained due to an overview of BLE through TI’s kit. Thus, it has paid attention to how the data is interchanged in BLE in order to see how the features of BLE would be exploited for indoor positioning.

BLE has two manners to interchange data: with a connection link or advertisement.

On the one hand, when a master and a slave make a connection between them, the master usually acts as a client that asks for data to the server, making use of GATT protocol. In GATT protocol, the data are structured in characteristics, services and profiles. The master device (client) can use read and write operations to move data to the slave (server). The client can also request to be informed by the server with notify and indicate operations in order to push data to the client from the server.

However, Bluetooth profiles are definitions of possible use-case applications for devices. Bluetooth low energy has adopted several profiles so far but right now, none are designed for indoor navigation or positioning. There is one profile that could work on this project in case it was based on proximity. The proximity profile defines the behaviour when a device moves away from a peer device so that the connection is dropped or the path loss increases above a fixed level, causing an immediate alert to the user. Nevertheless, it has low degrees of freedom for a positioning application. So, a custom GATT profile adapted to positioning features should be created in order to have the possibility of having a positioning system in this project.

Moreover, BLE slaves can only maintain one connection link to one master. So, a connection-based solution wouldn’t be valid for application’s purpose because it is designed to work simultaneously with a lot of users.

On the other hand, advertisement can be done in different types (as described in 2.2.1). The four types of advertisement could work well in this project, as only the RSSI value and some extra data is needed and can be extracted from all advertising modes.

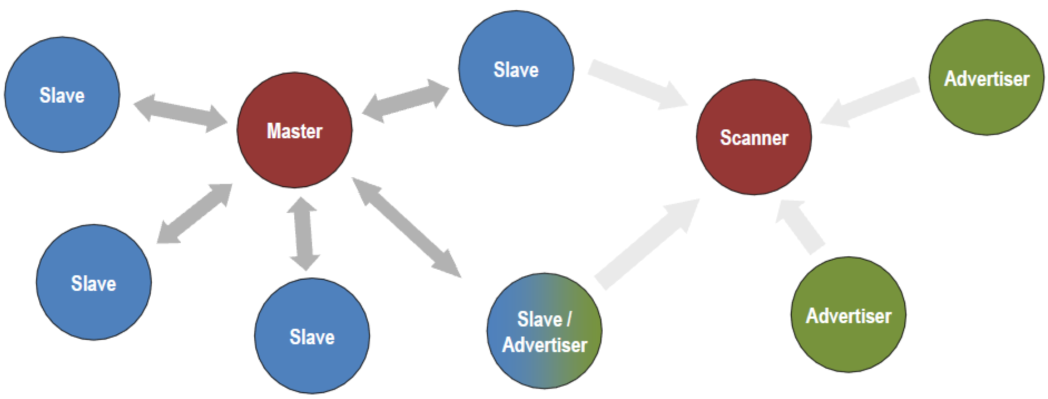


Fig 3.3. BLE network topology: connection link and advertisement

CC2540 BLE slave device has the disadvantage of needing a fully re-programming of the chip with the CC Debugger using IAR Embedded Workbench in order to adapt the advertising data and BLE behaviour to the one required by the project. So, this limitation of the chip was a determinant in looking for other solutions for indoor positioning, in order to facilitate the development of the project.

3.1.2. *Estimote* beacons. The *iBeacon* protocol.

Estimote beacons Developer Preview Kit contains three *estimote* beacons based on *iBeacon* concept. Each beacon is built around the Nordic Semiconductor nRF51822 with a 32-bit ARM Cortex CPU with 256KB of flash and 16KB of RAM with a built-in 2.4GHz radio supporting Bluetooth Low Energy. The chip also includes a temperature sensor and an accelerometer that have been recently functional.

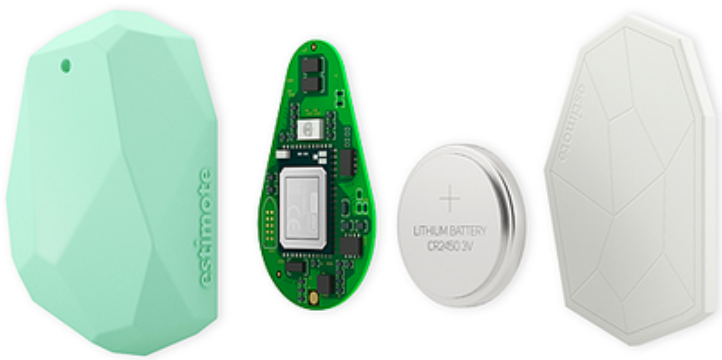


Fig 3.4. Estimote beacon's hardware

Estimotes follow *iBeacon* concept, that is part of the CoreLocation framework in iOS. *iBeacon* is similar to geofencing (virtual barriers or boundaries in some applications based on geopositioning) but uses BLE signal power instead of GPS to know one's proximity to the beacon. It has the advantage of detecting proximity when the target is near the beacon instead of having to rely on a fixed GPS location. So, the proximity is relative to an identified beacon placed intentionally in one place.

BLE communication consists of two main parts (as seen in 2.2.1): advertising and connecting. *Estimotes* are based on advertising but can be connected in order to tune some parameters as broadcasting power, advertising interval and some identifiers.

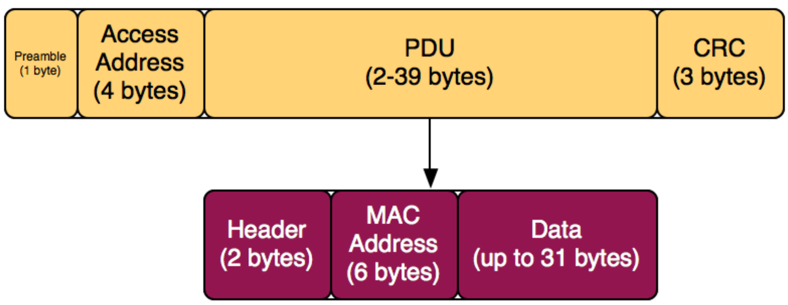


Fig 3.5. Advertisement packet

The advertising packet of BLE consists on:

- **Preamble** (1 byte)
- **Access Address** (4 bytes)
    - For advertisement channels the access address is always 0x8E89BED6.
- **PDU** (2-39 bytes)
    - Header: size of payload and its type (2 bytes).
    - MAC Address: always present after the header (6 bytes).
    - Data: up to 31 bytes of advertising data.
- **CRC** (3 bytes)

Beacons use advertising channel for communicate the data of *iBeacon* protocol. The total amount of data is 30 bytes, which fits very well the 31 bytes limit.

An example of the data of an advertising packet from a beacon would be:

0201061AFF4C000215 B9407F30F5F8466EAFF925556B57FE6D 0049 000A C5

Table 3.2. Advertising packet example

These are split into:

- ***iBeacon* prefix**: is fixed by the protocol and gives the information that a beacon follows *iBeacon* protocol (9 bytes). The bytes of *iBeacon* prefix are 02 01 06 1A FF 4C 00 02 15.
- **UUID identifier**: is an identifier to distinguish the beacons of one company from others (16 bytes). For example, all the beacons belonging to a brand would have the same UUID. The *Estimote*'s UUID is B9 40 7F 30 F5 F8 46 6E AF F9 25 55 6B 57 FE 6D.
- **Major**: is also an identifier for a group of beacons (2 bytes). For example, all the beacons placed in a store from one company would have the same major. The beacons of other store from the same company shares the UUID but with different major. In this case, the major is 00 49.
- **Minor**: is used to distinguish the beacons from one group of beacons (2 bytes). For example, the beacons placed in a store share the same major and are identified by the minor number.
- **Measured Power**: is the last byte of data advertised and is the 2's complement power signal level measured one meter away from a beacon. This value is prefixed by *iBeacon* protocol in *estimotes* so it can't be changed in order to calibrate the value. In this case, 0xC5 is 197. So, the measured power at 1m would be 256-197 = -59 dBm.
- **RSSI:** the radio signal strength indicator is implicit in every BLE advertising packet but not inside PDU data.

Analysing this data, *estimote* and *iBeacon* concept are seem to be thought for indoor positioning. Even though they are based on proximity, they could work for positioning because there are some information as Measured Power, RSSI and some extra information (major/minor) that can help to locate devices.

Although beacons are thought of for the advertisement mode, they can maintain a connection in order to tune some parameters:

- **Advertising interval:** is the period when an advertising packet is broadcasted. It can be configured from 50ms to 2000ms. This parameter is critical in battery life so it has to be configured according to the requirements of the application.
- **Broadcasting power:** is the physical power of the transmitted signal. Its values go from -30dBm to +4dBm. This parameter is also critical in battery life and has to be configured according to the requirements of the application.
- **Major/Minor/UUID:** These values seen before can be configured in order to distinguish all the beacons or put some data for application's purpose.

iBeacon and Bluetooth Low Energy, are both good solutions for positioning that present the following advantages:

- BLE is a good solution for an indoor environment, having a range up to 50 meters approximately and good accuracy.
- Beacons are easily identifiable in an indoor environment (not only MAC Address), with its parameters.
- *iBeacon* allows data transmission related to positioning and is easy to extract.
- *Estimote* minimizes power consumption using Bluetooth Low Energy technology.
- *iBeacon* is a new protocol presented by Apple and is earning the support of the industry.
- The accuracy can be improved depending on the situation, by adjusting the advertising interval and power transmission.
- *Estimote* is secure for user privacy because no connection or pairing is needed between the beacon and user device.
- Beacons are also low-cost; the deployment is not as expensive as other technologies (e.g. IEEE 802.11).
- There is no interference with IEEE 802.11 technology in BLE's advertisement channels.

Even thought *estimote* is not the unique solution for *iBeacon* protocol (as seen in 2.4), it was the available one for the department to use in this project.

3.1.2.1. *Estimote* SDK

Estimote SDK is available for Android and iOS. It allows working with *iBeacon* technology using *Estimote* beacons.

Android 4.3 (API Level 18) introduced built-in platform support for Bluetooth Low Energy providing APIs to discover devices, query for services, and read/write characteristics. In contrast to iOS, which supports the central and peripheral/broadcaster role, Android 4.3 only supports central role, which is enough for purpose of the project as the programming part is in the central device.

Even though the project can be done in iOS, Android has been chosen due to facilities to the author. The knowledge of Android programming language before doing this thesis was basic, but also the hardware available was based on Android. So, Android has been the programming language chosen to do the project.

Estimote SDK for Android is based on Android API for Bluetooth and allows:

- Beacon ranging in foreground (scan beacons and filter them).
- Beacon monitoring in background (scan beacons filtered by some values).
- Beacon characteristic reading and writing.

So, it achieves the application requirements that are:

- Scanning BLE devices.
- Identify BLE beacons.
- Get advertisement data.

Products that use *iBeacon* protocol can be easily identified by its logo.

Fig 3.6. iBeacon and Estimote logos.

3.2. Indoor Path Loss Model

The main objective of this part is to obtain accurate distances in order to use it in trilateration positioning techniques. A study of indoor environment is done to better adapt the path loss model. Moreover, a developed application that allows capturing beacon's RSSI and distances is described below.

3.2.1. RSSI

The Radio Signal Strength Indicator (RSSI) is the measured power (in dB/dBm) at the receiver. The RSSI can be used to estimate distance as its value decreases over it. Several studies point out that RSSI signals follow a log-normal distribution in indoor environment. So, path loss models will take in consideration this feature.

Apple's documentation points out that *iBeacon* (and thus, BLE) is not a standard for indoor positioning, but for proximity positioning. The ranges of RSSI values will show where the device is located depending on the proximity of the beacon. Normally, the three zones are immediate, near and far.

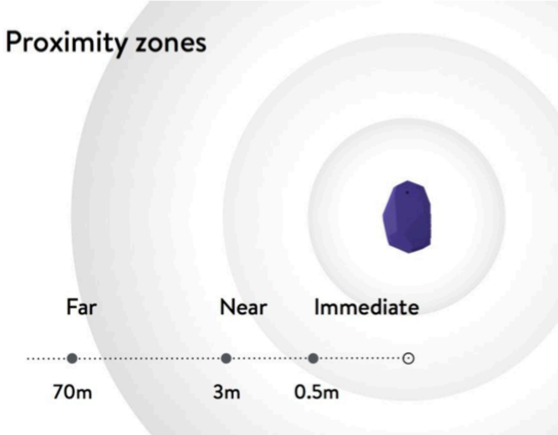


Fig 3.7. Proximity zones in iBeacon protocol

The truth is that RSSI fluctuates rapidly over time and this makes it difficult to estimate its position with high accuracy but it can be a good estimation that can be improved over other positioning techniques.

iBeacon protocol also gives some parameters to adjust distance estimation like measured power at 1 meter. *Estimotes* allow the tuning of broadcast power and advertising interval. The higher the broadcast power is, the higher the range and accuracy will be. On the other hand, the smaller the advertising interval is, the higher the accuracy and stable the RSSI will be.

So, even though RSSI is not the perfect parameter for positioning, it has shown good solutions so far with good accuracies and all commercial solutions are based on it.

3.2.2. Indoor Path Loss

Indoor Positioning System is a solution to locate objects or people inside a building, normally based on received signal power indicator (RSSI). Path loss is the attenuation of signal power of a signal as it propagates through space. It may be due to some of following effects:

- **Free space loss:** assuming a complete line of sight (LOS) between transmitter and receiver, free space loss is the degradation of the signal due to distance.
- **Fading losses:** it refers to the time variation of received signal power caused by changes in the transmission channel. In an outdoor environment, fading is due to atmospheric conditions and others. Whereas in an indoor environment, fading can be due to furniture distribution, human interferences or movement of the devices.
- **Multipath losses (also known as fast fading):** multipath consists of the contribution of a large number of scattered rays. The main signal is reflected due to objects changing its phase, attenuation and delay. The result of the sum of all multipath components can degrade the signal. Multipath is caused by reflection, diffraction and scattering.
    - **Reflection:** is the change in direction of an electromagnetic wave when it impacts upon an object. The reflected waves may interfere constructively or destructively at the receiver. It occurs when the object dimensions are greater than wavelength of the signal.
    - **Diffraction:** it occurs when an object obstructs a wave where its wavelength is similar to the size of the diffracting object. The secondary waves resulting from the obstructing surface are present throughout the space, bending the wave signals around the obstacle.

