Filipe Salgado Almeida

Sistema de Posicionamento de Luz Visível Interior com Redes Neurais Artificiais

Indoor Visible Light Positioning System with Artificial Neural Networks

Universidade de Aveiro
2022

Filipe Salgado Almeida

Sistema de Posicionamento de Luz Visível Interior com Redes Neurais Artificiais

Indoor Visible Light Positioning System with Artificial Neural Networks

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Eletrónica e Telecomunicações, realizada sob a orientação científica do Doutor Luis Alves, Professor Associado do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro, e do Doutor Zabih Ghassemlooy, Professor Catedrático da Universidade de Northumbria

.

o júri / the jury

presidente / president

Professor Doutor António José Ribeiro Neves
Professor Auxiliar, Universidade de Aveiro

vogais / examiners committee

Professor Doutor Luis Filipe Mesquita Nero Moreira Alves
Professor Auxiliar, Universidade de Aveiro

Professor Doutor Vicente Matus Icaza
Investigador de Pós Doutoramento, Universidad de Las Palmas de Gran Canária

agradecimentos / acknowledgements

Desejo exprimir os meus agradecimentos a todos aqueles que, de alguma forma, permitiram que esta dissertação de mestrado se concretizasse.

Ao Professor Doutor Luis Nero Alves, pela sua orientação, total apoio, disponibilidade, pelo saber que transmitiu, pelas opiniões e críticas, total colaboração no solucionar de duvidas e problemas que foram surgindo ao longo da realização deste trabalho e por todas as palavras de incentivo.

Aos meus colegas de curso, do Mestrado Integrado em Engenharia Eletrónica e Telecomunicações, quero agradecer-lhes os momentos, que passámos. Agradeço o bom convívio, as boas discussões e o bom ambiente que se instalava ao nosso redor que me permitiu ultrapassar as dificuldades encontradas ao longo deste ciclo de estudos.

Por último, tendo consciência que sozinho nada disto teria sido possível, dirijo um agradecimento especial aos meus pais pelo seu apoio incondicional, incentivo, amizade e paciência demonstrados e total ajuda na superação dos obstáculos que ao longo desta caminhada foram surgindo.

A todos os meus sinceros agradecimentos.

Palavras Chave

aprendizado de máquina, corrente contínua, comunicações ópticas sem fio, díodos emissores de luz, rede neural.

Resumo

Num ambiente interno, a tecnologia de comunicações ópticas sem fio é usada para criar um sistema de posicionamento usando vários algoritmos de posicionamento e modelos de aprendizado de máquina que estimam a posição de um recetor de fotodíodo que recebe as informações transmitidas por quatro transmissores de díodos emissores de luz. O recetor fotodíode fornece apenas um sinal misturado composto pelo somatório dos quatro sinais enviados pelos quatro transmissores. Nesta dissertação, recorre-se ao uso de equações da transformada rápida de Fourier na análise da potência dos quatro sinais que compõem o sinal misturado, no domínio da frequência, com o objetivo de recuperar o ganho de corrente contínua dos quatro sinais que chegam ao recetor. Os algoritmos de posicionamento utilizam o valor de ganho de corrente contínua de cada uma das componentes singulares do sinal misturado para estimar um valor de posição do recetor, sendo que são implementados, os algoritmos de interpolação com médias pesadas, algoritmos de trilateração com e sem ajustes feitos por algoritmos de aprendizado de máquina e modelos de rede neural que estimam essa posição. A eficiência dos algoritmos é avaliada segundo o valor de erro quando o sistema é afetado por um erro gaussiano. Para o sistema apresentado nesta dissertação, o algoritmo mais eficiente implementado é o algoritmo de trilateração, sem o ajuste de aprendizado de máquina. Este algoritmo produz um erro de posicionamento máximo de 7.31 milímetros, quando a potência do ruído inserida no sistema é nula.

Keywords

direct current, light emitting diodes, machine learning, neural network, optical wireless communications.

Abstract

In an indoor environment, optical wireless communications technology is used to create a positioning system using several positioning algorithms and machine learning models that estimate the position of a photodiode receiver that receives the information transmitted by four light emitting diodes transmitters. The photodiode receiver only provides a mixed signal composed of the sum of the four signals sent by the four transmitters. In this dissertation, we resort to the use of Fourier transform equations in the analysis of the power of the four signals that compose the mixed signal, in the frequency domain, in order to recover the direct current gain of the four signals that arrive at the receiver. The positioning algorithms use the direct current gain value of each of the singular components of the mixed signal to estimate a position value. The position algorithms implemented that estimate the receiver position are, interpolation algorithms with weighted averages, trilateration algorithms with and without machine learning adjustments and neural network model. The efficiency of the algorithms is evaluated by the position error values when the system is affected by Gaussian noise. For the system presented in this dissertation, the most efficient algorithm implemented is the trilateration algorithm, without the machine learning adjustment. This algorithm, produces a maximum positioning error of 7.31 millimeters, when the noise power inserted in the system is zero.

Contents

List of Figures

List of Tables

Glossary

AI	Artificial Intelligence
ANN	Artificial Neural Network
CDF	Cumulative Distribution Function
DC	Direct Current
FFT	Fast Fourier Transform
LED	Light Emitting Diode
ML	Machine Learning
MATLAB	MATrix LABoratory
NN	Neural Network
NLOS	Non Line Of Sight
LOS	Line Of Sight
OWC	Optical Wireless Communication
RF	Radio Frequency
ReLU	Rectified Linear Unit
RMSE	Root Mean Square Error
SNR	Signal to Noise Ratio
SPR	Signal Power Recovery
TSM	Three Steps Method
SVD	Singular Value Decomposition
VLP	Visible Light Positioning

CHAPTER 1

Introduction

Optical Wireless Communication (OWC), is an innovative form of optical communication technologies, that has become essential in the industry during the last decades [1].

OWC systems, transmit data through the free-space channel using light and are a complementary access technology to the Radio Frequency (RF) techniques [2], [3]. RF technologies have limitations regarding to a lower data rate, congested spectrum and an expensive licensing. The licenses to use parts of the spectrum are expensive and the portion of the electromagnetic spectrum used by wireless systems is limited in capacity.

As demand for capacity continues to increase driven by data-heavy wireless communications and new paradigms and devices such as machine-type communication for autonomous systems and smart glasses that utilize augmented reality and virtual reality, the demand for RF spectrum is outstripping supply. Which leads the industry to consider options for using parts of the electromagnetic spectrum. This is where OWC comes to provide innovative solutions by using the optical carriers of visible, infrared, and ultraviolet light to transmit data.

OWC is one of the alternative technologies for indoor and outdoor applications that contribute to high-speed data access to cities, as well as, provide high-speed, cost-effective and highly secure wireless broadband connectivity for other series of applications including voice and data, business connectivity, video and entertainment [3]–[5].

1.1 Scope

The implementation setup, where this dissertation is carried out, takes place in a simulation of an indoor environment where we are working with optical wireless communication systems. In this simulation setup we have 4 Light Emitting Diode (LED), signal transmitting sources, and an optical signal photodiode receiver that detects the optical signal transmitted by the LEDs.

The LEDs used in the scope of this dissertation only provide information in the form of binary codes, 0 or 1, when sending optical signals. These signals propagate in the indoor environment, so they are always subject to propagation losses in the channel. Furthermore, a

signal transmitted in an indoor environment is subject to reflections by obstacles or by surfaces in the middle, such as walls, ceilings, floors at the limits of the environment, which affect the power of the signal received at the receiver and consequently the loss of the transmitted signal.

However, these channel propagation losses can be calculated. Therefore, based on the Direct Current (DC) gain of each of the 4 signal codes, which are detected in the photodiode receiver, it is possible to estimate the position of the photodiode receiver through positioning algorithms. For example, it is possible to estimate the distance between the photodiode receiver and each of the signal transmitting LEDs and based on these 4 distances it is possible to estimate the position of the receiver in relation to the position of the LEDs. The objective of this dissertation is to estimate the position of the photodiode receiver using the information sent by 4 transmitters in the indoor environment.

1.2 Motivation

The signal that is detected in the photodiode receiver is a mixture of the 4 signal codes transmitted by the 4 LEDs that are part of the system. This mixed signal that is provided by the photodiode receiver is composed of the sum of these 4 signals, which are received and detected by the photodiode receiver, after suffering propagation losses in the channel.

To determine the position of the receiver, firstly it is determined the information of the DC gain value of each of the 4 independent signals that are detected in the receiver, using only the information of the mixed signal which is provided by the receiver. Therefore, in a MATrix LABoratory (MATLAB) environment, the signal provided by the photodiode receiver undergoes signal processing, to obtain information about the DC gain of the four signals that are detected in the receiver.

These signal reflections, which make the calculation of the signal propagation path loss more complex, are like unwanted multipath signals in our system. The value of the receiver's DC gain translates into the effect of a successive infinity of reflections of the transmitted signal in the indoor environment. That is, the receiver receives replicas of the signal after it undergoes the action of propagation mechanisms, such as reflection, which make it difficult to calculate the position of the receiver. To solve this problem, machine learning algorithms and techniques, such as, Fingerprint Search are used, which try to cancel the effect of reflections. In a simulated indoor environment, these algorithms and techniques use the information of the DC gain values of the four signals, obtained after the mixed signal being processed in MATLAB, to predict the position of the photodiode receiver In Figure 1.1, is presented the simulation system illustration in a block diagram.