- **Scattering:** scattering is a general physical process where some forms of radiation are forced to deviate from a straight trajectory by one or more paths due to localized non-uniformities in the medium through which they pass. It occurs when the medium is smaller than the wavelength of the signal.

- **Refraction:** is a change in the direction of an electromagnetic wave due to changes in the medium where signals travel.
- **Noise and interferences:** noise and interferences are unwanted signals within the medium that distort original signal at the receiver. It may be avoided increasing transmit power.
- **Atmospheric absorption:** is due to atmospheric elements. It is usually not present in indoor environments.

The main attenuations suffered by an electromagnetic wave in an indoor environment are due to fading and multipath. These phenomenons change in each scenario because they depends on some factors such as furniture distribution, human body interferences or placement of the walls. Attenuations based on atmospherics phenomenons are rejected, so inside a building do not make sense.

The indoor path loss should be modelled in every situation in order to have good distance estimation, knowing that attenuation depends on it. There are two ways of modelling a path loss: empirically or theoretically.

- **Theoretical models:** they are based on principles of physics studying radio wave propagation phenomenon.
- **Empirical models:** they are based on measurements, taking into account all environmental influences in the path loss for every scenario.

Empirical models seem to be the most appropriate because it models all attenuations suffered by an electromagnetic wave inside a building and can be particularized in each scenario based on previous measurements. On the other hand, theoretical models are more complex to be implemented, as they need huge databases of all environments's features in order to particularize for each scenario.

In this project, two empirical models have been analysed before designing a positioning technique based on distances. The model that will have better distance accuracy, will be the one used in the positioning system.

3.2.2.1. Log-distance Path Loss Model

This model is based on that RSSI follows a log-normal distribution over distance and it is applied inside a building or densely populated areas.

So, the path loss model is expressed as:

$$PL(dB) = P_{TX}(dBm) - P_{RX}(dBm) = PL_0(dB) + 10 \cdot n \cdot log_{10}\left(\frac{d}{d_0}\right) + Xg(dB); \tag{3.1}$$

So, the total path loss PL is the difference between transmitted power and received power. And it is expressed as function of the path loss (PL_0) at reference distance d_0. The multipath phenomenon is represented by n, which is known as the path loss exponent. On the other hand, Xg represents the fading and multipath in indoor scenario. Xg is a normal random variable with zero mean and its value is the standard deviation in dB.

Applying some transformations, log-distance path loss model can be expressed in function of RSSI:

$$RSSI = RSSI_0 - 10 \cdot n \cdot log_{10}\left(\frac{d}{d_0}\right) - Xg; \quad (3.2)$$

Where $RSSI_0$ is the received signal power at d_0 distance from the transmitter. The path loss exponent n and the standard deviation Xg values will be defined empirically through some measurements in the indoor scenario.

Finally, distance can be obtained from the following expression:

$$d = d_0 \cdot 10^{\left(\frac{RSSI_0 - RSSI - Xg}{10 \cdot n}\right)}; \quad (3.3)$$

Path loss exponent n may vary depending on the environment and the frequency.

Following table shows some of these values:

Indoor environment	**n**
Free Space	2
Urban area	2.7 to 3.5
In building line-of-sight	1.6 to 1.8
Obstructed in building	4 to 6
Office	2 to 3

Table 3.3. n path loss exponent typical values

3.2.2.2. ITU Indoor Propagation Model

ITU indoor propagation model also estimates the path loss inside a room or a closed area inside a building.

$$L(dB) = 20 \log f(MHz) + N \log d + P_f(n) - 28 \quad (3.4)$$

Where L refers to the total path loss, f the frequency of transmission (MHz), d the distance in meters, N the distance power loss coefficient (given by the ITU depending on the environment), n the number of floors between transmitter and receiver and Pf(n) the floor loss penetration factor also calculated by expressions from ITU Propagation Model.

This model is applicable for frequencies from 900 MHz to 5.2 GHz and 1 to 3 floors.

The distance power loss coefficient depends on the frequency and environment. Some values recommended by the ITU are represented below.

Frequency	Office area	Commercial area
1.8 GHz	30	22
2.4 GHz	30	-
5.2 GHz	31	-

Table 3.4. Power loss coefficient typical values

On the other hand, floor loss penetration factor Pf(n) is also defined by ITU tables depending on the number of floors n:

Frequency	Number of floors	Office area	Commercial area
1.8 GHz	n	15+4(n-1)	6+3(n-1)
2.4 GHz	-	14	-
5.2 GHz	-	16 (1 floor)	-

Table 3.5. Floor loss penetration factor typical values

However, these values are recommendations and two parameters will need to be estimated in order to improve accuracy on distance estimation.

3.2.2.3. *Estimote*'s Path Loss Model

Estimote's SDK for Android has its own method to compute distance given a RSSI value. The method ***computeAccuracy()*** in class Utils from package com.*estimote*.sdk calculates distance following its own path loss model, which is not public.

The model is based on empirical experiments in indoor environments; distance is obtained following curve fitting that models indoor propagation.

This method is studied and compared with other path loss model in evaluation section. It has the disadvantage of not being configurable and opaque to the user. The formula used by *Estimote* Inc. cannot be adapted for each indoor environment.

3.2.3. BLEScan RSSI

'BLEScan RSSI' is a developed Android application for helping to model indoor path loss.

Android 4.3 (API Level 18) introduced built-in platform support for Bluetooth Low Energy providing APIs to discover devices query for services, and read/write characteristics. In contrast to iOS, which supports the central and peripheral/broadcaster role, Android 4.3 only supports central role, which is enough for purpose of the project as the programming part is in the central device.

The application is based on an Android application demo from Android Developers portal for discovering BLE devices. So, 'BLEScan RSSI' scans the medium for discovering BLE devices that are advertising and it also shows the available ones in rows with its name, MAC Address and RSSI.

Demo application has been modified introducing functionalities for saving RSSI measurements at a distance previously set. When the measurements are completely done, they can be saved in a comma separated value (CSV) file in order to be processed and analysed by other programs.

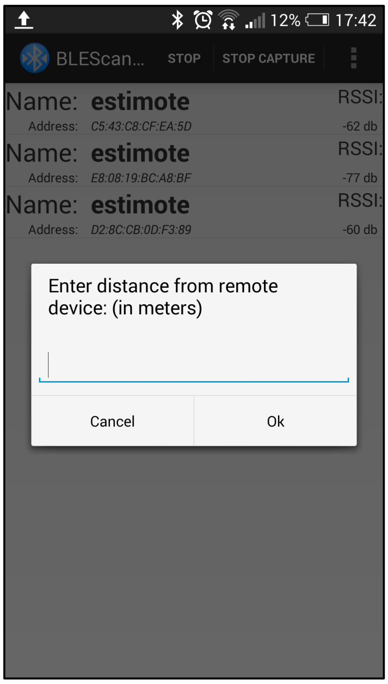
The menu of the application is:

- **SCAN/STOP:** SCAN starts the device discovery and shows the devices with its current RSSI in rows. STOP stops the device discovery showing the devices discovered with its last RSSI value measured.
- **START/STOP CAPTURE:** When START CAPTURE is pressed, a dialog box appears in the display asking for distance between transmitter and receiver. Then, the RSSI values captured are sequently saved in a local variable.
- **Get .csv file:** Generates a CSV file with all RSSI values captured at distances previously set in the application.

The following images show application GUI and some of its functions:

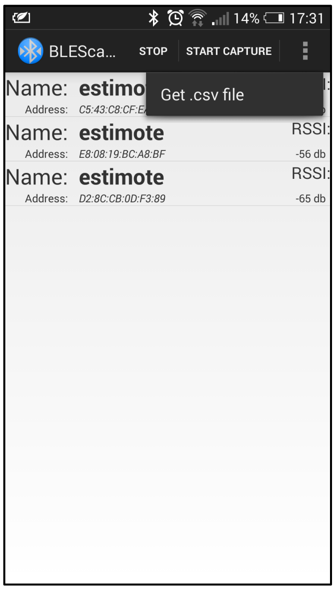
It is the fist view/activity of the application. When application starts to run, it starts to scan for discovering BLE devices and shows its current RSSI that is updated during the scanning time.

Fig 3.8. BLEScan RSSI functionality #1.



When START CAPTURE is pressed, this dialog box is shown on the application. It asks for a distance in meters to store RSSI values associated to that distance. When user presses STOP CAPTURE, it stops storing data.

Fig 3.9. BLEScan RSSI functionality #2.



The whole menu is shown in this picture. START/STOP for scanning devices, START CAPTURE/STOP CAPTURE for storing RSSI values and GET .CSV FILE that generates a .csv file in order to be processed by other program such as MATLAB.

Fig 3.10. BLEScan RSSI functionality #3.

3.2.3.1. Path Loss Model MATLAB script

MATLAB is a high-level language and interactive environment for numerical computation, visualization, and programming.

It has been used to analyse and process the data captured by ‘BLEScan RSSI’ application. ‘Path Loss Model’ script has been programmed in order to study two empirical path loss models: log-distance model and ITU indoor model.

The script follows the following instructions:

- Store data in vectors. Column 1 of the vector is the real distance introduced by the user in 'BLEScan RSSI' application. The rest of columns are RSSI values captured at this distance.
- The distribution of RSSI values is analysed in a histogram and compared to a log-normal probability plot.
- The RSSI values are plotted over distance to see the propagation losses through distance.
- For each vector/distance, the mean and deviation of RSSI values is computed and stored in an array.
- Then, the parameters of each path-loss model are compared graphically and computed in order to minimize the mean square error (MSE) of all distance measurements.

3.3. Indoor Positioning System

This section is a description of the techniques used in the final application for indoor positioning system. These techniques are based on trilateration method, where position is calculated through distances estimated over RSSI measurements.

Trilateration is a positioning technique, which uses measured distances to determine the position of a point. The system measures distances from the point to (at least) three references and forms circles or spheres where the intersection of these three gives the location of the point.

There are several mathematical methods to resolve trilateration equations. In order to solve the positioning problem, first step is linearize the equations of the circles or spheres with its radius as the distance from a transmitter to the receiver. So, distance equation would be:

$$(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 = r_i{}^2 \tag{3.5}$$

Where (x,y,z) is the target point, (x_i,y_i,z_i) is the position where beacon is located, and r_i is the distance from the beacon to the target device.

The system follows solutions presented in [17]. The solutions for solving the nonlinear system generated by trilateration equations are commented and discussed in the thesis. The two with best results have been chosen in order to integrate its algorithmic process in the application.

The linear system would be:

$$\boldsymbol{Ax} = \boldsymbol{b} \tag{3.6}$$

Where,

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

$$b_{n1} = \frac{1}{2}\left(r_1{}^2 - r_n{}^2 + d_{n1}{}^2\right); \tag{3.7}$$

- r_i is the distance from beacon i to target device.
- d_{n1} is the distance from beacon n to beacon 1.

Besides this method has not good accuracy, mathematical method is needed in order to solve trilateration system with better results.

These methods of trilateration need minimum 4 beacons to work in 3D dimension. With 3 beacons, trilateration works in 2D.

3.3.1. Mathematical techniques

The mathematical methods studied and analysed in this project were linear least squares (LSQ) and non-linear least squares (NLS). These methods are compared and used for positioning in evaluation section.

3.3.1.1. Linear Least Squares Method

LSQ consists in minimize the sum of the square of the residual errors of the linear system. So, the function to be minimized is

$$S = (\boldsymbol{b} - \boldsymbol{Ax})^{\boldsymbol{T}}(\boldsymbol{b} - \boldsymbol{Ax}); \tag{3.8}$$

That leads to the following solution:

$$\boldsymbol{x} = (\boldsymbol{A^T A})^{-1}\boldsymbol{A^T b}; \tag{3.9}$$

$\boldsymbol{A^T A}$ will determine if the method is optimal for positioning or not. If $\boldsymbol{A^T A}$ is nonsingular and well-conditioned, then the solution presented above is used. In case $\boldsymbol{A^T A}$ was singular and badly conditioned, then a Singular Value Decomposition (SVD) is needed and following solution is used:

$$\boldsymbol{x} = \boldsymbol{A}'\boldsymbol{b}; \tag{3.10}$$

where $\boldsymbol{A}'$ is the pseudo inverse of $\boldsymbol{A}$.

This method is widely used in positioning systems based on trilateration. The accuracy can even be improved by a weighted matrix $\boldsymbol{W}$**.** The LSQ has different weights according to the accuracy of each measurement, choosing a good weighted matrix, the position accuracy can be improved. But weighted matrix has not been studied in this project, so first LSQ solution is enough for this thesis.

3.3.1.2. Non Linear Least Squares Method

NLSQ consists in minimize the sum of the squares of the errors. So, function to be minimized is the square error of the distances:

$$F(x,y,z) = \sum_{i=1}^{n}(\hat{r}_i - r_i)^2 = \sum_{i=1}^{n} f_i(x,y,z)^2 \,, \tag{3.11}$$

$$f_i(x,y,z) = \hat{r}_i - r_i = \sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} - r_i \,. \tag{3.12}$$

So, deriving F(x,y,z):

$$\boldsymbol{g} = 2\mathbf{J}^T\boldsymbol{f} \tag{3.13}$$

Where,

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

- $\boldsymbol{J}$ is the jacobian matrix.

Thus, R gives the device position (x,y,z) after k iterations:

$$\boldsymbol{R}_{\{k+1\}} = \boldsymbol{R}_{\{k\}} - (\boldsymbol{J}_{\{k\}}{}^{\boldsymbol{T}}\boldsymbol{J}_{\{k\}})^{-1}\boldsymbol{J}_{\{k\}}{}^{\boldsymbol{T}}\boldsymbol{f}_{\{k\}} \tag{3.14}$$

Where $R_{\{1\}}$ is the initial guess of the position and after some iterations the error is minimized until the iteration converges.

3.3.2. Calibration

Calibration is the offline part in the indoor positioning systems. Indoor channel changes rapidly over the time and needs to be manually calibrated for obtaining the best results. Thus, path loss model needs to be optimally calibrated through RSSI measurements in order to estimate distances with high accuracy.

Three techniques have been used in this thesis in order to improve accuracy on the results. They are compared in evaluation section.

3.3.2.1. Point calibration

Point calibration consists in calibrate an indoor environment or a room, having one point as reference. The device is placed at one point known by the application, and computes its distance to the beacons (also at known points). With the RSSI measurements at the reference point and the distances known by the device, the application can calibrates the path loss parameters in order to estimate any distance inside the room.

This technique has the advantage that needs short time of manually calibration, as only one point is needed for doing the calibration. The data to be stored, the path loss parameters, is small so parameters can be stored in *estimote* identifiers in order to broadcast it to new devices that would enter in that place.

On the other hand, this method has the disadvantage that calibration is weak because only one point is calibrating all the indoor environment which changes rapidly over the time and distance.

3.3.2.2. Environment calibration

This type of calibration requires high time consumption in the offline phase. Environment calibration consists in calibrate the environment over distance for each beacon. In this method, the path loss is calibrated at each d distance from the beacon.

‘BLEScan RSSI’ can be a tool to do this kind of calibration. The RSSI measurements are collected at each d distance and then, an overall solution for the path loss model that minimizes the mean square error is assigned to that beacon. So, with the calibration of all the beacons, indoor environment is calibrated depending on beacon position and the distance from the device to the beacon.

This technique has the advantage of being apparently a good solution so that environment is calibrated over a lot of positions inside a room. Moreover, data can be stored in beacons to broadcast it to new devices that enter in that place.

On the other hand, it has the disadvantage of being high timely consuming so that a lot of measures should be done for each new beacon placed inside an indoor environment.

3.3.2.3. Point calibration based on proximity

This method shares similitudes with point calibration technique. Point calibration based on proximity is based on *iBeacon* proximity features. *iBeacon* is not a solution planned for positioning coordinates of a device inside a room, but for proximity to the beacons.

With this solution, the environment is calibrated at one different point for each beacon placed in a room. So, each beacon has different path loss configurations depending on position of the device. For example, when the device is at a point near to beacon 1, the path loss model is calibrated at that point for every beacon. So, when the device is at an unknown point near to beacon 1, the path loss model calibrated at the nearest beacon will be used in order to estimate distances.

Point calibration based on proximity has the advantage of reducing time consumption in comparison with environment calibration because it calibrates at one point per beacon instead of several distances. Moreover, it seems to be more accurate than point calibration because it takes into account more points inside the room to be calibrated.

On the other hand, using this method, a server-based solution would be necessary to store path loss configurations for each beacon because there will be n^2 path loss configurations with an indoor environment for n beacons.

3.3.3. BLE Positioning

‘BLE Positioning’ is the final application of the project programmed in Android. It has been totally programmed using two libraries:

- ***Estimote* SDK:** for the integration and management of Bluetooth Low Energy in Android.
- **EJML:** is a linear algebra library for manipulating dense matrices.

The java project FinalApp is structured in two packages:

- **com.finalapp.trilateration:** is the mathematical computation part of the project. It contains classes for path loss and trilateration.
    - **BeaconCoord:** is the description of a Beacon. It stores the coordinates, the Beacon (*estimote* SDK) object assigned, the RSSI values measured and the path loss configuration.
    - **BeaconCoordList:** contains an ArrayList of BeaconCoord objects.
    - **MathMethod:** abstract class for mathematical techniques for trilateration.
    - **LinearLeastSquares:** LSQ for positioning is programmed in this class.
    - **NonLinearLeastSquares:** NLS for positioning is programmed in this class.

  - **PathLossModel:** abstract class for path loss models.
  - **LogDistance:** path loss modelled as log-distance is programmed in this class.

- **com.finalapp.blepositioning:** it contains the android activities and some helping classes for the functionality of the application.
  - **MainActivity:** is the main activity of the Android project. It contains all the functionalities of the application and integrates all the classes of the project.
  - **DrawView:** is a View for drawing points in the layout of the application.
  - **Calibration:** class for computation the calibration of path loss for beacons.
  - **CalibratePowerActivity:** activity for calibration of the measured power at 1m for each beacon.
  - **Point:** a class that represents a point with its coordinates.
  - **ReferenceCoordSystem:** it transforms the coordinate system used by the user to the display coordinates in order to show the points on the display.

Fig 3.11. BLEPositioning application logo

The GUI of the application is simple, it contains a menu with all the functionalities of the app and a big display for showing an image such as a plan or a drawing, reference points for beacon placement, position of the target device, and interested points drew by the user.

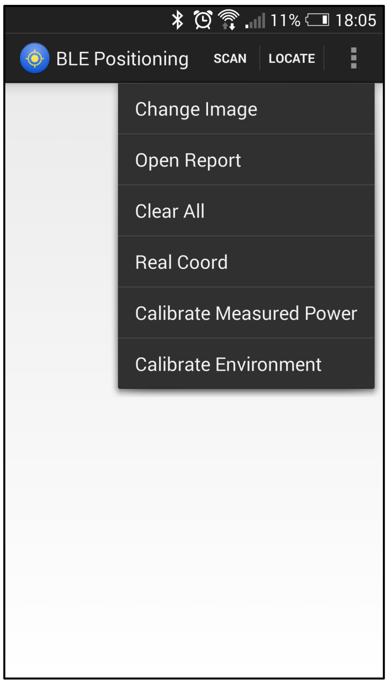
The menu of the application and its function is described below:

- **SCAN:** starts device discovery of the beacons detected by the device. Also asks for enable Bluetooth in case it is not activated in the device.
- **LOCATE:** starts positioning algorithm for locate the device once the beacons are placed.
- **On Touch Screen:** it opens a dialog box asking for placing a beacon with its coordinates and show it on the screen.
- **Change Image:** change the background image to the one choosed by the user from its file explorer.
- **Open Report:** opens a file with all configurations and results reported.
- **Clear All:** clears all configurations of the application. Restarts the application.
- **Real Coord:** a dialog box is open asking for coordinates in order to draw it on the display.
- **Calibrate Measured Power:** opens a new activity that scans beacons for calibrating its measured power at 1m away from them.
- **Calibrate Environment:** it depends on application configuration.
  - **Point Calibration:** it starts a new activity asking for a point and calibration of the beacons referred to that point.
  - **Point Calibration based on proximity:** it opens a dialog box asking for a point to be calibrated depending on beacons proximity.

RSSI is measured each 200 ms during 15 s for positioning, which means that approximately 75 RSSI samples are captured for locating a device.

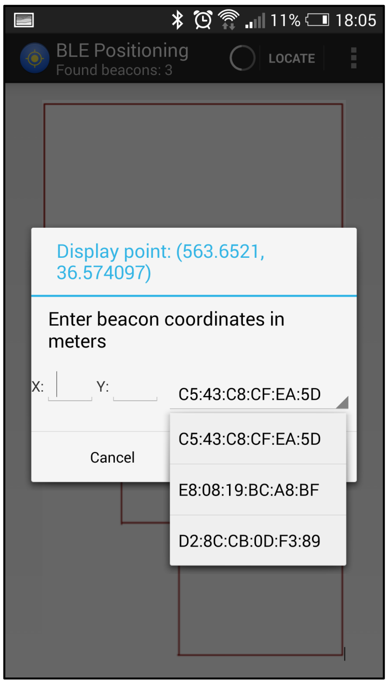
This application has been used for the evaluation of BLE as positioning technology and the algorithms programmed in this project.

Application's functionalities are commented in the following images.



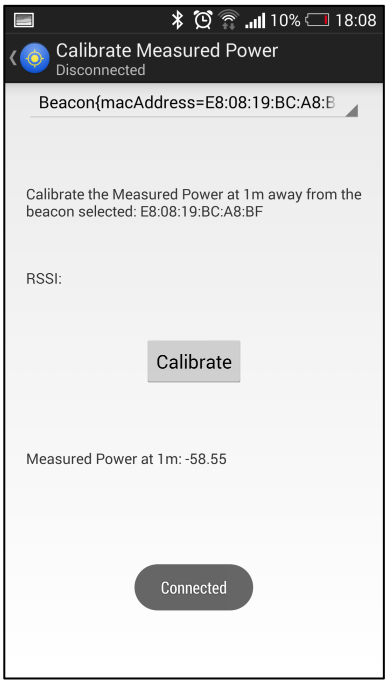
The whole menu is shown in this image. The application starts with a blank layer that can be changed by a custom image with 'Change Image' option.

Fig 3.12. BLE Positioning functionality #1



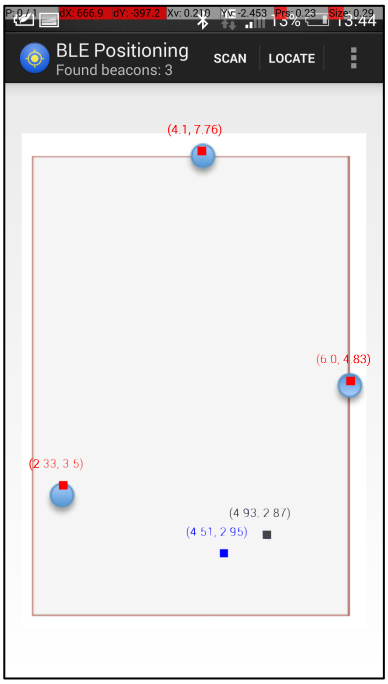
Once SCAN is pressed and devices are discovered, beacons can be set at its position touching the screen. So, a coordinate is associated with selected beacon.

Fig 3.13. BLE Positioning functionality #2



Calibrate Measured Power activity allows the calibration of the RSSI measured at 1 meter away from the beacons. The calibration is done when a beacon is selected from the spinner and the Calibrate button is pressed.

Fig 3.14. BLE Positioning functionality #3



This image is a sample of the indoor positioning system working. When SCAN is pressed, RSSI values of the position is stored in a variable inside the program. Then, when LOCATE is pressed, the positioning algorithm computes device position based on RSSI values measured previously, and shows its position in blue.

Fig 3.15. BLE Positioning functionality #4

4. Results

This section presents and discusses the results of the experiments carried out during project development. The measures and experiments have been done with a HTC One (M7) Android phone.

4.1. Scenarios

It has been used a total of three scenarios in this project. They are commented below including their results.

4.1.1. Living room

The first scenario was a living room of 5.76 x 4 meters approximately (23 m^2). The furniture distribution was normal for a living room with tables, sofa, and furniture distributed along the walls and no object limiting beacons signal coverage.

In addition, there were some objects between target device and beacons that could have introduced some multipath component but it was taken into account in the path loss model. In addition, objects were small enough to not block the signal from target device to the beacons.

The placement of the beacons was quite randomly, beacons were placed on walls taking into account coverage and distance estimation. They were placed separately in order to have a beacon relatively near wherever target device would have been inside the room.

This room was used for an indoor environment study for determining path loss model and also for positioning results.

Fig 4.1. Living room plan and photos

4.1.2. Office

The second scenario was in the office C3101 in Telematics department of UPC. There were tables and PCs distributed over the room and a big column in the middle that could have interfered between target device and beacon signal.

The multipath propagation was more present in this room as big column interferes in all signals wherever beacons were placed. Table and PCs didn't introduce as much multipath as the column.

This room was used only for path loss study and comparison between logdistance and ITU indoor model. Beacons were placed strategically in positions of the room for the study of indoor propagation.

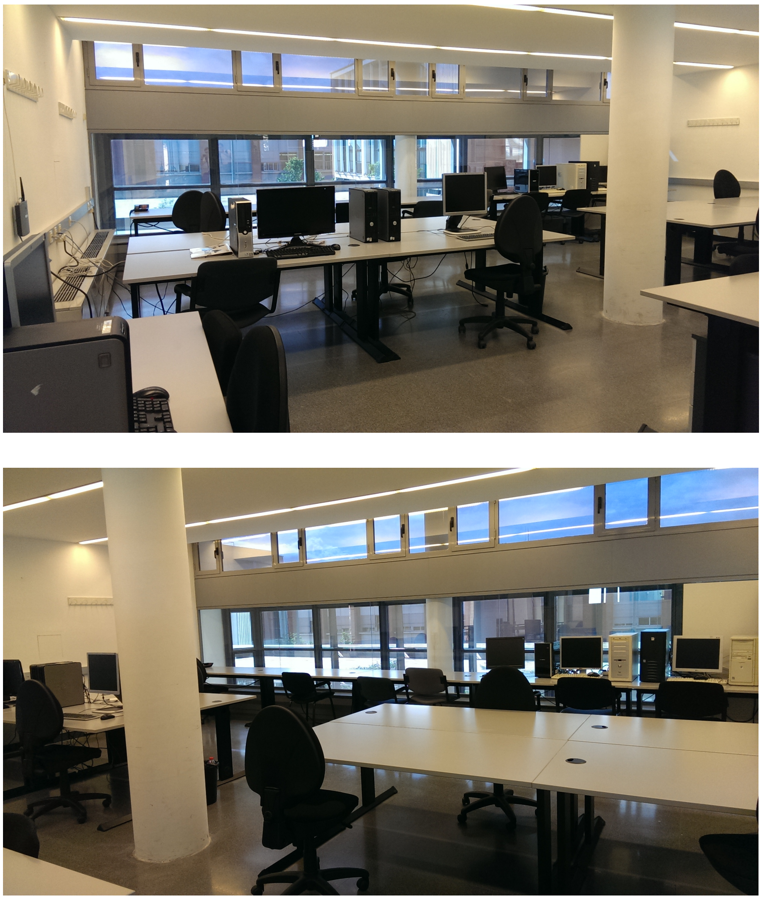


Fig 4.2. Office photos

4.1.3. Workshop

The third scenario was an upholstery workshop of 60 m^2 approximately. The furniture or auxiliary material such as tables, machines or shelves were along the room.