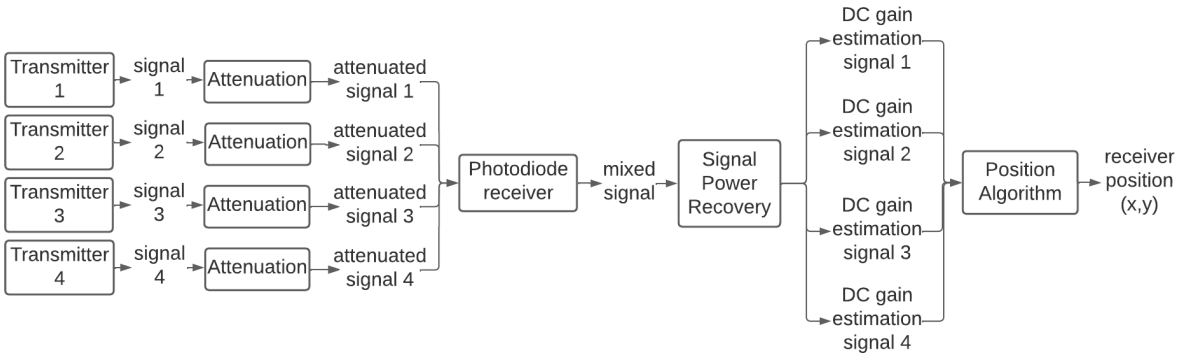

Figure 1.1: System block diagram.

1.3 Dissertation Structure

This dissertation is divided into 4 Chapters. In Chapter 1, the system, problem and objectives of this dissertation are presented. The fundamental knowledge necessary to understand this dissertation is introduced in Chapter 2, where the different algorithms tested in this dissertation are developed, as well as the losses associated with the propagation channel of a signal in the indoor environment are studied. The simulation, tests and the study of these systems are carried out in Chapter 3. Finally, the conclusions found with the tests carried out are presented in Chapter 4, as well as the future work that could be done to improve this work.

CHAPTER 2

Fundamental Concepts

2.1 Signal Propagation In Indoor Environment

The Indoor Optical Wireless Communication system used in this dissertation uses LEDs that act as transmitters and a photodiode as receiver. In this Chapter, the propagation losses of a signal in an indoor environment are studied, so it is necessary to understand the power losses of the signal that is received at the receiver when it is sent by a transmitter LED. However, it is also necessary to understand that in an indoor environment the transmitted signal is subject to reflections caused by obstacles in the environment that affect the power of the received signal at the receiver and consequently the losses of the transmitted signal. That is, the propagation of the signal is dispersed, and the receivers receive replicas of the signal after it undergoes the action of propagation mechanisms, such as reflection, with Non Line Of Sight (NLOS). In this Chapter, the expression that characterizes the value of the power of the signal received at the receiver is presented, taking into account the direct Line Of Sight (LOS) disregarding the reflections to which the signal is subject. It is also presented the expression that characterizes the value of the signal strength received at the receiver, taking into account the NLOS configuration, that is, taking into account the replicas detected by receiver in the indoor environment.

2.1.1 Line of Sight Propagation

Having a LOS configuration of the transmitted signal, the signal propagation loss verified in the receiver is caused by the distance between the transmitter and the receiver and by the transmission and reception angle. In the study of OWC systems, the most important and well known equation that describes the attenuation in the DC gain of the signal received at the receiver is given by Eq. 2.1, according to [6], [7].

$$H_{los}(\psi) = \frac{A_r(m+1)}{2\pi d^2} cos^m(\phi) T_s(\psi) g(\psi) cos(\psi), 0 \leq \psi \leq \psi_c \tag{2.1}$$

Where, the receiver has an active area A_r, has an optical transmission bandpass filter $Ts(\psi)$ and a nonimaging concentrator $g(\psi)$. By Figure 2.1, the receiver is at a distance d and

angle ϕ from the transmitter. Furthermore, m is the Lambert's mode number that expresses the directivity of the source beam.

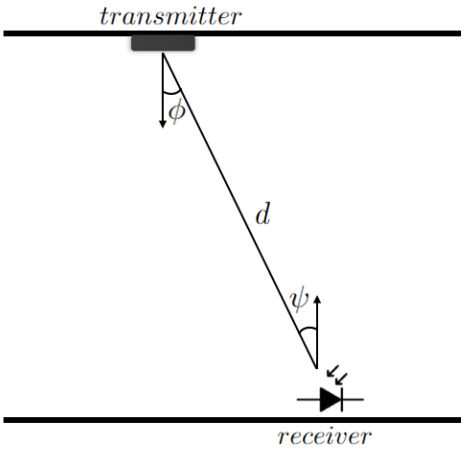

Figure 2.1: Illustration line of sight.

Lambert mode number, m, is related to the semi-angle of the transmitter at half power $\phi_{1/2}$, according to [7], given by,

$$m = \frac{-ln(2)}{ln(cos(\phi_{1/2}))} \tag{2.2}$$

The optical gain of an ideal nonimaging concentrator with refractive index, n, is given by,

$$g(\psi) = \frac{n^2}{sin^2(\psi^2)}, 0 \leq \psi \leq \psi_c \tag{2.3}$$

Knowing that $\psi_c < \frac{\pi}{2}$, so when the value of ψ is larger than ψ_c we have that the $g(\psi)$ is null. Therefore, the value of the receiver's DC gain is also null.

Having the expression of the DC gain of the receiver, then the value of the received signal power, $P_{r_{los}}$, depending on the value of the power transmitted by the transmitter, P_t, is,

$$P_{r_{los}} = H_{los} P_t \tag{2.4}$$

2.1.2 Non Line of Sight Propagation

In a NLOS configuration, we have to take into account the reflections caused by the surface of the room and by objects in the indoor environment. These signal reflections, which make the calculation of propagation loss in the path more complex, are like unwanted multipath signals in our system. Therefore, the value of the receiver's DC gain in this NLOS configuration translates into the effect of a successive infinity of reflections of the signal transmitted in the indoor environment. However, in this dissertation it is not intended to accurately calculate these reflections, only to study their influence on the increase in the complexity of calculating the losses in the propagation channel. So the system model is based on the influence of all the first reflections caused by the surface of the room walls that are detected by the receiver with $\psi < \psi_c$, as can be seen in Figure 2.2 for one of the possible positions of the receiver, on

the surface of the room, where the reflected signal is detected in the receiver with a non-zero value.

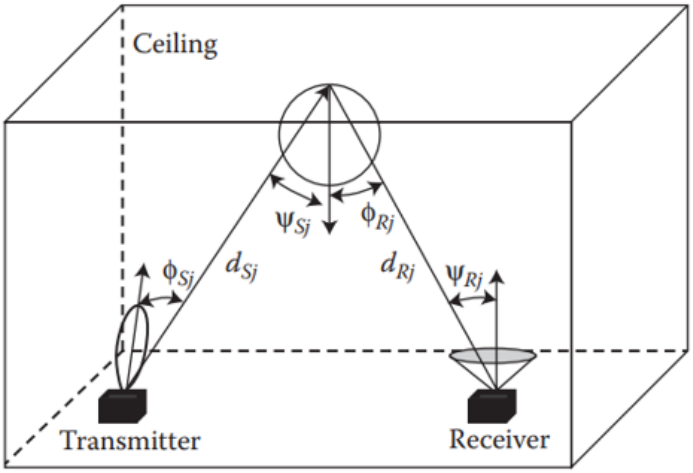

Figure 2.2: Illustration non line of sight with reflection in surface area of the ceiling [7].

Thus, for a transmitter and receiver position, according to [7], the DC gain of the NLOS configuration, H_{nlos}, after a reflection, in N_a points, on a surface of the room, is given by,

$$H_{nlos} = \sum_{j=1}^{N_a} \frac{(m+1)\rho_j A_r \Delta A}{2\pi d_{Sj}^2 d_{R_x j}} cos^m(\phi_{Sj}) cos(\psi_{Sj}) cos(\psi_{Rj}) \tag{2.5}$$

Where, the reflective surface of the room is subdivided into N_a subareas comprising each of these N_a reflecting elements with a value area ΔA and reflection coefficient ρ_j. Furthermore, d_{Sj} is the distance between the transmitter and the reflecting element j on the surface of the room. However $d_{R_x j}$ is the distance between the position R_x of the receiver and the reflector element j on the surface of the room.

Knowing the expression of the receiver's DC gain in an NLOS and LOS configuration, then, the value of the received signal strength, $P_{r_{nlos}}$, depending on the P_t, is,

$$P_{r_{nlos}} = (H_{los} + H_{nlos}) P_t \tag{2.6}$$

2.2 Signal Processing Methods

In a three dimensions global positioning system, four transmitters are needed to calculate the spatial coordinates of an object or receiver. Each distance difference between transmitter and receiver corresponds to a hyperboloid of revolution. The line joining the transmitter and receiver forms the axis of the hyperboloid, and the receiver is located at the point where three hyperboloids intersect. So much so that 4 transmitters are needed to calculate the position in three dimensions spaces. However, in this dissertation, 4 transmitter will be used in a simulation to calculate the position of a receiver in two dimension space.

Thus, having a simulated indoor system that uses a photodiode as a receiver and four transmitter LEDs, then the signal that is received by the receiver is a mixture of the four

signals sent by each of the transmitters. This mixed signal, S_{mix}, which is received by the receiver at a given position is given by,

$$S_{mix} = \sum_{i=1}^{N_t} M_i P_{t_i} \tag{2.7}$$

Where, i is the index of N_t transmitters used and M_i represents, in an NLOS configuration, the DC gain attenuation value in the receiver of the signal that is sent by each of the transmitter i.