There was no element blocking beacons signal in the middle of the room but in comparison to other scenarios, this has human interference because measures were done during a workday.

The placement of the beacons was randomly as living room scenario, beacons were placed on walls taking into account coverage and distance estimation.

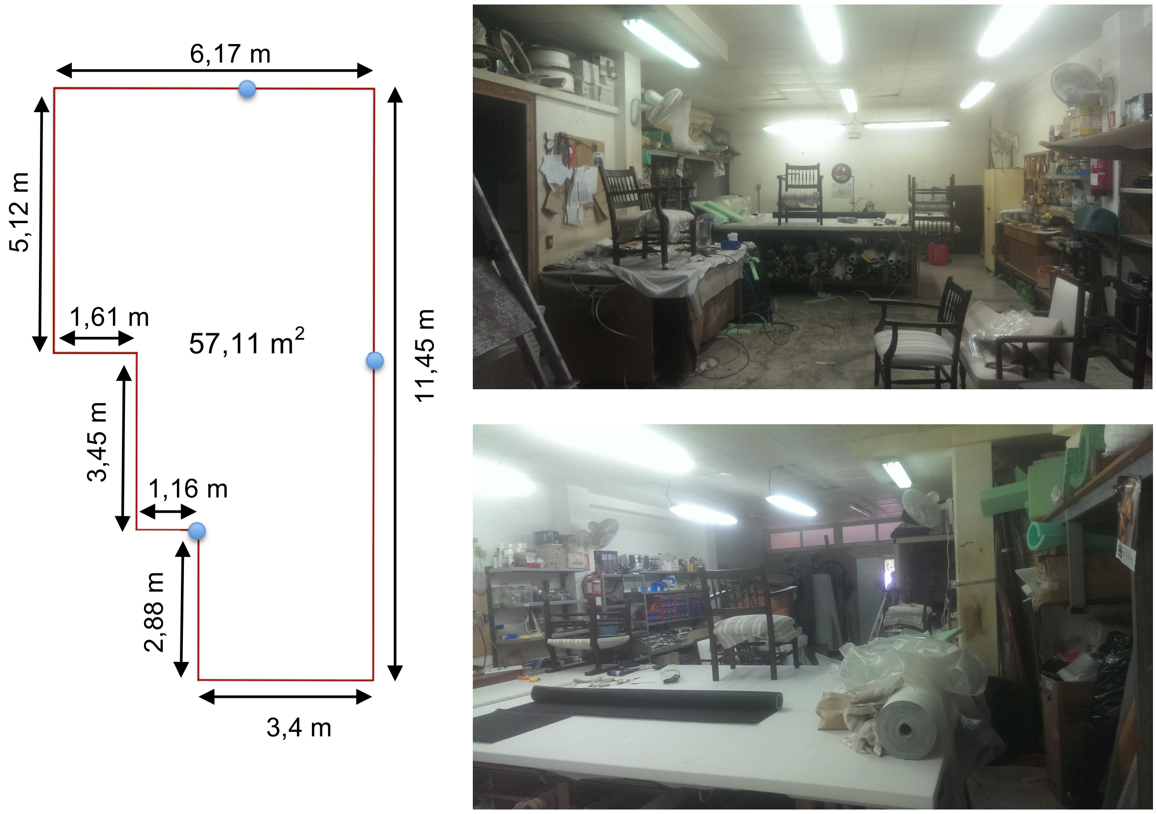


Fig 4.3. Upholstery plan and photos

4.2. Path Loss Model

The main objective of the analysis of the path loss models was to discard one empirical model studied during the project. Remember that the two studies to be analysed were log distance model and ITU indoor propagation model:

Log distance model	ITU Indoor propagation model
$RSSI = RSSI_0 - 10 \cdot n \cdot log_{10}\left(\frac{d}{d_0}\right) - Xg$	$L(dB) = 20\log f(MHz) + N\log d + P_f(n) - 28$

Table 4.1. Path loss models summary

The study has been carried out with ‘BLEScan RSSI’ android application developed during the project. The procedure to get the results was the following one:

- Place a beacon at desired position inside the room.
- Measure RSSI values each 0.5 or 1 meter.
- Get the CSV file from BLEScan Rssi.
- Analyse data with MATLAB: RSSI distribution, path loss model estimations, etc.

4.2.1. Living Room

Estimote beacon was placed on a wall before to starting capturing RSSI values at distances within range of 0.5 meters to 8 meters. The broadcast power of the beacon was 0 dBm and the advertising interval was 200 ms.

- **RSSI**

The RSSI captured at some distance shows that RSSI has a lognormal distribution in indoors:

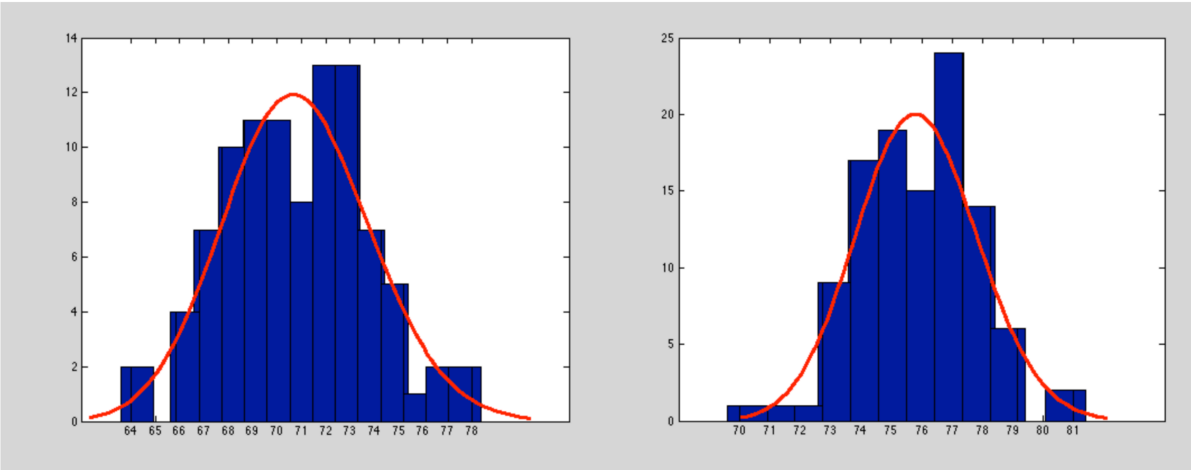


Fig 4.4. RSSI distribution in living room

A comparison between the RSSI data and lognormal distribution is compared in next plot:

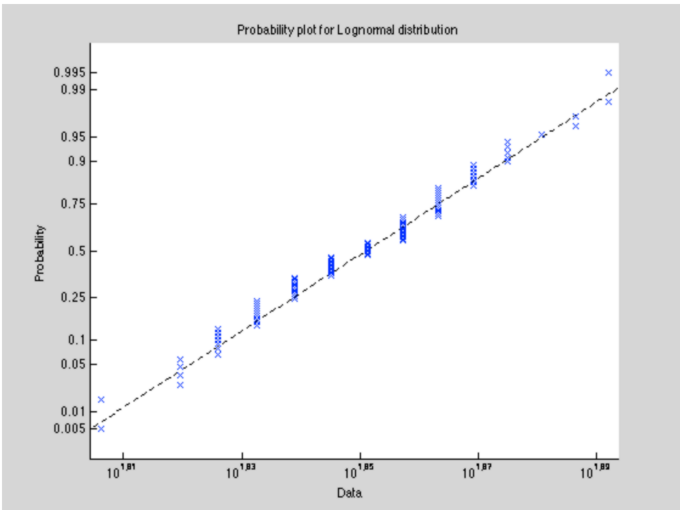


Fig 4.5. RSSI distribution compared to log normal distribution 1

So, it can be said that RSSI values in BLE follows a lognormal distribution.

In following figure, it can be seen that RSSI decreases over distance so it is a good parameter to use for distance estimation.

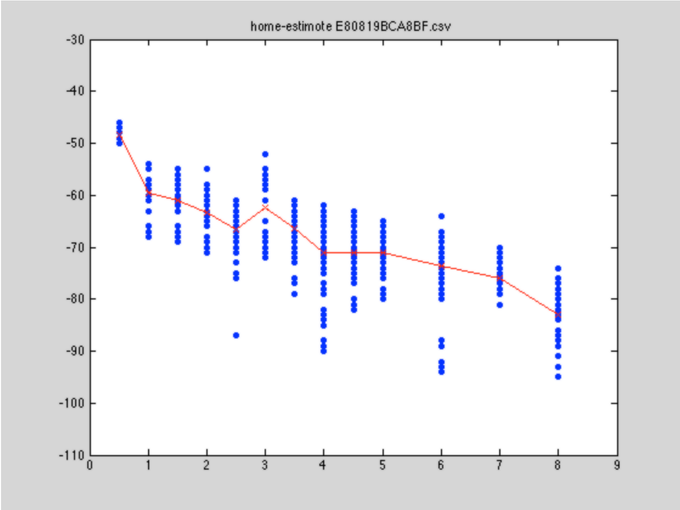


Fig 4.6. RSSI over distance plot 1

- **Log distance model**

The log distance model was applied to the data in order to find the parameters of the model that introduces the minimum mean square error:

$$MSE = \frac{1}{n}\sum_{n}(d_{estimate} - d_{real})^2 \quad (4.1)$$

Log distance model has two parameters that can be tuned: path loss exponent (n) and standard deviation (Xg). Standard deviation is implicit in the data, being the standard deviation of RSSI measured values. So, given a standard deviation, path loss exponent n which introduces minimum square error will be selected to set path loss model that follows the indoor propagation model of the living room.

Measured Power at 1m (dBm)	-52.36
Xg (dB)	4.95
n	2.4
MSE	1.1

Table 4.2. Parameters of logdistance path loss model in living room

With the logdistance path loss model parameterized as above, and the data captured, distance would be estimated as follows:

real (m)	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	6	7	8
estimation (m)	0.57	1.51	1.7	2.06	2.75	1.93	2.69	4.03	4	4	5.02	6.03	11.01
error (m)	0.07	0.51	0.2	0.06	0.25	1.07	0.81	0.03	0.5	1	0.98	0.97	3.01
								Average error (m)					0.73

Table 4.3. Distance estimation in living room with logdistance model #1

There are some values that are good estimated and others not, this is because of RSSI fluctuations and multipath components. But in general, the estimation is acceptable when the data fits the model. In addition, the MSE is small.

But for a correct modelling, this model should be applied in similar experiments with new data captured to see if the estimation really matches indoor environment changes.

So, applying this model to a collection of RSSI data measured (with measured Power at 1m of -59.58 dBm) under the same conditions, the estimation was:

real (m)	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	6	7	8
estimation (m)	0.69	0.81	1.3	1.43	2.5	1.74	2.67	4.76	3.35	3.94	3.83	5.93	11.15
error (m)	0.19	0.19	0.2	0.57	0	1.26	0.83	0.76	1.15	1.06	2.17	1.07	3.15
										Average error (m)			0.97

Table 4.4. Distance estimation in living room with logdistance model #2

The MSE has increased (MSE = 1.65) because the parameters chose were obtained minimizing square error of another experiment's data. This shows us how indoor environment changes over time and the difficulty to model indoor propagation. Estimation accuracy should be improved in positioning system.

- **ITU Indoor Propagation model**

The ITU Indoor propagation model was also applied to the data in order to find the parameters of the model that introduces the minimum square error (4.1).

This model has two parameters to be configured: distance power loss coefficient N and the floor loss penetration factor Pf(n). These parameters have no dependency with the data but with the environment, and their value should be found empirically or choosing a value recommended by the ITU depending on environment (whether is office or commercial).

So, using the same data as logdistance model and following the recommendations of ITU in a residential environment, the parameter's values should be:

$P_f(n)$	14
N	30
MSE	3.76

Table 4.5. Recommended Parameters of ITU Indoor Propagation Model

The minimum square error (MSE) is very high because of:

- Model has no dependency with the data.
- Recommendations from ITU are for specific environments.

In order to improve the results, N and $P_f(n)$ was chosen empirically with the ones which minimize the square error.

$P_f(n)$	15
N	27
MSE	2.38

Table 4.6. Parameters empirically found of ITU Indoor Propagation Model

The MSE has been reduced notoriously and improved results in distance estimation can be seen below:

real (m)	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	6	7	8
estimation (m)	0.51	0.62	1.06	1.18	2.21	1.47	2.37	4.54	3.06	3.67	3.56	5.82	11.84
error (m)	0.01	0.38	0.44	0.82	0.29	1.53	1.13	0.54	1.44	1.33	2.44	1.18	3.84
								Average error (m)					1.18

Table 4.7. Distance estimation in living room with ITU model #1

But applying this last model to new data increases MSE error (MSE = 5.3) due to that indoor environment changes and the model fits the data analysed previously.