So, M is a matrix of N_t elements that contain each of the DC gain attenuation values at the receiver of each of the N_t transmitted signals, being represented as,

$$M(i) = \begin{bmatrix} H_{los_1} + H_{nlos_1} & H_{los_2} + H_{nlos_2} & \ddots & H_{los_i} + H_{nlos_i} & \ddots & H_{los_{N_t}} + H_{nlos_{N_t}} \end{bmatrix} \tag{2.8}$$

The value of each element of the matrix M is obtained by the expression,

$$M_i = H_{los_i} + H_{nlos_i}, i \in \mathbb{R}^*_+ \tag{2.9}$$

Where, the sum of H_{los_i} and H_{nlos_i} represents, in the NLOS configuration, the DC gain attenuation of the signal i detected by the receiver when this signal i is sent by the transmitter i.

Note that in this simulated system we only have access to the output of the receiver, the mixed signal S_{mix}. Therefore, it is necessary to know the DC gain of each signal, that compose the mixed signal detected by the receiver, that is sent by each of the four transmitters in order to predict the position of the photodiode receiver.

Therefore, in this dissertation we intend to use trilateration, interpolation and machine learning algorithm that predict the position of the photodiode receiver based on the data after the proper processing of the mixed signal that is received at the photodiode receiver.

To obtain the individual signals that compose the mixed signal, can be used methods, such as, Singular Value Decomposition (SVD) that decomposes a matrix into its single values, that is, this method allows the decomposition of a multivariate signal into independent signals [8]–[10]. This is an approach to factoring the matrix into a series of linear approximations that expose the underlying structure and discover the redundancy of the matrix.

However, with this objective of processing this signal, one method is thought and studied. This method, Signal Power Recovery (SPR), analyzes the mixed signal in the frequency domain to return the DC gain of the original signals that compose the mixed signal.

2.2.1 Fast Fourier Transform

This Section has been devoted to Fast Fourier Transform (FFT), applied to a square wave signal, in order to provide prior content to the SPR, in 2.2.2, described in the following Section.

Fourier analysis converts a signal from the original domain to a representation in the frequency domain, which allows to visualize the same signal in the time domain as well as

in the frequency domain [11]. This brings several applications and, in the case applied to this thesis, it will be used to visualize a signal that is the sum of four square wave signals with different frequencies. In MATLAB the function *fft(X)* calculates the discrete Fourier transform of the signal X using a fast Fourier transform algorithm.

The signal X consists of N temporal samples, equally spaced from T, being T the sampling interval. The FFT algorithm allows calculate $\frac{2}{N}$ spectrum components contained in the range $0 \leq \omega \leq \frac{\pi}{T}$, from $N/2$ the spectrum repeats with a mirroring. In the frequency domain, the sample spacing, $\Delta\omega$, is given by,

$$\Delta\omega = \frac{\frac{\pi}{T}}{\frac{N}{2}} = \frac{2\pi}{NT} \tag{2.10}$$

Where every harmonic, $h = \{0, 1, 2, ..., N-1\}$, , is at the frequency ω_h, given by,

$$\omega_h = \frac{2\pi h}{NT} \tag{2.11}$$

If the discrete Fourier transform of the sampled signal $x(n)$ is $X(k)$, such that,

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}kn} \tag{2.12}$$

And,

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi}{N}kn} \tag{2.13}$$

Illustrating in MATLAB environment, $x(t)$ is a square wave signal of amplitude between 0 and 1 and a frequency $f1 = 5$ Hz. $x(n)$ is the discrete sign of $x(t)$ and is sampled at a frequency of 200 Hz, so it has $N = 40$ samples per signal period. The function *fft(X)*, from MATLAB, gives the discrete Fourier transform of $x(t)$ in order of frequency, in Hz, given by $X(f)$, as shown in Figure 2.3 and in Figure 2.4.

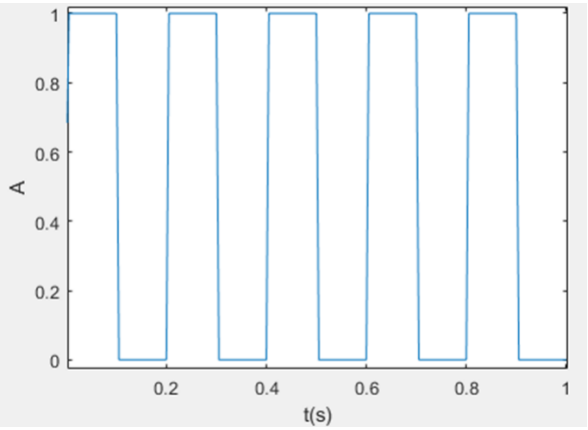

Figure 2.3: Signal x(t) in the time domain.

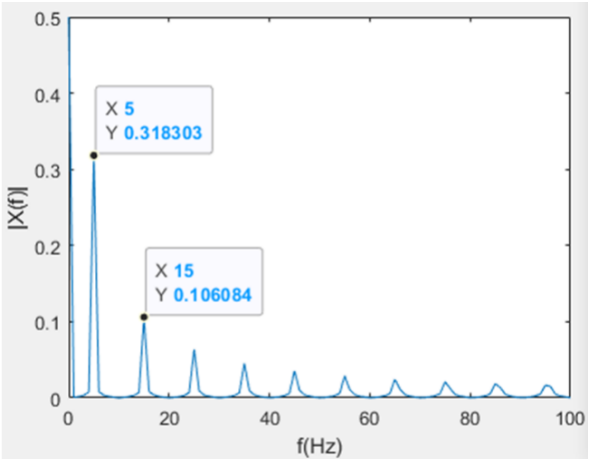

Figure 2.4: Signal x(t) in the frequency domain.

In Figure 2.4, it can be seen that the first harmonic is at the signal frequency, $f1 = 5$ Hz, and the spacing between the odd harmonics is $2 \times f1$. Note that only odd harmonics are shown.

It can also be seen in Figure 2.5, the discrete Fourier transform of $x(t)$ in order to $w(rad/s)$, where the first harmonic, according to Eq. 2.11, is at the frequency $w_1 \approx 0.785(rad/s)$, and the spacing between the harmonics is $2 \times \Delta\omega = 0.5\pi(rad/s)$.

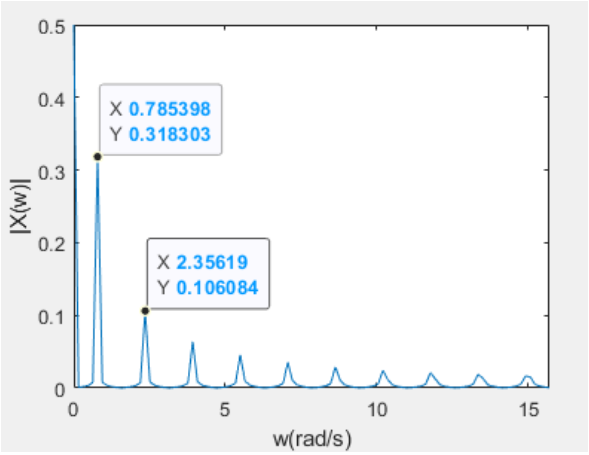

Figure 2.5: X(w) in order to w(π/s).

Thus, the odd harmonics, ω_{2r+1}, com $r = \{0, 1, 2, ...\frac{N-1}{2}\}$, are given by,

$$\omega_{2r+1} = \frac{2\pi(2r+1)}{NT}, \tag{2.14}$$

2.2.2 Signal Power Recovery

Briefly, as the system consists of four signal transmitting sources, which operate at different frequencies, the information provided by the receiver, in any position in the room, is a mixture of signals that translates into the sum of the four signals, sent by the transmitters after suffered losses in the propagation channel, according to Eq. 2.6, as explained in 2.1.2.

This first part of method 2.3.1 solves the definite problem of knowing how it is possible to retrieve the information on the value of the amplitudes of the different signals arriving at the receiver based on the mixed signal that is provided by the receiver.

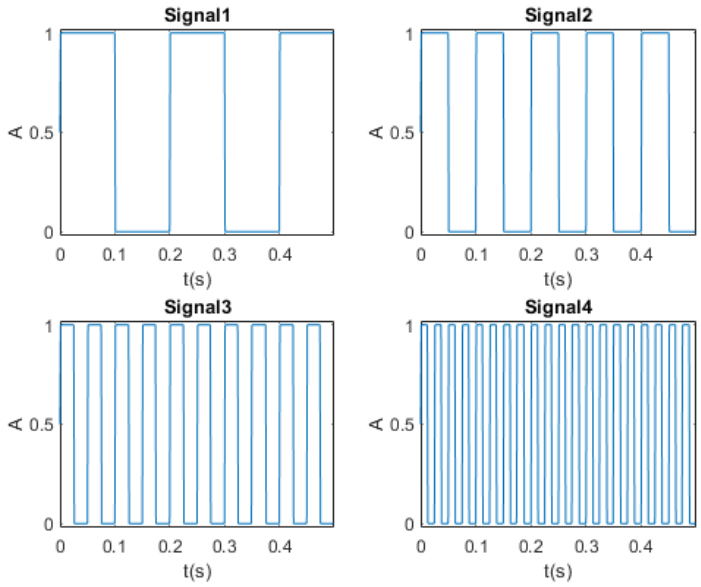

Figure 2.6: Received signals in photodiode receiver.