Moreover, distance estimation gets a bit worse:

real (m)	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	6	7	8
estimation (m)	0.21	0.62	0.71	0.88	1.22	0.82	1.19	1.88	1.86	1.86	2.41	2.97	5.84
error (m)	0.29	0.38	0.79	1.12	1.28	2.18	2.31	2.12	2.64	3.14	3.59	4.03	2.16
								Average error (m)					2

Table 4.8. Distance estimation in living room with ITU model #2

- ***Estimote* Path Loss Model and Logdistance model comparison**

As has been commented in section 3.2.2.3, *estimote*'s SDK has a method to compute distances based on RSSI following a indoor path loss model that is not public. This method has been compared to logdistance path loss model in living room. A broadcast power of -8 dBm and advertising interval of 200 ms were used.

The parameters of logdistance model were:

Measured Power at 1m (dBm)	-59.58
Xg (dB)	2.3
n	1.9
MSE	0.05

Table 4.9. Logdistance parameters for living room (comparison with estimote SDK)

Several distance estimations were done with this path loss model reaching a MSE of 0.05 with next average estimations and errors:

real distance (m)	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5
estimated distance (m)	0.58	1.21	1.55	2.19	3.08	2.95	2.72	4.22	4.48	4.58	4.06
error (m)	0.08	0.21	0.05	0.19	0.58	0.05	0.78	0.22	0.02	0.42	1.44
								Average error (m)			0.37

Table 4.10. Distance estimation in living room with logdistance model (comparison with estimote SDK)

On the other hand, using *estimote*'s SDK method that computes distances, following results were obtained:

real distance (m)	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5
estimated distance (m)	0.28	0.73	1	1.59	2.43	1.73	1.39	2.14	1.73	1.77	1.59
error (m)	0.22	0.27	0.5	0.41	0.07	1.27	2.11	1.86	2.77	3.23	2.91
									Average error (m)		1.42

Table 4.11. estimote's SDK method distance estimation

It can be seen that the error is higher in *estimote*'s SDK case, this is because it is difficult to model an indoor propagation model for all cases, indoor path loss models need to be calibrated in each scenario because it varies over time and space.

So, it can be said that logdistance model presents better results than *estimote*'s SDK method computing distances and therefore can be discarded for the indoor positioning system.

4.2.2. Office

Estimote beacons were placed on the walls at different positions to see how indoor environment changes depending on beacon placement and where the device is. The broadcast power in this case was 0 dBm and the advertising interval was 200 ms.

Three experiments have been done in order to see the changes that furniture distribution introduces in RSSI values. So, first experiment was done in the right part of the office, second experiment was done in the left part of the office and the last one was done perpendicular to the last two experiments having components in beacon signal direction that introduces high multipath components such as tables, PCs and a column.

So, the RSSI follows a lognormal distribution as the case studied in the living room, but the degradation of the power over distance changes depending where device is. In next page, it can be seen RSSI distribution in three different areas inside the office.

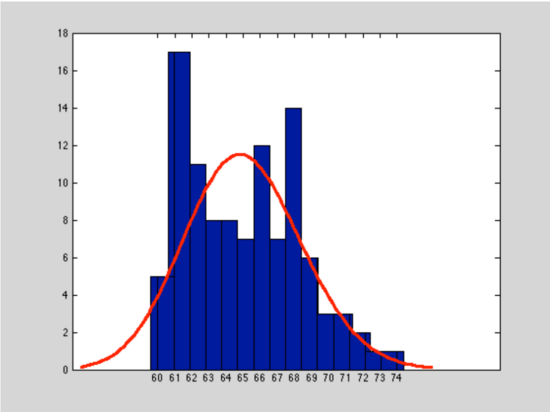
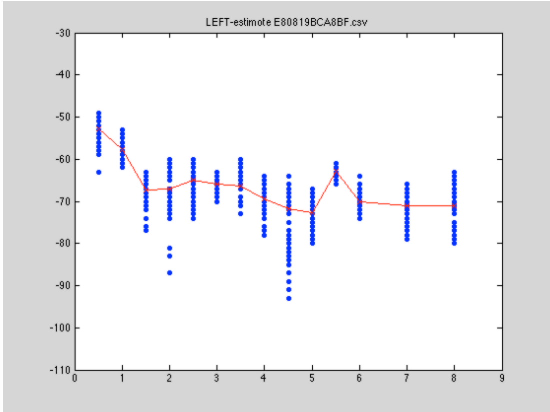
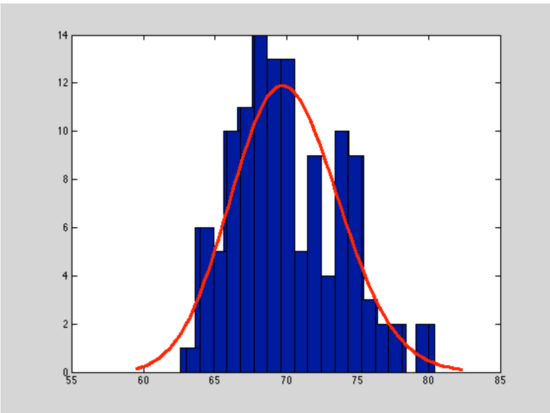
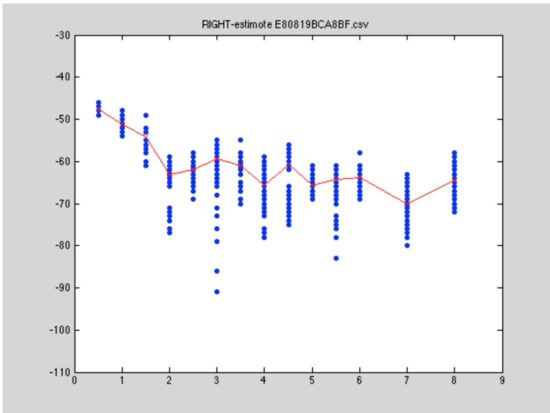
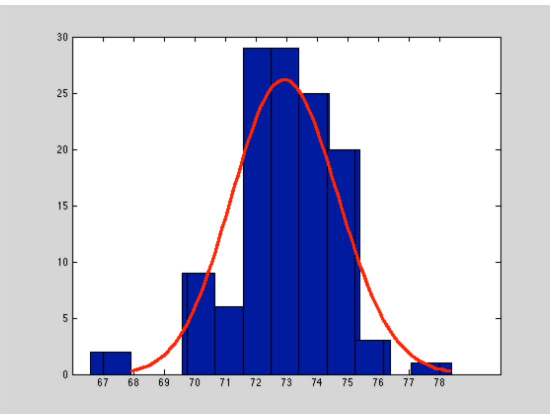
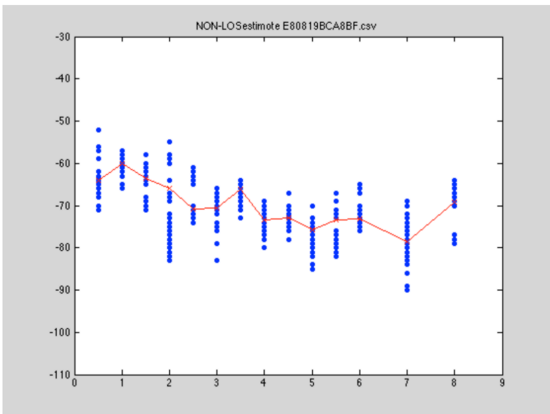
LOS (left)	
RSSI distribution	**RSSI over distance**
	
LOS (right)	
RSSI distribution	**RSSI over distance**
	
NLOS	
RSSI distribution	**RSSI over distance**
	

Table 4.12. LOS left and right, and NLOS RSSI office analysis

Analysing the results, RSSI over distance plot changes depending where beacons are placed or device is. So, a path loss modelling will have to be done for each beacon and zone within indoor environment to be modelled.

- **Log distance model**

Following the same instructions as living room case: Xg is the deviation of RSSI measured values and n is the one that minimizes square error. Following results were obtained from the measures at each part of the room.

	LOS (left)	LOS (right)	NLOS
Measured Power at 1m (dBm)	-57.71 dBm	-51.23 dBm	-60.15 dBm
Xg (dB)	4.63 dB	4.19 dB	3.89 dB
n	1.1	1.4	1.4
MSE	0.17	0.19	0.41

Table 4.13. Parameters of logdistance path loss model in office

In summary, it can be seen that indoor environment changes depending on beacon placement and which zone target device is. Measured power changes notoriously whenever device is, and also the standard deviation and path loss exponent. This introduces losses in estimation and increases mean square error.

- **ITU Indoor Propagation model**

The same study as living room case was done in order to see how environment changes depending on beacon placement.

Firstly, the ITU indoor propagation model parameters were the ones recommended by ITU in office case for a frequency of 2.45 GHz:

$P_f(n)$	N
14	30

Table 4.14. ITU recommendations in office environment

So, the results are presented below:

	LOS (left)	LOS (right)	NLOS
MSE	10.61	15.54	4.20

Table 4.15. MSE of distance estimation with ITU recommendations in office

It is unacceptable results for estimation, so, new parameters has been found empirically minimizing square error:

	LOS (left)	LOS (right)	NLOS
$P_f(n)$	11	14	16
N	27	16	25
MSE	0.06	0.08	0.07

Table 4.16. ITU parameters found empirically in office

As can be seen, parameters differ a lot depending on the zone analysed, as logdistance model case.

4.2.3. Summary Results

Analysing the results, two studies can be differed: living room case studies how indoor environment varies with time and office case studies what happens when device move for different zones within indoor office.

The following conclusions can be extracted:

- Indoor environment is difficult to model because it varies with time and location.
- RSSI follows a lognormal distribution.
- ITU recommendations for ITU indoor model gives bad results and need to be calculated empirically for each case.
- Path loss model is based on actual data such as the measured power at 1m and standard deviation.

So, even though that both ITU indoor propagation model and logdistance model presents good results, path loss model will be the method applied in the positioning system because:

- It has data dependency.
- It is the most used in other indoor positioning systems.
- It has high accuracy performance.
- ITU indoor propagation model is different for each location and time so it is difficult find a good configuration that fits better for all cases.

4.3. Positioning Results

The indoor positioning system designed in section 3.3 was applied in two scenarios: living room and workshop. The path loss model applied was logdistance model as discussed in section 4.2. Calibration methods were analysed in the first scenario and the best calibration method of these results was applied in the positioning system of the workshop.

There were placed 3 beacons, each one with its MAC Address and color as identifications.

Beacon (color)	MAC Address
Blueberry Pie	E8:08:19:BC:A8:BF
Mint Cocktail	D2:8C:CB:0D:F3:89
Icy Marshmallow	C5:43:C8:CF:EA:5D

Table 4.17. Beacons description (MAC Address)

4.3.1. Living Room

The beacons were placed randomly on the walls at separated distances allowing full coverage inside the room. There were used 3 beacons, so this means that trilateration was done in 2D. The measures were done in the reference plane formed by 3 beacons placed on the walls. For this positioning system, the reference coordinates are (2,2) in bottom left and (6, 7.76) in top right. In addition, broadcast power was -8 dBm and the advertising interval was 200 ms.

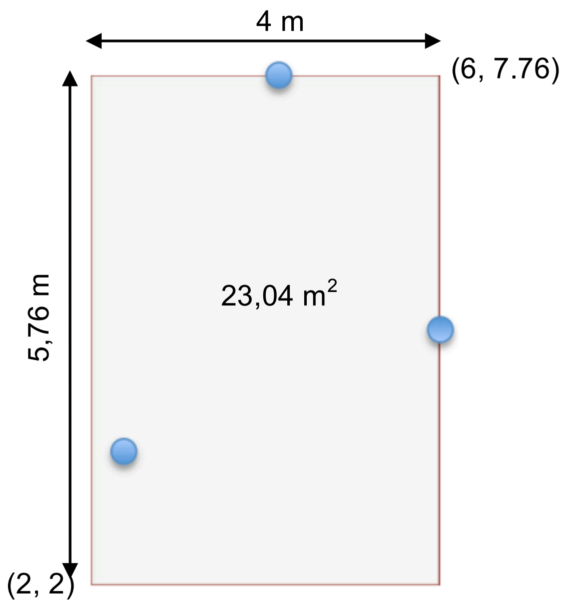


Fig 4.7. Living room plan and beacons placement

Beacons were placed at positions:

E8:08:19:BC:A8:BF	D2:8C:CB:0D:F3:89	C5:43:C8:CF:EA:5D
(2.33, 3.505)	(4.1, 7.56)	(6, 4.83)

Table 4.18. Beacons position in living room

- **Mathematical method**

Before starting analysing positioning calibrations, a comparison between Linear Least Squares and Non Linear Least Squares was done in order to choose the method that presents best results. So, some points were estimated with these two methods.

(NLSQ had been configured with 50 iterations)

	(4.14, 5.59)	error (m)	(4.93, 6.21)	error (m)	(3.515, 6.83)	error (m)	(3.39, 3.41)	error (m)
LSQ	(4.21, 5.31)	0.29	(4.76, 5.83)	0.42	(3.73, 5.38)	1.47	(3.76, 5)	1.63
NLSQ	(4.21, 5.31)	0.29	(5.82, 6.54)	0.95	(3.86, 5.36)	1.51	(3.85, 4.97)	1.63

Table 4.19. Comparison between LSQ and NLSQ techniques

Even though that results are very similar, LSQ presented better results than NLSQ. In addition, LSQ is simpler and has less computational cost. So, following results will be obtained computing coordinates with LSQ method.

- **Point calibration**

Point calibration was done at point (4.14, 5.59), more or lees in the middle of the room. The parameters of logdistance model for each beacon were calibrated two times in order to see how indoor environment changes (as seen in section 4.2) and its effect on results.