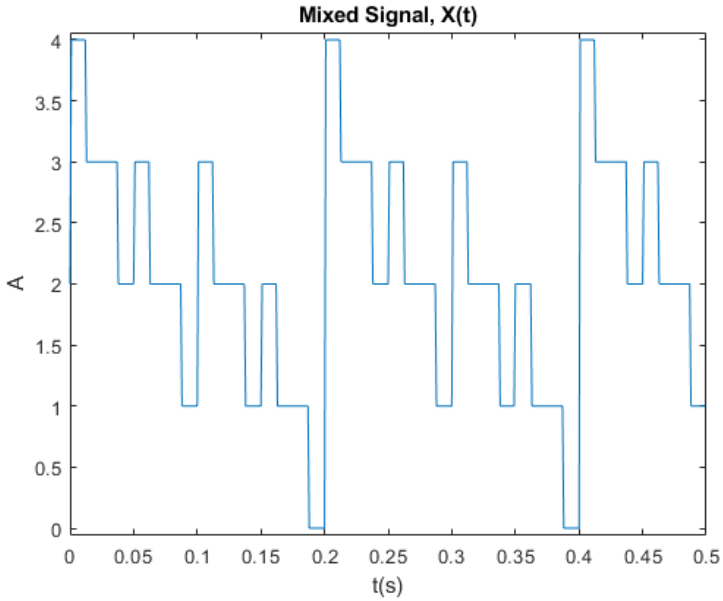

Figure 2.7: Signal provided by the photodiode receiver.

The Figures 2.6 and 2.7 show, respectively, the signals received at the photodiode receiver at any position in the room and the signal supplied by the receiver, denominated mixed signal, which is the sum of the four signals received at the receiver, shown in Figure 2.6.

The mixed signal presented in the time domain does not visually present a clear distinction between the different signals, as can be seen in Figure 2.7. However, the mixed signal in the

frequency domain, shown in Figure 2.8, already presents a distinction of the four signals, so it is possible to separate them in the frequency domain.

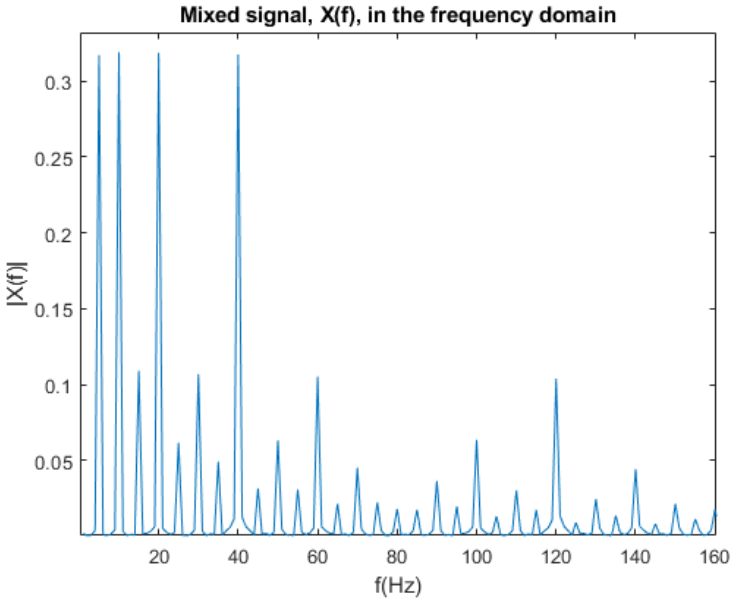

Figure 2.8: Signal provided by the receiver in the frequency domain.

The position of harmonics in the Figure 2.8 is explained using the explanation given in 2.2.1.

Thus, in the frequency domain it is possible to determine, through the Eq. 2.14, the position of each of the harmonics associated with each of the signals that make up the mixed signal. Being f_i the frequency of each signal i received at the receiver and i takes values $1, 2, 3$ and 4, representing each of the signals received at the receiver. As explained in 2.2.1, the odd harmonics of each signal i are spaced by $2 \times f_i$ and the first harmonic of each signal i is at the frequency f_i, being distanced by $2 \times f_i$ of the second harmonic. By establishing a relationship between the corresponding power value of the harmonics and the amplitude value of each of the received signals, the receiver, for different receiver positions in the room, the model that best describes the behavior of the data used is determined. The model created, based on the power value of the harmonics that correspond to the signal i, provides an estimate of the signal amplitude value i received at the receiver.

Thus, the SPR technique provides an estimate of the amplitude value of the four signals received at the receiver.

In Figure 2.9 is presented the SPR algorithm using a block diagram.

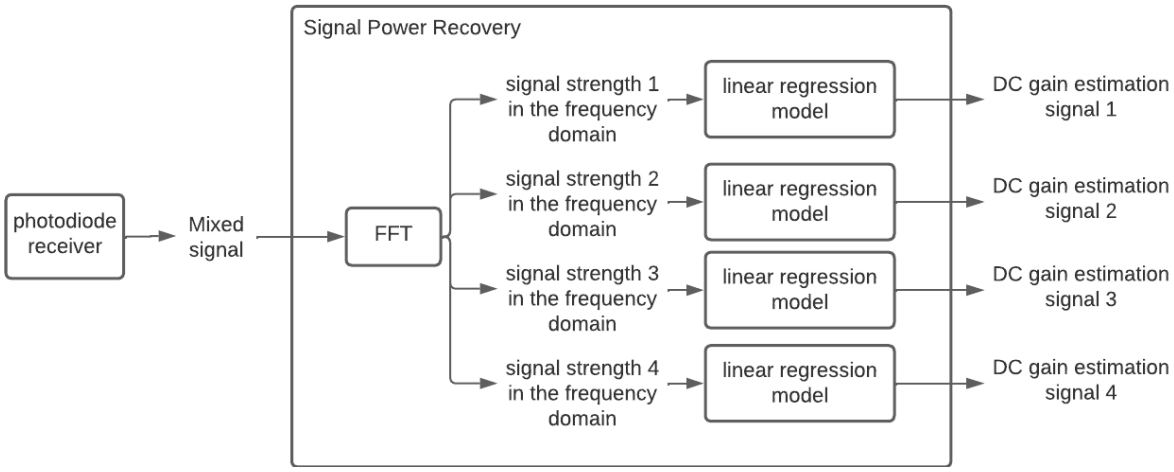

Figure 2.9: Signal power recovery block diagram.

2.3 Position Algorithm Methods

In this Section are presented methods that allow the calculation or prediction of the position of the receiver, in simulation. These positioning methods are implemented in series with the SPR method, so they use the DC gain information of each of the signals that compose the mixed signal to estimate the position of the receiver. Here, the methods presented are: Three Steps Method (TSM), Trilateration with transmitters and methods implemented with Machine Learning (ML).

2.3.1 Three Steps Method

This Section presents a method called the TSM, which is described in three sequential parts. First, we have the SPR technique that allows us to discover the amplitude of each signal that is received in the photodiode receiver, when it is placed at any point in the room, then a technique called Fingerprint Search is used that provides four points that delimit the region, more smaller than the area of the room, where the receiver is located and finally, based on this smaller region, two positioning algorithms, Interpolation with Weighted average and Trilateration, that estimate the coordinates of the position where the photodiode receiver is located.

Fingerprint Search

Fingerprints are unique to each individual, being distinct even between identical twins. This characteristic, called uniqueness, makes them used as a way of identifying people for centuries. There are many approaches to Visible Light Positioning (VLP), such as, trilateration and structured illumination provided by micro-LED arrays. The structured illumination can be used to estimate the position based on fingerprint signals provided to a spatial grid [12]–[14]. Where, micro-LED array, is generated to a tracking area, with each pixel illuminating a spatially detach area.

Similarly, Fingerprint Search performs a search on a matrix structure of unique values, which characterize a given object or resource, to obtain a region where the photodiode receptor

is located. This matrix structure is called a Fingerprint map and contains the amplitude value of each of the four signals received at the receiver, estimated by the SPR technique presented above, and which characterize and identify the position of the photodiode receiver. The Fingerprint map represents a map of evenly spaced points on the floor of the room where we can place the photodiode receiver, which contain information on the power values of the signals detected in the receiver at that coordinate point on the floor of the room. Thus, the Fingerprint search technique provides the four points that define the limits of the smallest square region where the photodiode receptor is located through the search carried out on the Fingerprint map.

This Fingerprint search technique is applied in sequence to the SPR technique, presented above, and the points that compose the fingerprint map work as a database. This database, previously created, is then used as a search that takes as a search parameter the power vector provided by the SPR technique, associated with any position of the receiver, and provides the four points on the map that define the coordinates of the vertices of the smallest region where the photodiode receptor is located.

In order to define this square region delimited by four coordinate points belonging to the Fingerprint map, the referred research is carried out using a previously trained NN model. This model estimates the distance, in meters, between two power vectors through the difference of two power vectors and the values of the two vectors. As already mentioned, for each position of the photodiode receiver, four signals are detected whose amplitude value of these four signals constitutes a power vector that characterizes the position of the photodiode receiver. So the data used for training the model as 13 features, being that, each power vector is characterized by 4 features, the difference between the vectors is characterized by 4 features and 1 features characterizes the distance between the vectors. Since the Fingerprint map is a matrix of dimensions $m \times n$, then to create the data used for training the NN model based on these $m \times n$ power vectors, are calculated all combinations of difference between power vectors and the distance, in meters, between these power vectors. Thus, the data used for training the model has $(m \times n)^2$ elements that represent the relationship between the difference and the distance between two power vectors.

In the Fingerprint search technique, with the distance between two vectors being estimated by the NN model, the search algorithm developed, firstly, is limited to finding the 3 points on the Fingerprint map whose power vector minimizes the distance to the power vector measured at photodiode receiver position, that is, get the 3 closest coordinate points to the receiver position. In other words, firstly, the NN model is used to estimate the distance between the receiver and the map points, according to the respective power vectors, and the search algorithm looks for the 3 points that minimize this distance. The algorithm, firstly, only looks for 3 point algorithm in order to guarantee that the region is a square because the four closest point to the receiver do not necessarily represent a square region.

Then, based on these 3 points already determined, the fourth point of the Fingerprint map is determined, which allows obtaining a smaller square region where the receiver is located. To this end, in the second and final part of the search algorithm, using the NN model again, the

point on the Fingerprint map closest to the other 3 points already chosen is determined. In other words, the NN model is now used to determine the distance between all the remaining m × n - 3 points on the Fingerprint map and the 3 points on the map already determined, from the respective power values. The search is limited to looking for the point that minimizes the sum of these distances, that is, the closest point to the other 3 already defined.

In Figure 2.10, the three points already defined in the first part of the research are represented in green and the fourth point that defines the desired square region is represented in blue, whose sum of the distances between them and the three green points, within all the points on the map that have not yet been chosen, is smaller.

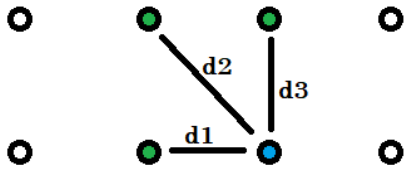

Figure 2.10: Second phase of the search algorithm, fourth point represented in blue and the three points in the fingerprint map obtained in the first phase of search.

Therefore, the Fingerprint search technique determines the 4 points that define the smallest square region where the receiver is located, through a point map called Fingerprint map. Note that in order to determine the receiver region at any position in the room, the Fingerprint map must have at least four points that define the corners or boundaries of the room.

In Figure 2.11 is presented the Fingerprint Search algorithm using a block diagram.

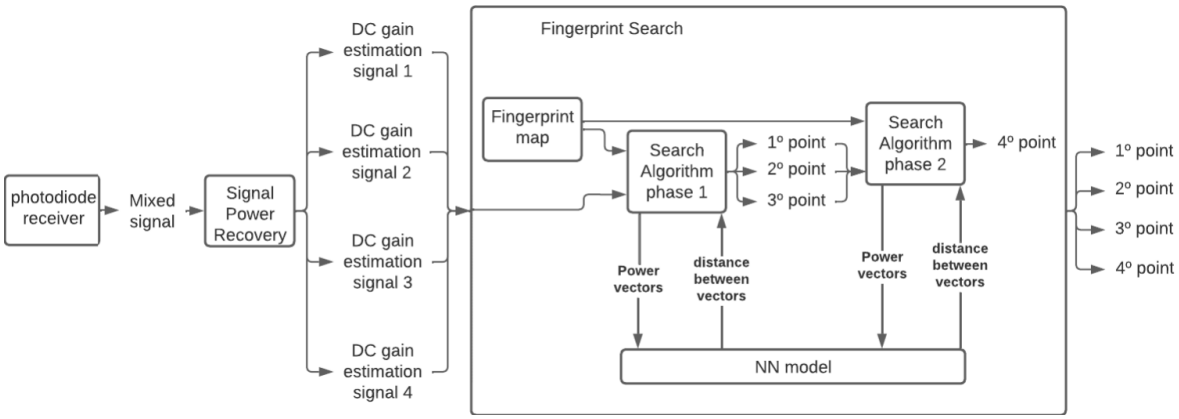

Figure 2.11: Fingerprint search block diagram.

Positioning Algorithms

Finally, we have the two 2D positioning algorithms that estimate the position of the photodiode receiver within the region provided by the Fingerprint Search algorithm. There are some positioning algorithms like interpolation [15] and trilateration [16], [17] that are usually used to determined position.

Therefore, the two position algorithms developed are Interpolation with Weighted Averages and Trilateration.

Interpolation With Weighted Averages

This algorithm allows estimating the coordinates X and Y,that define the position of the receiver, and are calculated according to the two equations, Eq. 2.15 and Eq. 2.16, presented below.

$$X = \frac{\sum_{k=1}^{4}(cos(\frac{\pi}{2} \times \frac{d_k}{d_{max}}) \times X_k)}{\sum_{k=1}^{4}(cos(\frac{\pi}{2} \times \frac{d_k}{d_{max}}))} \tag{2.15}$$

$$Y = \frac{\sum_{k=1}^{4}(cos(\frac{\pi}{2} \times \frac{d_k}{d_{max}}) \times Y_k)}{\sum_{k=1}^{4}(cos(\frac{\pi}{2} \times \frac{d_k}{d_{max}}))} \tag{2.16}$$

Where, d_k is the distance between the power vector estimated by the technique SPR, with powers $S_p^{(i)}$, and the vertices power vector k, with powers $S_k^{(i)}$, with $i = \{1, 2, 3, 4\}$ representing respectively each one of the four signals received at the receiver. The value $d_{max} = max(d_k)$, $k = \{1, 2, 3, 4\}$. Thus, d_k is presented by Eq. 2.17 below.

$$d_k = sqrt((S_k^{(1)} - S_p^{(1)})^2 + (S_k^{(2)} - S_p^{(2)})^2 + (S_k^{(3)} - S_p^{(3)})^2 + (S_k^{(4)} - S_p^{(4)})^2) \tag{2.17}$$

Being the weight of the equations, which define the coordinates X and Y, is defined by cosine, so when the distance value d_k is smaller means that the node k is closer to the receiver, and the weight associated with the coordinates X_k and Y_k is the highest.

In this dissertation we will also test the weighted average algorithm with a weight value now defined by the exponential, $e^{-\frac{d_k}{d_{max}} \times n}$. Where n takes values in the real domain. The larger the n the greater the range of values to which the weight can take. If $n > 0$, so, $e^{-\frac{d_k}{d_{max}} \times n} \in]0, 1]$.

In this way the coordinates X and Y are estimated, respectively, by Eq. 2.18 and Eq. 2.19, presented as follows,

$$X = \frac{\sum_{k=1}^{4}(\exp(-\frac{d_k}{d_{max}} \times n) \times X_k)}{\sum_{k=1}^{4}(\exp(-\frac{d_k}{d_{max}} \times n))}, n > 0 \tag{2.18}$$

$$Y = \frac{\sum_{k=1}^{4}(\exp(-\frac{d_k}{d_{max}} \times n) \times Y_k)}{\sum_{k=1}^{4}(\exp(-\frac{d_k}{d_{max}} \times n))}, n > 0 \tag{2.19}$$

Trilateration

This algorithm allows estimating the coordinates X and Y,that define the position of the receiver based on the power vector estimated by the SPR technique, Q, in position (X, Y), and the power vectors, Q_k, of the vertices k, with $k = \{1, 2, 3, 4\}$, provided by the Fingerprint search algorithm, in position (X_k, Y_k). The Figure 2.12, illustrates these points.

In Figure 2.12, r_k represents the distance between the position (X_k, Y_k) of the vertice k and the position (X, Y) of the receiver and is represented according to Eq. 2.20.

$$r_k^2 = (X - X_k)^2 + (Y - Y_k)^2 \tag{2.20}$$

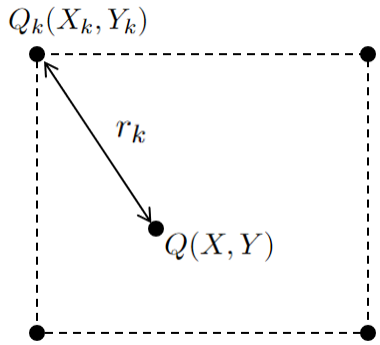

Figure 2.12: Illustration of trilateration algorithm.

In Figure 2.13, the system is illustrated in three dimensions. Where we have represented the transmitter T^i in position (X_T^i, Y_T^i, h), i takes values 1, 2, 3 and 4 representing the four transmitters and h is the height of the room. The distances d_k^i and d^i represent, respectively, the distance between the transmitter i, in position (X_T^i, Y_T^i, h), and the vertice k, in position $(X_k, Y_k, 0)$, the distance between the transmitter i, in position (X_T^i, Y_T^i, h), and the receiver position $(X, Y, 0)$. The distances r^i and r_k^i represent, respectively, the distance between the transmitter position at ground level, $(X_T^i, Y_T^i, 0)$, and the receiver, in position $(X, Y, 0)$, and the distance between the transmitter position at ground level, $(X_T^i, Y_T^i, 0)$, and the vertice k, in position $(X_k, Y_k, 0)$.

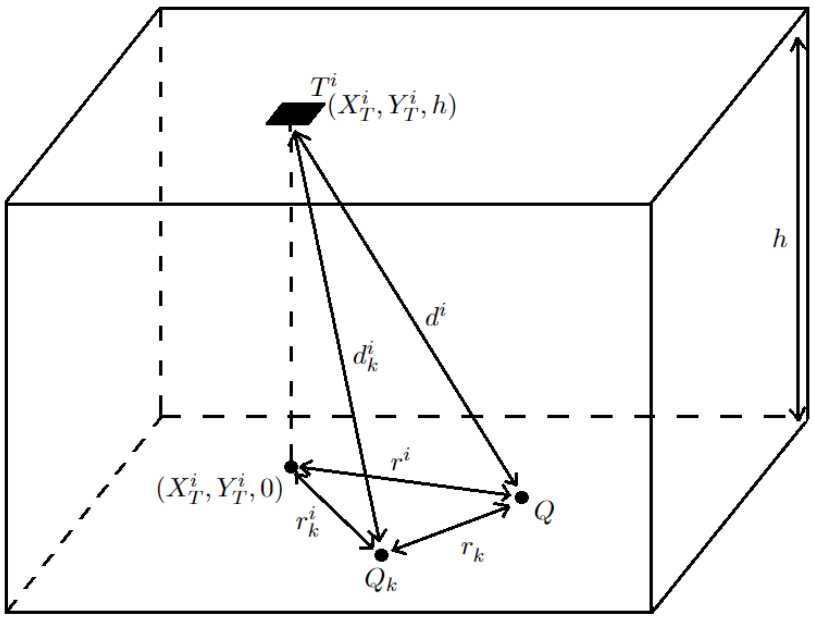

Figure 2.13: Illustration 3D of trilateration algorithm.