Beacon	n		Xg	
	1	2	1	2
E8:08:19:BC:A8:BF	1.6	1.6	2.91	1.24
D2:8C:CB:0D:F3:89	3.5	3	3.56	5.87
C5:43:C8:CF:EA:5D	3.7	4.2	2.8	2.94

Table 4.20. Logdistance values for calibration 1 and 2 in living room

It can be seen some fluctuations in standard deviation of the data and in path loss exponent because of indoor environment changes, but it must be analysed with the results. So, some position estimation was done in several points in order to see the error estimation between both calibrations.

There were three estimations for each point to be located in order to see the variations of the RSSI in the results. Below are presented the points to be located, the best estimated position and the average error of three positions. Finally, the average error of all the estimations is done for each calibration.

	Calibration 1			Calibration 2		
Point	**Best estimation – error (m)**	**Average estimation – error (m)**	**Average error (m)**	**Best estimation – error (m)**	**Average estimation – error (m)**	**Average error (m)**
(4.14, 5.59)	(4.16, 5.45) – 0.14	(4.28, 5.47) – 0.18	0.3	(4.2, 5.78) – 0.2	(4.46, 5.82) – 0.39	0.42
(4.93, 6.21)	(4.72, 5.92) – 0.36	(5.57, 6.83) – 0.89	1.16	(5.66, 6.58) – 0.82	(5.9, 7) – 1.25	1.3
(3.51, 6.83)	(3.69, 5.5) – 1.34	(4.06, 5,57) – 1.37	1.42	(4.39, 6.05) – 1.17	(4.49, 5.64) – 1.54	1.6
(3.39, 3.41)	(3.72, 5.15) – 1.77	(3.79, 5.45) – 2.08	2.15	(3.82, 3.87) – 0.63	(3.86, 4.46) – 1.15	1.18
(5.24, 3.1)	(5.72, 4.28) – 1.27	(5.65, 5.39) – 2.33	2.34	(4.16, 5.3) – 2.45	(4.7, 5.52) – 2.48	2.58
Total average error (m)	**0.98**	**1.37**	**1.47**	**1.05**	**1.36**	**1.42**

Table 4.21. Calibration 1 and 2 positioning results in living room

The results are quite good when target device is near to the calibration point. As device gets far from this point, the estimation gets worse increasing the error more than 2 meters. Moreover, even though two calibrations give different path loss configurations, the average error is similar in these two experiments.

Point calibration is a good solution that provides the optimal balance between estimation and fast calibration.

- **Environment calibration**

Environment calibration calibrates the path loss of all the room for each beacon. The calibration is done measuring RSSI every 0.5 meters. Then, path loss is calibrated due to RSSI and distance measured.

In the experiment, the beacons were calibrated as follows:

- **C5:43:C8:CF:EA:5D**

This beacon presented a measured power at 1m of -64 dBm and RSSI over distance:

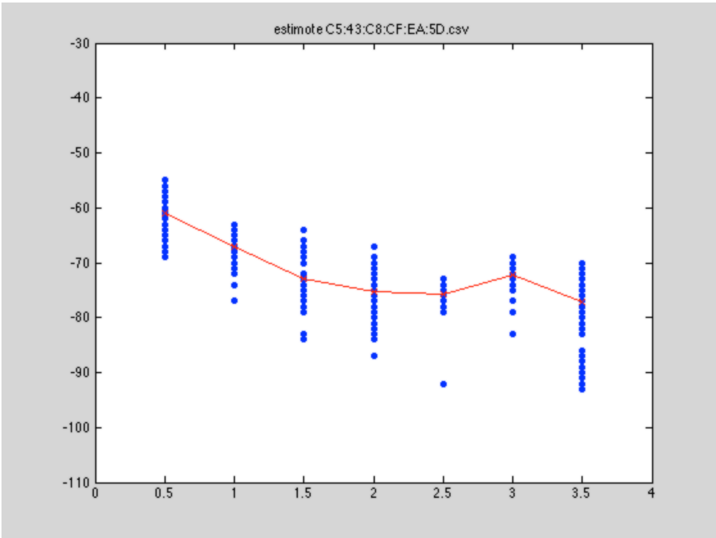


Fig 4.8. RSSI over distance for beacon C5:43:C8:CF:EA:5D

Path loss model could be calibrated given the standard deviation of the data which was Xg = 3.57. So, comparing the different path loss exponent (n) values, the value that has minimum MSE is n = 1.8 that presents a MSE = 0.03.

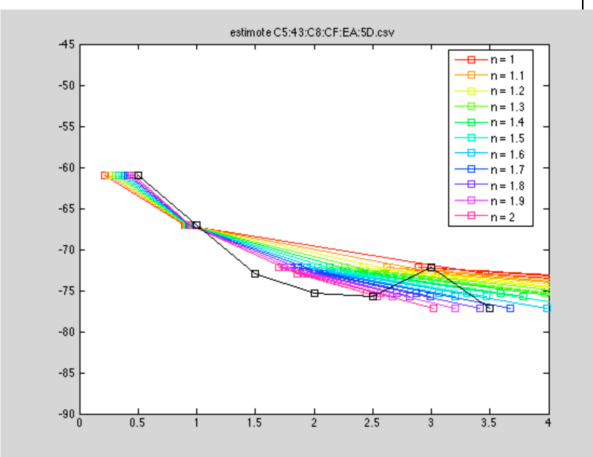
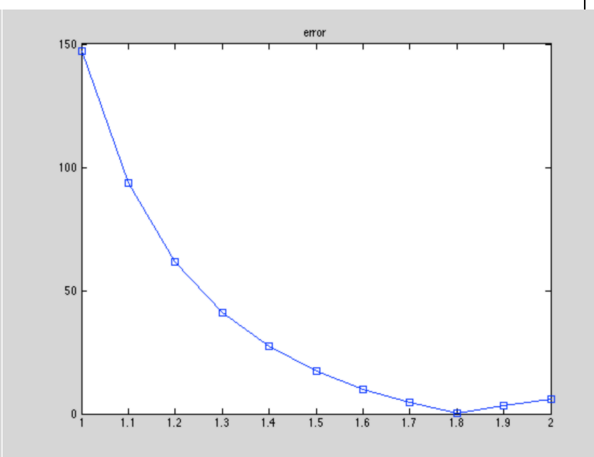
Log distance model	MSE error
	

Table 4.22. Logdistance model and MSE error for beacon C5:43:C8:CF:EA:5D

-   **D2:8C:CB:0D:F3:89**

This beacon presented a measured power at 1m of -61 dBm and RSSI over distance:

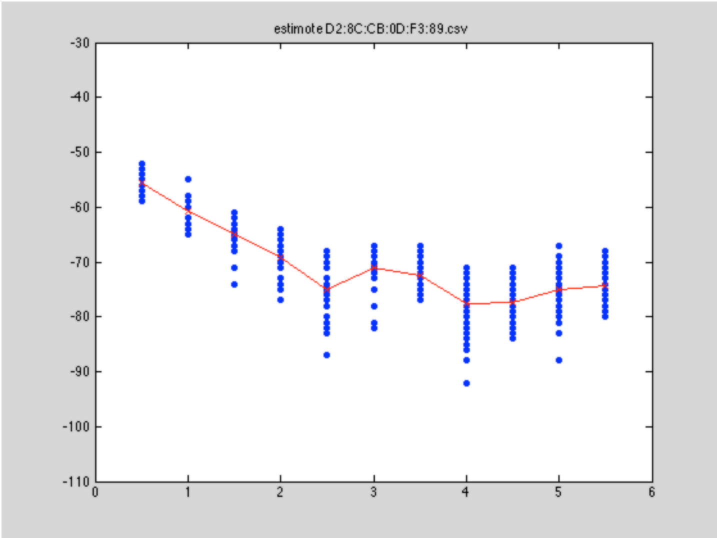


Fig 4.8. RSSI over distance for beacon D2:8C:CB:0D:F3:89

Path loss model could be calibrated given the standard deviation of the data which was Xg = 3.31. So, comparing the different path loss exponent (n) values, the value that has minimum MSE is n = 1.8 that presents a MSE = 0.47.

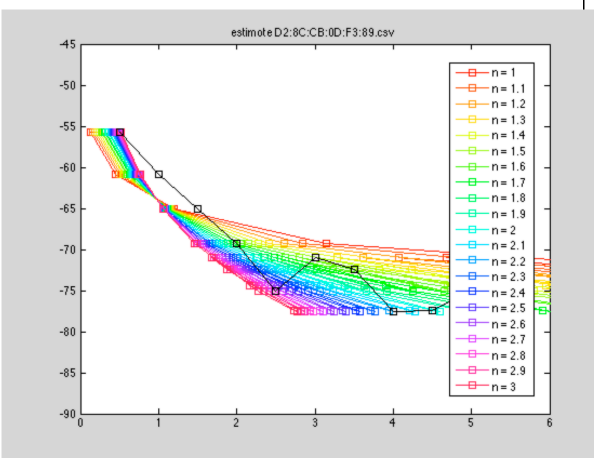
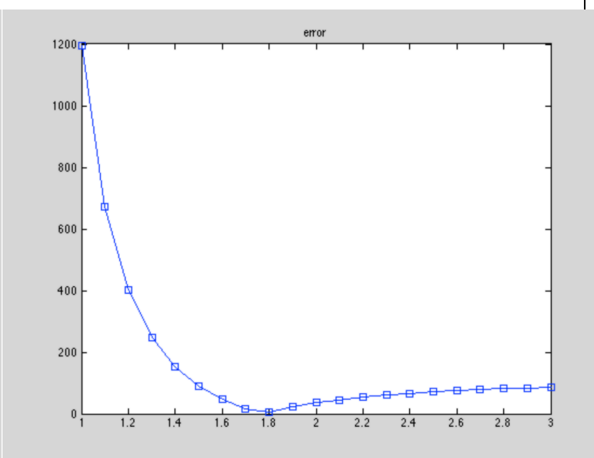
Log distance model	**MSE error**
	

Table 4.23. Logdistance model and MSE error for beacon D2:8C:CB:0D:F3:89

- **E8:08:19:BC:A8:BF**

This beacon presented a measured power at 1m of -63 dBm and RSSI over distance:

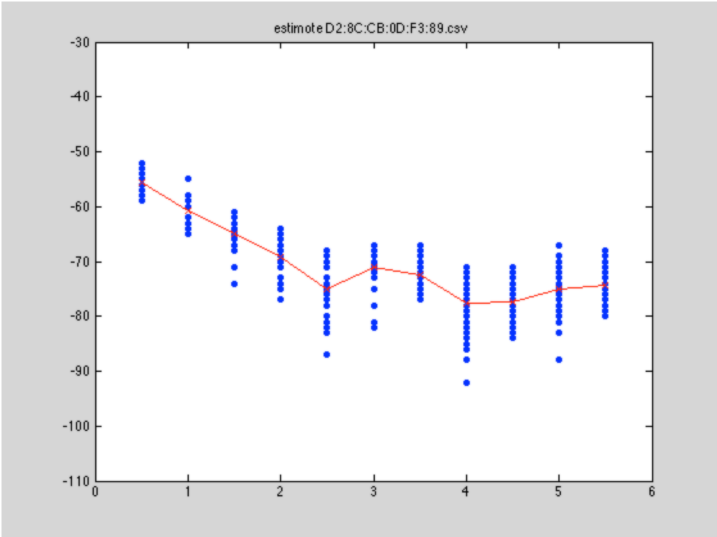


Fig 4.9. RSSI over distance for beacon E8:08:19:BC:A8:BF

Path loss model could be calibrated given the standard deviation of the data which was Xg = 3.73. So, comparing the different path loss exponent (n) values, the value that has minimum MSE is n = 2.2 that presents a MSE = 0.16.

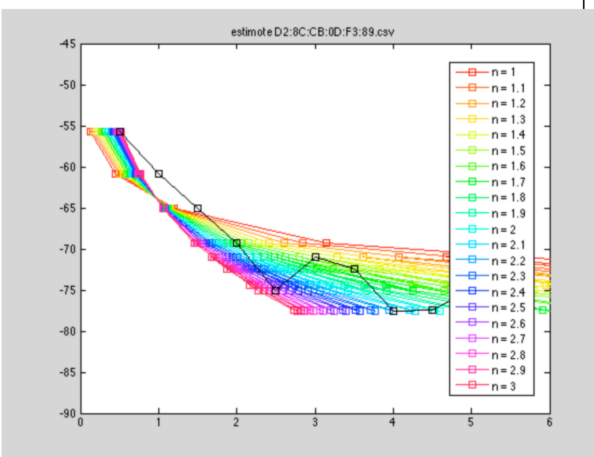
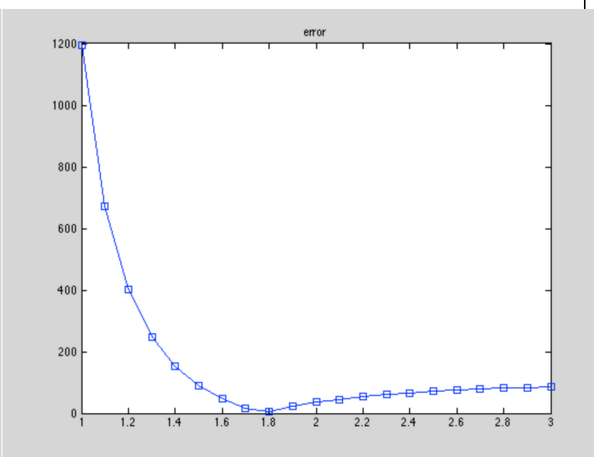
Log distance model	**MSE error**
	

Table 4.24. Logdistance model and MSE error for beacon E8:08:19:BC:A8:BF

In summary, indoor environment was finally calibrated with next values:

Beacon	n	Xg
C5:43:C8:CF:EA:5D	1.8	3.57
D2:8C:CB:0D:F3:89	1.8	3.31
E8:08:19:BC:A8:BF	2.2	3.73

Table 4.25. Path loss configuration for environment calibration

Then, the estimated position at the same points as point calibration case was done with following results. There are presented the best estimated point, the average of three estimations and the average error of the error of each estimation.