Furthermore, according to the similarity relations of the sides of a triangle we obtain that,

$$r_k^2 = (r^i)^2 + (r^k)^2 - 2 < r^i \cdot r_k^i > \tag{2.21}$$

Where, $(r^i)^2$ and $(r_k^i)^2$ are estimated values from Eq. 2.4 and $< r^i \cdot r_k^i >$ is such that,

$$< r^i \cdot r_k^i >= (X - X_T^i)(X_k - X_T^i) + (Y - Y_T^i)(Y_k - Y_T^i) \tag{2.22}$$

Deducting the value of $(r^i)^2$, assuming that $T(\psi) = 1$, $g(\psi) = 1$ and $P_{r_k}^i$ is the signal strength i estimated at the receiver at the position $Q(X, Y)$, is started by,

$$P_r^i = \frac{A_r(m+1)}{2\pi} P_t^i cos^m(\phi) cos(\psi), 0 \leq \psi \leq \psi_c \tag{2.23}$$

Where, $cos^m(\phi) = \frac{h^m}{d^m}$ and $cos(\psi) = \frac{h}{d}$, such that,

$$P_r^i = \frac{A_r(m+1)}{2\pi} P_t^i \frac{h^{m+1}}{d^{i(m+3)}} \tag{2.24}$$

So, if $d^{i(m+3)} = ((r^i)^2 h^2)^{\frac{m+3}{2}}$, then,

$$(r^i)^2 = (\frac{A_r(m+1)}{2\pi} \frac{P_t^i}{P_r^i} h^{m+1})^{\frac{2}{m+3}} - h^2 \tag{2.25}$$

In the same way, it is obtained that $(r_k^i)^2$ is such that,

$$(r_k^i)^2 = (\frac{A_r(m+1)}{2\pi} \frac{P_t^i}{P_{r_k}^i} h^{m+1})^{\frac{2}{m+3}} - h^2 \tag{2.26}$$

Where, $P_{r_k}^i$ is the signal strength i estimated at the receiver at the node position $Q_k(X_k, Y_k)$. Substituting Eq. 2.31 into Eq. 2.30, it is obtained that,

$$X^2 - 2XX_T^i + Y^2 - 2YY_T^i = b_k^i \tag{2.27}$$

Where, b_k^i, is such as,

$$b_k^i = (r^i)^2 + (r_k^i)^2 - (X_k)^2 + 2X_T^i X_k - 2(X_T^i)^2 - (Y_k)^2 + 2Y_T^i Y_k - 2(Y_T^i)^2 \tag{2.28}$$

The Eq. 2.27 obtained is of second order, for $k = \{1, 2, 3, 4\}$ and $i = \{1, 2, 3, 4\}$. To eliminate the second order component X^2 and Y^2, it can be perform a simple manipulation on the equations which is based on subtracting Eq. 2.27 by Eq. 2.27 for $k = 4$ and $i = 4$. However, subtracting, for example, Eq. 2.27 for $k = 1$ and $i = 4$, by Eq. 2.27 for $k = 4$ and $i = 4$, the obtained result is null. For this reason, it is necessary to assign values to i such that the subtraction between the two equations is done for values of i distinct. Thus, it is assumed $i = k$.

Now, solving Eq. 2.27 using the aforementioned manipulation, subtracting from Eq. 2.27 to $k = 4$ and $i = 4$, it is obtained that,

$$-2(X_T^k - X_T^{k_4})X - 2(Y_T^k - Y_T^{k_4})Y = b_k \tag{2.29}$$

Where, b_k is given by Eq. 2.30, as followed,

$$
\begin{aligned}
b_k &= (r^k)^2 + (r_k^k)^2 - (r^{k_4})^2 - (r_{k_4}^{k_4})^2 - (X_k)^2 + (X_{k_4})^2 + 2X_T^k X_k - \\
&\quad - 2X_T^{k_4} X_{k_4} - 2(X_T^k)^2 + 2(X_T^{k_4})^2 - (Y_k)^2 + (Y_{k_4})^2 + 2Y_T^k Y_k - \\
&\quad - 2Y_T^{k_4} Y_{k_4} - 2(Y_t^k)^2 + 2(Y_t^{k_4})^2
\end{aligned} \tag{2.30}
$$

Where, $k = \{1, 2, 3, 4\}$. However, for $k = 4$ the solution is zero. In this way we obtain a system of three equations, given by Eq. 2.31, to estimate the position $Q(X, Y)$ of the receiver, such that,

$$
\begin{cases}
\text{-2}(\text{X}_T^{k_1} - X_T^{k_4})X - 2(Y_T^{k_1} - Y_T^{k_4})Y = b_{k1} \\
\text{-2}(\text{X}_T^{k_2} - X_T^{k_4})X - 2(Y_T^{k_2} - Y_T^{k_4})Y = b_{k2} \\
\text{-2}(\text{X}_T^{k_3} - X_T^{k_4})X - 2(Y_T^{k_3} - Y_T^{k_4})Y = b_{k3}
\end{cases} \tag{2.31}
$$

Where, b_k is the entire expression on the second member of Eq. 2.29 for $k = 1$, 2 and 3.

Therefore, these three equations can be written in a matrix of the form $AQ = B$, such that,

$$
A = \begin{bmatrix}
-2(X_T^{k_1} - X_T^{k_4}) & -2(Y_T^{k_1} - Y_T^{k_4}) \\
-2(X_T^{k_2} - X_T^{k_4}) & -2(Y_T^{k_2} - Y_T^{k_4}) \\
-2(X_T^{k_3} - X_T^{k_4}) & -2(Y_T^{k_3} - Y_T^{k_4})
\end{bmatrix} \tag{2.32}
$$

$$
Q = \begin{bmatrix} X \\ Y \end{bmatrix} \tag{2.33}
$$

$$
B = \begin{bmatrix} b_k 1 \\ b_k 2 \\ b_k 3 \end{bmatrix} \tag{2.34}
$$

Solving the equation $AQ = B$ in order to Q it is obtained that,

$$
\begin{aligned}
&AQ = B \\
&\Leftrightarrow (A^T A)Q = A^T B \\
&\Leftrightarrow (A^T A)^{-1}(A^T A)Q = (A^T A)^{-1} A^T B, (A^T A)^{-1}(A^T A) = I
\end{aligned} \tag{2.35}
$$

Where I is the identity matrix, then the equation as follows is obtained.

$$
Q = (A^T A)^{-1} A^T B \tag{2.36}
$$

Thus, using Eq. 2.36, it is possible through the trilateration algorithm to calculate the coordinates (X, Y) of the receiver's position.

2.3.2 Trilateration With Transmitters

This algorithm allows estimating the coordinates X and Y,that define the position of the receiver in a position (X, Y) based on the power vector estimated by the SPR technique, in position $Q(X, Y)$, and transmitter positions T_t in the position (X_t, Y_t, h), $t = \{1, 2, 3, 4\}$. The Figure 2.14 illustrates these points.

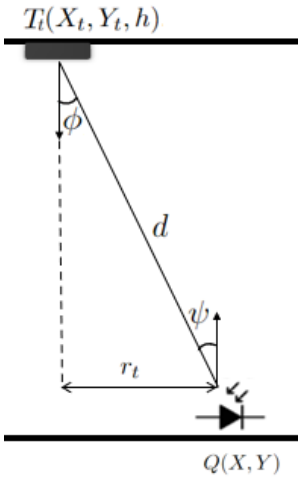

Figure 2.14: Illustration of trilateration algorithm with transmitters positions.

In Figure 2.12, r_t represents the distance between the position $(X_t, Y_t, 0)$ of the transmitter T_t and the position (X, Y) of the receiver and is represented according to Eq. 2.37.

$$r_t^2 = (X - X_t)^2 + (Y - Y_t)^2 \tag{2.37}$$

Where, r_t can be approximated according to Eq. 2.4. Assuming that $T(\psi) = 1$, $g(\psi) = 1$ and P_{r_t} is the signal strength t estimated at the receiver in position $Q(X, Y)$, then in the same way as in the deduction of Eq. 2.25, r_t^2 is such that,

$$r_t^2 = (\frac{A_r(m+1)}{2\pi}\frac{P_t^i}{P_{r_t}^i}h^{m+1})^{\frac{2}{m+3}} - h^2 \tag{2.38}$$

The Eq. 2.37 obtained is of second order, for $t = \{1, 2, 3, 4\}$. To eliminate the second order component one can perform the manipulation of subtraction of Eq. 2.37 by Eq. 2.37 for $t = 4$. Solving Eq. 2.37 using the aforementioned manipulation, we obtain that three equations to estimate the position $Q(X, Y)$ of the receiver, in the form $-2(X_t - X_4)X - 2(Y_t - Y_4)Y = r_t^2 - r_4^2$, for $t = \{1, 2, 3\}$, such that,

$$\begin{cases} \text{-2}(\text{X}_1 - X_4)X - 2(Y_1 - Y_4)Y = r_1^2 - r_4^2 \\ \text{-2}(\text{X}_2 - X_4)X - 2(Y_2 - Y_4)Y = r_2^2 - r_4^2 \\ \text{-2}(\text{X}_3 - X_4)X - 2(Y_3 - Y_4)Y = r_3^2 - r_4^2 \end{cases} \tag{2.39}$$

Therefore, these three equations can be written in a matrix of the form $AQ = B$, such that,

$$A = \begin{bmatrix} -2(X_1 - X_4) & -2(Y_1 - Y_4) \\ -2(X_2 - X_4) & -2(Y_2 - Y_4) \\ -2(X_3 - X_4) & -2(Y_3 - Y_4) \end{bmatrix} \tag{2.40}$$

$$Q = \begin{bmatrix} X \\ Y \end{bmatrix} \tag{2.41}$$

$$B = \begin{bmatrix} r_1^2 - r_4^2 \\ r_2^2 - r_4^2 \\ r_3^2 - r_4^2 \end{bmatrix} \tag{2.42}$$

Therefore, using Eq. 2.36, it is possible through the trilateration algorithm, based on the positions of the transmitters, to calculate the coordinates (X, Y) of the receiver's position.

2.3.3 Machine Learning

ML emerged from the search for Artificial Intelligence (AI) and is a type of AI which, as the name implies, is understood as the ability of computers to learn without being explicitly programmed. ML algorithms are algorithms that can learn from their mistakes using historical data as input to predict new output values and thus improve performance in some set of tasks. That is, making predictions or decisions guided by a pre-defined database.

ML algorithms are used to perform computational tasks where creating and developing explicit algorithms is difficult or impractical as in speech recognition and computer vision [18], [19], or even in medicine in disease detection and forecast predictions [20].

ML approaches are divided into four forms of computational learning, namely supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning [21].

The Supervised learning algorithms build a mathematical model that is trained on a dataset that contains labeled inputs and desired outputs. The data is known as the training database of the ML model and consists of a set of training examples with the desired input and output, represented by a matrix. Through iterative optimization of a cost function, ML algorithms learn a function that can be used to predict the output associated with new inputs. The cost function is a function that performs the quantification of a cost, in terms of error, associated with the forecast of the ML mathematical model used. That is, it maps an event of one or more variables to an actual quantifiable value that represents a cost associated with the event. The optimization of the cost function is generally known as Gradient Descent, which adjusts the weights of the ML mathematical model according to the error they cause in the prediction of the output, in order to minimize the cost function and consequently improve the prediction of the function obtained, decreasing the prediction error of the ML algorithm. Supervised learning algorithms are used in classification tasks, which choose between two or more categories or responses restricted to a limited set of values, and in regression, which predict continuous values and the output may have any numerical value.

In unsupervised learning, data has no labels, that is, inputs have no defined outputs. This form of learning looks for connections in the data and identifies similarities in the data, reacting based on the presence or absence of such similarities in each new piece of data.

This semi-supervised learning approach allows the algorithm to learn based on the data that is labeled as well as to establish connections over the dataset. Being that, it is a mixture of supervised and unsupervised learning.

Reinforcement learning aims to make the algorithm learn based on a set of rules in order to achieve a well-defined goal. This form of learning allows the algorithm to learn when it is performing a positive or beneficial action to reach the end goal and also when it is performing a harmful action that keeps the algorithm away from reaching its end goal. When the algorithm performs a positive action it is rewarded and when it performs a negative action it suffers a punishment. Thus, the algorithm tries to perform positive actions and avoids performing harmful actions. Reinforcement learning algorithms are used in robotics, for example in autonomous vehicles, or to learn how to play a game against a human opponent.

Problem Detection And Model Application

The trilateration algorithms, developed in 2.3.2 and 2.3.1, use the Eq. 2.4, as a way to approximate the values of r_k^2 and r_t^2, presented in equations 2.20 and 2.38, respectively. These values, r_k^2 and r_t^2, represent the distance between the transmitter at zero height and the receiver. However, as in practice the system is always subject to reflections from the transmitted signal, then, estimate the values of r_k^2 and r_t^2 based on Eq. 2.4, which gives the signal power value after it has suffered attenuation in the propagation channel in a LOS configuration, neglecting the reflections, will always produce errors in the calculation of these distance values.

Thus, the ML algorithm, applied to the two mentioned trilateration algorithms, comes to predict the value of the distance r_k^2 and r_t^2 based on the value of strengths of the four signals detected in the receiver, in order to try to eliminate the effect of the reflections associated with the trilateration algorithms.

The ML problem described is a supervised learning regression. Therefore, the Artificial Neural Network (ANN) model was chosen to predict these distance values. ANN, or just Neural Network (NN), is one of the machine learning algorithms with supervised learning.

Also, in this dissertation a NN model is used as a position algorithm. This NN model is used in series with the SPR method in order to estimate the position of the receiver based on the DC gain values of the four signals that are estimated by the SPR method.

Neural Network Model

NN models are modeled and inspired by the human brain, that are capable of learning and recognizing numerical patterns. As in the human brain, NNs are presented as systems of neurons connected to each other that process input values, simulating the behavior of biological NN. Thus, the implemented NN algorithm takes into account several layers of neurons as shown in Figure 2.15. The layer of input neurons, $InputLayer$, has a number of neurons equal to the number of features that characterize the input data plus one. The

output layer of neurons, *OutputLayer*, has a number of neurons equal to the number of classes or types of response that we have at the output. Between these two layers we have the hidden neurons layers, *HiddenLayers*, with a programmable number of layers and a number of neurons per layer. However, the greater the number of neurons in the hidden layers, the greater the computational effort of the implemented NN model.

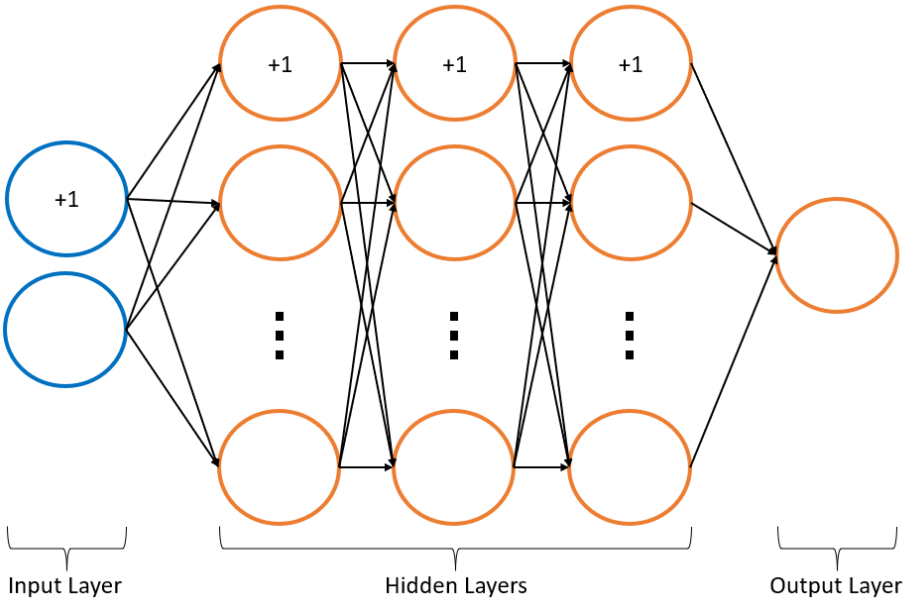

Figure 2.15: Neural networks model illustration.

The cost function, $J(\omega)$, which calculates the cost of the NN regression model, is given by,

$$J(\omega) = \frac{1}{2m}\sum_{i=1}^{m}(h_\omega(x^{(i)}) - y^{(i)})^2 \tag{2.43}$$

Where, m has a value equal to the number of samples in the input data set x, used in training the model, $h_\omega(x^{(i)})$ is the output value estimated by the NN model mathematical function for the input value $x(i)$ and $y^{(i)}$ is the original output value, defined on the training data of the NN model, and associated with the input value $x(i)$ well defined.

An important aspect refers to the uniformity of data values for NN models, because it improves the model training speed and helps to find global minima. The uniformity of the data, x_{norm}, carried out within the scope of this thesis is given by,

$$x_{norm} = \frac{x - u}{s} \tag{2.44}$$

Where, x is the original data set, input or output, to be standardized, u is the average of the data x and s is the standard deviation of the data x.

To obtain the real values again, x, it is enough to perform the denormalization, which is the inverse process of data standardization, x_{norm}, such that,

$$x = x_{norm} \times s + u \tag{2.45}$$

The basic unit of this NN algorithm is the neuron. Activation functions in NN algorithms decide whether the information the neuron is receiving is relevant to the information provided or should be ignored, activating or deactivating the neuron. The activation function chosen for NN models was Rectified Linear Unit (ReLU) which allows the fast convergence of the model. The neuron receives a certain number of inputs that are multiplied by a randomly distributed weight, with the condition that it has a null mean and a standard deviation equal to 1. The sum of all the results passes through the activation function, in order to transform the output, limited and predictable.

This model calculates the error value obtained in the output of all neurons in the hidden layer and calculates an error term that measures how much the neuron was responsible for the error in the model's output. With the sum of the error values obtained, the gradient is calculated and, finally, the weights of the NN mathematical model are updated. This adjustment of the model weights is done by the gradient descent function. That is, gradient descent is an iterative learning algorithm that uses a training dataset to update a model. In the NN model, the gradient descent function has two hyper-parameters, batch and epochs. The batch parameter defines the number of training samples that are computed before the model is updated and takes values between 1 and the size of the database set. The epochs parameter defines the number of complete passes through the training dataset, that is, the NN model analyzes the complete training data according to the value of the epochs parameter and adjusts its weights. This parameter can take integer values between 1 and infinity, and the model can be adjusted infinite times.