Point	Best estimation – error (m)	Average estimation – error (m)	Average error (m)
(4.14, 5.59)	(4.18, 4.76) – 0.83	(4.18, 4.12) – 1.47	1.59
(4.93, 6.21)	(5.28, 2.14) – 4.08	(5.97, 2.05) – 4.28	5.28
(3.51, 6.83)	(2, 4.52) – 2.76	(3.13, 3.51) – 3.34	3.57
(3.39, 3.41)	(4.02, 3.69) – 0.69	(2.81, 4.32) – 1.08	1.44
(5.24, 3.1)	(5.61, 2) – 1.16	(6.18, 2) – 1.45	1.48
Total average error (m)	**1.9**	**2.32**	**2.67**

Table 4.26. Positioning results for environment calibration

It can be observed that estimation gets worse with this calibration method than the point calibration. It is because path loss is configured over a straight line of distance with respect to the beacons. So, there are positions in the indoor environment that the path loss may differ due to indoor environment changes.

In conclusion, is difficult to obtain a good configuration of path loss that fits well on all the surface of a room. It would be always some points that would not fit the model. It can be seen in the results which deteriorate with respect to the point calibration case. The accuracy has been reduced to 2.32-2.67 meters when in point calibration we had an accuracy of 1.36-1.47 meters.

- **Point calibration based on proximity**

This method exploits *iBeacon* protocol features basing calibration on proximity. There were chosen three points (one per beacon) to calibrate path loss model for point and beacon. So, there were be different configurations of path loss model for each beacon depending where device would be, whether device is near one beacon or another, path loss parameters will take one values or others.

The calibration points and path loss parameters for each beacon were done two times and are presented below:

Calibrated at (4.14, 6.51)	**1**		**2**	
Beacon (nearest in bold)	**n**	**Xg**	**n**	**Xg**
C5:43:C8:CF:EA:5D	3.33	1.12	1.91	1.82
E8:08:19:BC:A8:BF	2.12	1.23	2.18	3.16
D2:8C:CB:0D:F3:89	3.44	3.41	6.37	1.25

Table 4.27. Point calibration with D2:8C:CB:0D:F3:89 as nearest beacon in living room

Calibrated at (4.31, 4.97)	**1**		**2**	
Beacon (nearest in bold)	**n**	**Xg**	**n**	**Xg**
C5:43:C8:CF:EA:5D	1.09	1.66	2.98	1.8
E8:08:19:BC:A8:BF	3.11	1.2	3.19	2.52
D2:8C:CB:0D:F3:89	1.97	2.08	3.57	3.2

Table 4.28. Point calibration with C5:43:C8:CF:EA:5D as nearest beacon in living room

Calibrated at (3.89, 3.41)	**1**		**2**	
Beacon (nearest in bold)	**n**	**Xg**	**n**	**Xg**
C5:43:C8:CF:EA:5D	2.24	3.31	3.34	4.41
E8:08:19:BC:A8:BF	2.8	2.06	1.86	3.48
D2:8C:CB:0D:F3:89	1.77	3.86	2.16	3.94

Table 4.29. Point calibration with E8:08:19:BC:A8:BF as nearest beacon in living room

It can be seen how indoor environment varies over time and how difficult is to model indoor propagation with RSSI values. As in other experiments, there were three measures for each point, the best estimation, the average estimation and the average of the error of three estimations are presented in next table:

	Calibration 1			Calibration 2		
Point	**Best estimation – error (m)**	**Average estimation – error (m)**	**Average error (m)**	**Best estimation – error (m)**	**Average estimation – error (m)**	**Average error (m)**
(4.14, 5.59)	(3.85, 5.76) – 0.34	(3.95, 5.57) – 0.19	0.58	(3.94, 5.2) – 0.44	(4.13, 5.46) – 0.13	0.61
(4.93, 6.21)	(4.54, 6.02) – 0.43	(5.51, 6.77) – 0.81	1.14	(4.26, 5.42) – 1.04	(4.12, 6.06) – 0.82	1.17
(3.51, 6.83)	(4.54, 6.39) – 1.11	(4.88, 6.58) – 1.39	1.41	(3.2, 6.6) – 0.39	(2.55, 7.17) – 1.02	1.1
(3.39, 3.41)	(3.25, 4.06) – 0.66	(3.64, 3.06) – 0.43	0.86	(3.97, 2.88) – 0.79	(3.73, 4.13) – 0.8	1.17
(5.24, 3.1)	(5.02, 2) – 1.12	(4.89, 2.71) – 0.52	1.45	(6, 3.75) - 1	(4.75, 2.6) – 0.7	1.37
Total average error (m)	**0.73**	**0.67**	**1.1**	**0.73**	**0.69**	**1.08**

Table 4.30. Positioning results for calibration 1 and 2 of point calibration based on proximity in living room

It can be seen how accuracy is improved from other calibration methods reaching values about 0.67-1.08 meters, which is a good result. So, time consumption doing the calibration is compensated by these excellent results. Moreover, the different values in path loss model for two calibrations mean that path loss model has not unique solution because the results are similar in two calibrations.

4.3.1.1. Beacon placement comparison

Another important factor in indoor positioning systems is how and where beacons are placed in the room. In this experiment, position of beacons were changed to another coordinates in order to see how this affect on results:

E8:08:19:BC:A8:BF	D2:8C:CB:0D:F3:89	C5:43:C8:CF:EA:5D
(2.77, 5.6)	(6, 7.11)	(5.81, 3.05)

Table 4.31. Beacons position in living room #2

The calibration used in this experiment was the point calibration based on proximity. It has done two different calibrations with its results. So, calibration set the path loss parameters:

Calibrated at (4.62, 6.83)	**1**		**2**	
Beacon (nearest in bold)	**n**	**Xg**	**n**	**Xg**
C5:43:C8:CF:EA:5D	1.9	2.09	1.82	3.26
E8:08:19:BC:A8:BF	0.75	3.28	0.7	4.95
D2:8C:CB:0D:F3:89	0.68	0.87	4.21	1.93

Table 4.32. Point calibration with D2:8C:CB:0D:F3:89 as nearest beacon in living room

Calibrated at (4.62, 3.72)	**1**		**2**	
Beacon (nearest in bold)	**n**	**Xg**	**n**	**Xg**
C5:43:C8:CF:EA:5D	0.96	3.26	0.95	2.2
E8:08:19:BC:A8:BF	2.58	5.6	3.39	2.39
D2:8C:CB:0D:F3:89	2.18	1.22	2.3	1.47

Table 4.33. Point calibration with C5:43:C8:CF:EA:5D as nearest beacon in living room

Calibrated at (3.82, 4.97)	**1**		**2**	
Beacon (nearest in bold)	**n**	**Xg**	**n**	**Xg**
C5:43:C8:CF:EA:5D	2.17	2.72	2.81	2.24
E8:08:19:BC:A8:BF	4.86	1.73	0.45	0.99
D2:8C:CB:0D:F3:89	3.05	4.06	2.2	4.07

Table 4.34. Point calibration with E8:08:19:BC:A8:BF as nearest beacon in living room

Obviously, path loss parameters were different than the others from last beacon placement configuration.

In addition, next estimated positions and errors were obtained for the same points evaluated in last beacon placement configuration:

	Calibration 1			Calibration 2		
Point	**Best estimation – error (m)**	**Average estimation – error (m)**	**Average error (m)**	**Best estimation – error (m)**	**Average estimation – error (m)**	**Average error (m)**
(4.14, 5.59)	(3.31, 5.95) – 0.83	(3.98, 5.07) – 0.54	0.94	(3.05, 3.3) – 2.54	(4.14, 3.07) – 2.52	2.79
(4.93, 6.21)	(4.33, 5.2) – 1.18	(4.25, 5.17) – 1.24	1.26	(2, 7.76) – 3.32	(3.33, 5.84) – 1.64	3.66
(3.51, 6.83)	(3.75, 7.38) – 0.6	(3.97, 5.82) – 1.11	1.46	(3.17, 4.79) – 2.07	(4.22, 3.43) – 3.47	3.57
(3.39, 3.41)	(3.43, 3.07) – 0.34	(2.95, 2.58) - 0.83	0.96	(2, 2.63) – 1.59	(2, 2.21) – 1.84	1.85
(5.24, 3.1)	(2.45, 2) - 3	(2.15, 2) – 3.28	3.28	(4.22, 3.45) – 1.08	(2.99, 2.85) – 2.26	2.33
Total average error (m)	**1.19**	**1.4**	**1.58**	**2.12**	**2.35**	**2.84**

Table 4.35. Positioning results for calibration 1 and 2 of point calibration based on proximity in living room

In this experiment, calibration 2 gave worse results than calibration 1 with big difference, it could be times that calibration is not well done or fails in its estimation of path loss model. So, possible errors in calibration must be considered in future work. Therefore, a validation after calibration would be needed in any positioning system.

Following table presents a comparison between these results and last results for the other beacon placement configuration:

Beacon placement	**Calibration**	**Minimum error**	**Average error of average estimation**	**Total average error**
1	1	0.73	0.67	1.1
	2	0.73	0.69	1.08
2	1	1.19	1.4	1.58
	2	2.12	2.35	2.84

Table 4.36. Positioning error results in living room

These results show how a change in configuration of beacon placement can affect negatively position estimation. In this case, beacon placement configuration 1 gives better results in both calibrations than in beacon placement configuration 2. This means that beacon placement configuration needs to be studied and analysed to find the best manners to place the beacons that gives best results in estimation.

4.3.1.2. Summary results

The results of all calibration methods are summarized in next table:

Beacon placement configuration	**Calibration method**	**Minimum error**	**Average error of average estimation**	**Total average error**
1	Point Calibration 1	0.98	1.37	1.47
	Point Calibration 2	1.05	1.36	1.42
	Environment Cal.	1.9	2.32	2.67
	Point cal. based on proximity 1	**0.73**	**0.67**	1.1
	Point cal. based on proximity 2	**0.73**	0.69	**1.08**
2	Point cal. based on proximity 1	1.19	1.4	1.58
	Point cal. based on proximity 2	2.12	2.35	2.84

Table 4.37. Summary results of all calibration methods in living room

In conclusion, the point calibration based on proximity method gives the best results (in bold) of estimation with an accuracy of 0.67-1.08 meters. This will be the calibration method chosen for future experiments. However, beacon placement configuration must be studied perfectly that has been seen that it introduces some error in position estimation depending how is configured.

4.3.2. Workshop

The beacons were placed randomly on the walls at separated distance allowing full coverage inside the room. There were used 3 beacons, so this means that trilateration was done in 2D. The measures were done in the reference plane formed by 3 beacons placed on the walls. For this positioning system, the reference coordinates are (0,0) in bottom left and (6.17, 11.4) in top right. In addition, broadcast power was -4 dBm and the advertising interval 200 ms. Estimated positions may be negatively affected due to the increase of the total area to be covered (57.11 m^2) with respect to the living room (23 m^2).

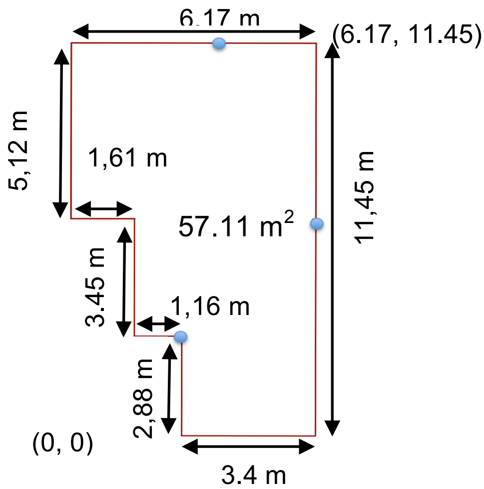


Fig 4.10. Workshop plan and beacons placement

Beacons were placed at positions:

E8:08:19:BC:A8:BF	D2:8C:CB:0D:F3:89	C5:43:C8:CF:EA:5D
(6.17, 6.22)	(3.68, 11.4)	(2.77, 2.88)

Table 4.38. Positions of beacons in workshop

- **Point calibration based on proximity**

Point calibration based on proximity presented the best results in the experiment in living room. So, this calibration method was the only applied in this method.

The calibration points and path loss parameters for each beacon were done two times and are presented below:

Calibrated at (3.91, 9.61)	**1**		**2**	
Beacon (nearest in bold)	**n**	**Xg**	**n**	**Xg**
C5:43:C8:CF:EA:5D	2.22	3.59	1.45	1.95
E8:08:19:BC:A8:BF	3.18	3.96	2.89	2.67
D2:8C:CB:0D:F3:89	0.7	1.60	2.79	2.56

Table 4.39. Point calibration with D2:8C:CB:0D:F3:89 as nearest beacon in workshop

Calibrated at (3.09, 4.69)	**1**		**2**	
Beacon (nearest in bold)	**n**	**Xg**	**n**	**Xg**
C5:43:C8:CF:EA:5D	0.608	1.52	2.51	1.48
E8:08:19:BC:A8:BF	3.25	2.52	2.65	5.62
D2:8C:CB:0D:F3:89	1.63	2.75	1.81	1.48

Table 4.40. Point calibration with C5:43:C8:CF:EA:5D as nearest beacon in workshop

Calibrated at (4.73, 5.92)	**1**		**2**	
Beacon (nearest in bold)	**n**	**Xg**	**n**	**Xg**
C5:43:C8:CF:EA:5D	1.73	4.31	1.83	3.57
E8:08:19:BC:A8:BF	3.96	0.86	3.97	1.47
D2:8C:CB:0D:F3:89	2.09	3.528	1.44	2.32

Table 4.41. Point calibration with E8:08:19:BC:A8:BF as nearest beacon in workshop

It can be seen how indoor environment varies over time and how difficult is to model indoor propagation with RSSI values.

As in other experiments, there were three measures for each point, the best estimation, the average estimation and the average of the error of three estimations are presented in next table:

	Calibration 1			Calibration 2		
Point	**Best estimation – error (m)**	**Average estimation – error (m)**	**Average error (m)**	**Best estimation – error (m)**	**Average estimation – error (m)**	**Average error (m)**
(4.32, 6.33)	(4.53, 6.93) – 0.64	(4.74, 7.33) – 1.08	1.49	(3.42, 5.9) – 1	(4.84, 5.58) – 0.91	1.57
(4.32, 3.87)	(3.78, 3.59) – 0.61	(2.28, 5.12) – 2.39	2.49	(3,74, 4.76) – 1.06	(2.96, 5.4) – 2.05	2.05
(3.9, 9.61)	(4.29, 9.12) – 0.63	(3.86, 8.99) – 0.62	0.71	(4.58, 10.25) – 0.93	(5.64, 10.94) – 2.19	2.19
(5.14, 7.56)	(2.93, 6.6) – 2.41	(4.08, 4.81) – 2.94	3.52	(3.58, 6.32) – 1.99	(3.21, 6.47) – 2.22	2.22
(2.9, 8.45)	(5.73, 9.05) – 2.71	(5.78, 7.25) – 3.12	3.22	(3.01, 6.54) – 1.91	(3.66, 6.6) - 2	2.03
Total average error (m)	**1.4**	**2.03**	**2.29**	**1.38**	**1.87**	**2.01**

Table 4.42. Positioning results for point calibration method based on proximity in workshop

It can be seen how accuracy gets worse as area increases. The accuracy has been reduced to 2.01-2.29 meters instead of 1.08 meters (in living room). This is because points are farther from calibration points and as device gets far from them, the accuracy is reduced. With a smallest area, more calibrated points or more beacons, good results can be obtained as it has seen in living room case.

5. Budget

The total budget of this project is presented in next table:

Description	Cost	Quantity	Total
CC2541DK-MINI Texas Instruments	80 €	1	80 €
estimote beacons Developer Preview Kit	80 €	1	80 €
Junior Engineer hour	15 €	300	4500 €
	Total:		4660 €

Table 5.1. Budget table

This project is not a prototype of a commercialized indoor positioning system, it is a first study of the opportunities of Bluetooth Low Energy in indoor positioning system. So, if there were some IPS commercial solution based on this project, it would need to analyse the possible extra costs of implementing and releasing an IPS in the market.

6. Environment Impact

The main feature of Bluetooth Low Energy is its low energy consumption in front of other technologies such as Wi-Fi or Bluetooth. Energy consumption is related to the contamination of a product. So, it does that BLE has a good environment impact.

Nowadays, low consumption and the way we producing energy is very important in our lives, population is more conscious in the manner energy is consumed, especially in smartphones, using a technology that increases battery life is a good manner of saving energy. Moreover, it means an opportunity for industry to make products with low cost and low environmental impact.

7. Conclusions and future development

7.1. Conclusion

The main objective of this project was to confirm that Bluetooth Low Energy could be used in an indoor positioning system, studying BLE features and applying them to obtain the maximum possible accuracy.

In this project, there were also studied the factors that are implied in an indoor positioning system such as path loss model and positioning techniques. Various models and techniques have been analysed in order to choose the one that presents better results in the IPS developed.

So, this conclusion presents an analysis of the results of the final version of the implemented indoor positioning system based on Bluetooth Low Energy:

- RSSI of BLE presents a lognormal distribution that allows using logdistance path loss model in order to estimate distance.
- Distance estimation with logdistance model presents good results with BLE: low MSE and accuracy of 1~1.5 meters in totally line of sight.
- Mathematical methods LSQ and NLSQ present good results on positioning, being LSQ the best technique with less error distance in location in this project.
- Point calibration is the best technique to calibrate the IPS if low time consumption in calibration is needed, reaching accuracy about 1.5 meters. On the other hand, point calibration based on proximity is the best solution that presents higher accuracy (1 meter), which is better than the papers studied previously.

So, even though Bluetooth Low Energy is not though for indoor positioning but for proximity, good results with high accuracy can be obtained implementing an IPS based on BLE. Moreover, BLE is widely available in smartphones, which becomes the optimal technology to use it in an IPS in front of other technologies such as Wi-Fi and Bluetooth. In this thesis, has been obtained better results with BLE than other indoor positioning systems based on Wi-Fi or Bluetooth. So, BLE transforms the way of commercial solutions for indoor positioning system adding a good technology for using for it.

A disadvantage of BLE beacons could be that they are not totally deployed, indoor positioning systems have recently started to be commercialized and its results can create confusion in some cases for retailers, even for discussing its utility. In addition, BLE beacons have recently been released. So, it is a new technology and need to be further studied. Moreover, beacons prices are too high for being commercially deployed because most companies are in a development phase. However, these prices will get down soon allowing a complete integration of beacons with society.

This project presented an Android application for indoor positioning. So, that means that IPS based on BLE can be perfectly used in smartphones and commercialized. Being compatible with smartphones allows exploiting indoor positioning systems in most situations reaching a widely range of consumers because nowadays, almost everyone has a smartphone.

Even though low consumption has not been analysed in this project, BLE is a technology with this aim. So, it is another advantage for using it in indoor positioning systems.

In comparison with other systems such as [4], [5] and others, this project introduces a new method of calibration based on proximity that presents better results in terms of accuracy: 1~1.5 meters in front of 1.5~2.5 meters.

7.2. Future work

The purpose of the project was to have a first approximation with Bluetooth Low Energy in an Indoor Positioning System, studying whether is a good solution or not. In order to have a complete development of an indoor positioning system, there are things that need to be improved and studied in the future.

- Path loss model needs to be further studied in the future. In this project, logdistance model has been used, presenting good results but it can be improved. It would need a complete statistical study of RSSI of BLE and a reconfiguration in the way data is captured and processed.

- Positioning techniques that requires 3 beacons for a 3D trilateration needs to be implemented in order to minimize the cost in a possible deployment of the system. In this project, positioning techniques required 3 beacons for a 2D trilateration and 4 beacons for a 3D trilateration. They have been used instead of the others because of simpler complexity and lower consumption.

- In order to have an IPS that could be adapted depending on situation, different calibration methods must be implemented in the application in order to change its configuration depending on its purpose. Moreover, in case of point calibrations, optimal points for calibration need to be chosen in order to estimate indoor propagation model as good as it could be.

- Beacon placement needs to be evaluated in order to improve accuracy in the system. In this project, there were no criterion in beacon placement, they were placed randomly allowing full coverage in the rooms. However, the optimal placement of beacons should be based on some factors such as coverage, height, and indoor propagation model. There are some researches that discuss the placement of beacons in an indoor positioning system. So, they need to be studied in the future.

- In this project, has been seen that accuracy is reduced as area to be covered is increased if number of beacons is maintained. Positioning estimation depends also on the number of beacons and area to be covered. So, a study of this relationship needs to be done in order to find the optimal number of beacons that gives good results for a specific area. Moreover, it depend on the cost, because having more beacons will give higher accuracy but it will have higher cost too.

Bibliography:

[1] *Bluetooth Core Specification v4.1*. Bluetooth Special Interest Group, June 2013.

[2] R. Heydon. *Bluetooth Low Energy. The Developer's Handbook*, 1st ed. New Jersey, USA: Prentice Hall, 2012.

[3] L.-D. Chou, C.-C. Chang. "A Hierarchical Architecture for Indoor Positioning Services". *Proceedings of the 2nd workshop on positioning, navigation and communication (WPNC'05) and 1st ultra-wideband expert talk (UET'05),* March 2004, Japan. pp. 141-146.

[4] F. Subhan, H. Hasbullah, A. Rozyyev, S. Tahir Bakhsh. "Indoor Positioning in Bluetooth Networks using Fingerprinting and Lateration approach". *International Conference on Information Science and Applications (ICISA), IEEE 2011,* 26-29 April 2011, Jeju Island, Republic of Korea. doi: 10.1109/ICISA.2011.5772436.

[5] A. Baniukevic, C. S. Jensen, H. Lu. "Hybrid Indoor Positioning With Wi-Fi and Bluetooth: Architecture and Performance". *International Conference on Mobile Data Management, IEEE 2013,* 3-6 June 2013. pp. 207-216. doi: 10.1109/MDM.2013.30.

[6] Y. Wang, X. Yang, Y. Zhao, Y. Liu, L. Cuthbert. "Bluetooth Positioning using RSSI and Triangulation Methods". *Consumer Communications and Networking Conference (CCNC), 2013 IEEE,* 11-14 Jan 2013, Las Vegas, USA. doi: 10.1109/CCNC.2013.6488558.

[7] T. M. Fernández, J. Rodas, C. J. Escudero, Daniel I. Iglesia, "Bluetooth Sensor Network Positioning System with Dynamic Calibration". *4th International Symposium on Wireless Communication Systems, 2007 IEEE,* 17-19 Oct 2007, Trondheim, Norway. doi: 10.1109/ISWCS.2007.4392299.

[8] A. K. M. Mahtab Hossain, W.-S. Soh. *"A Comprehensive Study of Bluetooth Signal Parameters for Localization". 18th International Symposium on Personal, Indoor and Mobile Radio Comunications, 2007 IEEE,* 3-7 Sept 2007, Athens, Greek. doi: 10.1109/PIMRC.2007.4394215.

[9] A. Thottam Parameswaran, M. Iftekhar. *"Is RSSI a Reliable Parameter in Sensor Localization Algorithms – An Experimental Study". Field Failure Data Analysis Workshop (F2DA'09), New York, 2009.*

[10] A. Bose, C. H. Foh. *"A Practical Path Loss Model for Indoor WiFi Positioning Enhancement". 6th International Conference on Information, Communications & Signal Processing, 2007 IEEE,* 10-13 Dec 2007, Singapore 2007. pp: 1-6. doi: 10.1109/ICICS.2007.4449717.

[11] J. Rodas, C. J. Escudero. *"Dynamic Path-Loss Estimation Using a Particle Filter". IJCSI International Journal of Computer Science Issues, 7(4), May 2010.*

[12] C. C. Pu, S. Y. Lim, P. C. Ooi. *"Measurement Arrangement for the Estimation of Path Loss Exponent in Wireless Sensor Network". 7th International Conference on Computing and Convergence Technology (ICCCT), 2012 IEEE*. 3-5 Dec 2012, Seoul, South Korea 2012. pp: 807-812.

[13] R. Mehra, A. Singh. *"Real Time RSSI Error Reduction in Distance Estimation Using RLS Algorithm". 3rd International Advance Computing Conference, 2013 IEEE*. 22-23 Feb 2013, Ghaziabad, India. pp: 661-665. doi: 10.1109/IAdCC.2013.6514305.

[14] *Propagation data and prediction methods for the planning of indoor radio communication systems and the radio local area networks in the frequency range 900 MHz to 100 GHz*. ITU-R Recommendations, Geneva, 2001.

[15] L. Pei, R. chen, J. Liu, H. Kuusniemi, T. Tenhunen, Y. Chen. *"Using Inquiry-Based Bluetooth RSSI Probability Distributions for Indoor Positioning"*. *Journal of Global Positioning Systems,* 2010. pp: 122-130.

[16] J. Sommer, S. Lüders, S. Schmitt, W. Rosenstiel. *"System-Level Power-Accuracy Trade-Off in Bluetooth Low Energy Networks". Proceedings of the 8th International Conference on Ubiquitous Intelligence and Computing,* Berlin, Germany 2011. pp: 379-392.

[17] William S. Murphy Jr. "Determination of a Positioning using Approximate Distances and Trilateration". M.S. thesis, Factulty and the Board of Trustees of the Colorado School of Mines, Golden, Colorado, 2007.

[18] K. Dzwinel. "Locating users inside a room with iBeacons". Youtube, 2013. [Online] Available: http://www.youtube.com/watch?v=dMWEl6GBGqk [Accessed: 10 July 2014]

[19] D. S. Benson. "Beacon Hackathon – ASSEMBL demo". Youtube, 2014. [Online] Available: https://www.youtube.com/watch?v=a01eiM4Pwks [Accessed: 10 July 2014]

[20] Quuppa. "Makes High Accuracy Indoor Positioning". In-Location Alliance, 2013. [Online] Available: http://www.in-location-alliance.com/files/ILA/ILA_Munich_Quuppa.pdf [Accessed: 10 July 2014]

[21] GreenPeak Technologies. "In-Home Wireless: The Next Frontier for Industrial Engineers". [Online] Available: http://www.rtcmagazine.com/articles/view/101748 [Accessed: 10 July 2014]

Glossary

ATT: Attribute Protocol

BLE: Bluetooth Low Energy

B-SIG: Bluetooth Special Interest Group

CRC: Cyclic Redundancy Check

CSV: Comma Separated Value

GAP: Generic Access Profile

GATT: Generic Attribute Profile

IPS: Indoor Positioning System

ISM: Industrial, Scientific and Medical radio band

ITU: International Telecommunication Union

L2CAP: Logical Link Control and Adaption Protocol

LAN: Local Area Network

LOS: Line of Sight

LSQ: Least Squares Method

MAC: Media Access Control address

MSE: Mean Square Error

NLOS: Non Line of Sight

NLSQ: Non Linear Least Squares Method

PAN: Personal Area Network

RF: Radio frequency

RSSI: Received Signal Strength Indication

SDK: Software Development Kit

TI: Texas Instruments

UUID: Universally Unique Identifier