The evaluation of the prediction of the NN model is made by the quantitative value of Root Mean Square Error (RMSE) which is the measure that calculates the root mean square of the errors between the m original data values, y_j, and the predictions made by the mathematical function of the model, $\hat{y}_j$, such as,

$$RMSE = \sqrt{\frac{1}{m}\sum_{j=1}^{n}(y_j - \hat{y}_j)} \tag{2.46}$$

Where, m defines the number of samples or output values present from the dataset used in training the NN model.

The smaller the RMSE value, the greater the prediction or response of the NN model in predicting the output values.

CHAPTER 3

Simulation

This Chapter, presents the parameters of the simulated system that is used to simulate the developed algorithms.

In simulation, the photodiode receiver, positioned at a certain point in the room, as explained above, detects the signals that are transmitted by the 4 transmitters, in fixed positions in the ceiling. Also, each transmitter sends the signal with a different frequency and can be reflected by the walls of the simulated indoor environment.

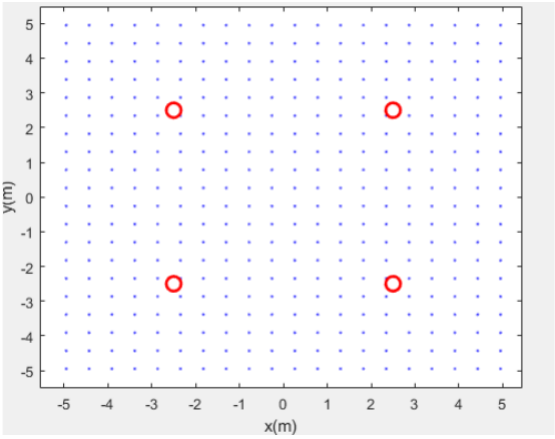

Figure 3.1: Simulated positions, position of the transmitters in red and the possible positions of the receiver in blue.

The Figure 3.1, presents a flat illustration of the positioning of the 4 transmitters in relation to the limits of the room used in the simulations. In this Figure 3.1, in blue are represented the possible locations where the photodiode receiver can be positioned, on the floor of the room, and in red are represented the fixed positions of the transmitters, on the ceiling. In the simulations the transmitters are at the same distance from each other and each transmitter is at the same distance from the closest corner that delimits the room. There is a symmetry in the positioning of the transmitters in this simulation to make it easier to detect non-symmetric errors. Transmitters 1, 2, 3 and 4 are positioned at a height of 3 meters from

the floor of the room, respectively, at the point of spatial coordinates (-2.5,-2.5,3), (2.5,-2.5,3), (-2.5,2.5,3) and (2.5,2.5,3). In the simulations, the transmitters only send signal with unitary power. Also in the simulations, the receiver, with an active area, Ar, of 1×10^{-3} m^2, is only positioned at a height of 0 meters from the floor of the room,taking values $(x, y, 0)$, where $x \in [-5, 5]$ and $y \in [-5, 5]$. Moreover, the order m of the Lambertian radiation pattern is 1. Furthermore, for the parameters that allow the calculation of the signal strength that is reflected by the four walls of the room are such that: the wall reflection coefficient is 0.7, ceiling reflection coefficient is 0.9 and the reflective surface of the wall and ceiling is subdivided into areas of 0.01 m^2.

The block system that describes the flow and correlation of each of the implemented algorithms is shown in Figure 3.2 and the continuation in Figure 3.3.

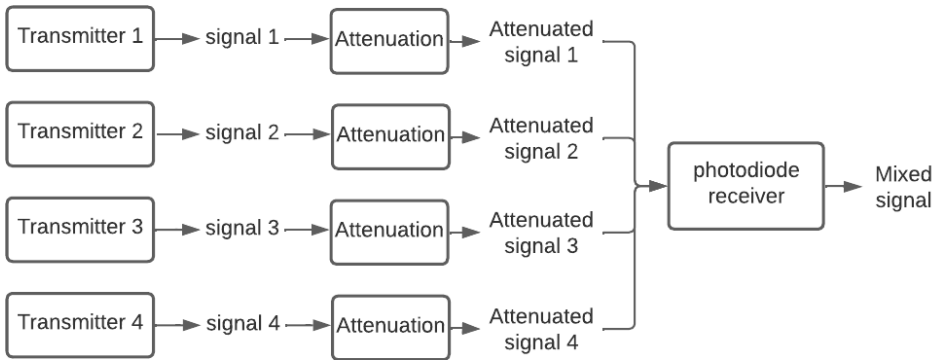

Figure 3.2: Block diagram of the implemented systems.

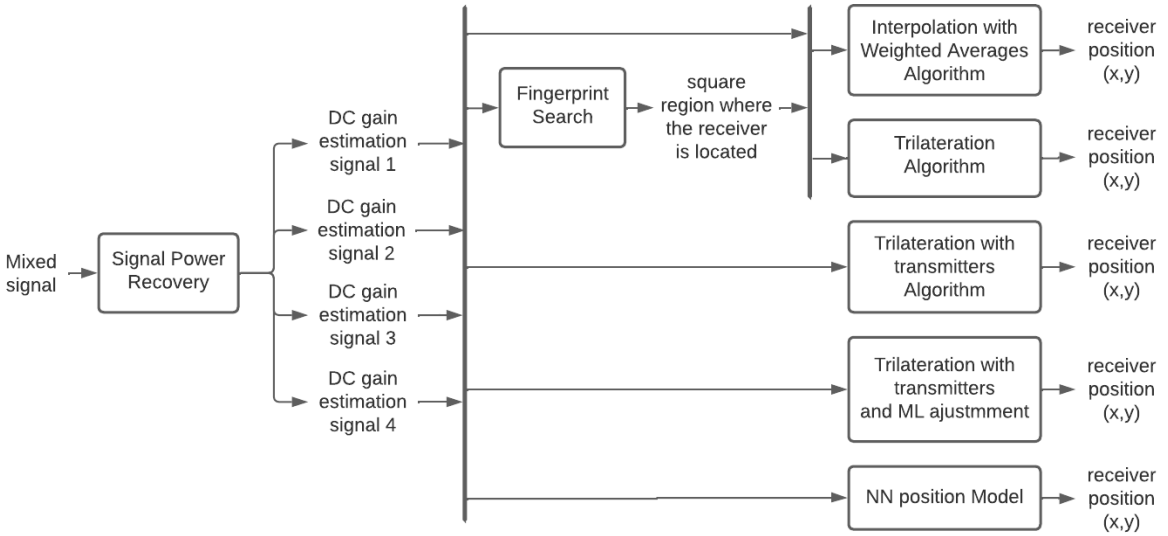

Figure 3.3: Block diagram of the implemented systems (continuation).

3.1 Signal Processing Methods

This Section presents the SPR method which, through the mixed signal that is detected by the photodiode receiver, provides an estimate of the DC gain values of each of the signals that compose the mixed signal.

3.1.1 Signal Power Recovery

As already explained in Section 2.3.1, the amplitude value of each signal i received at the photodiode receiver will be determined based on the relationship between the amplitude value of signal i, which is received at the photodiode receiver, and the corresponding power value in the frequency domain of the signal i, using MATLAB's FFT function. Being f_i the signal frequency i transmitted by light transmitter i, then the signal i in the frequency domain shows the first odd harmonic peak at the frequency f_i and a distance of $2f_i$ between the harmonic peaks. Knowing the position of each peak and based on the power information of each peak, a model is then created that describes the linear evolution of the power value of the peaks in the frequency domain of the signal i and the signal amplitude value i received at the receiver. This model called power model is a linear regression model that receives as input the power value of the peaks in the frequency domain of the signal i and provides an estimate of the signal amplitude value i received at the receiver and is created with a data of $m \times n$ points. Where, $m \times n$ points correspond to the number of points in the room that belong to the dimension Fingerprint map $m \times n$.

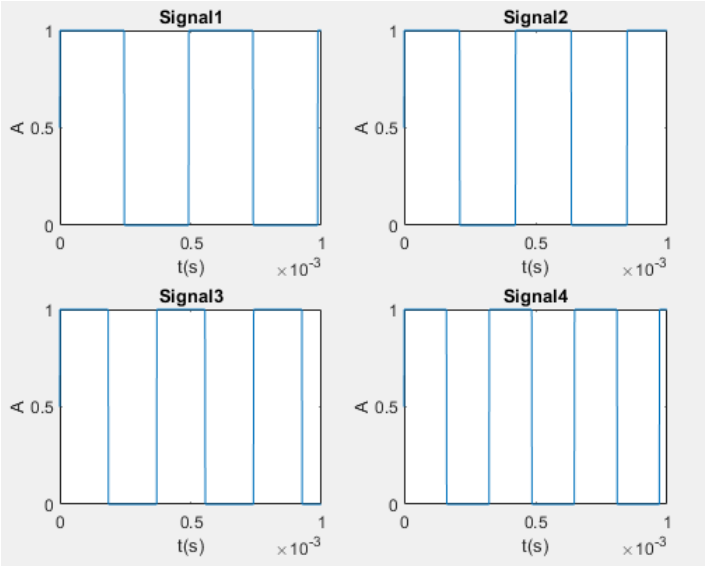

Figure 3.4: Signals transmitted.

In Figure 3.4, are presented the signals, *Signal*1, *Signal*2, *Signal*3 and *Signal*4, quadratic amplitudes, A, transmitted by the transmitters 1, 2, 3 and 4, respectively, with frequencies 2030 Hz, 2360 Hz, 2700 Hz and 3090 Hz, respectively, used in this simulation.

In Figure 3.5, the relationships between the amplitude values of the signal that is received at the photodiode receiver along the different points of the room that belong to the Fingerprint map 10x10 are presented. for each signal 1, 2, 3 and 4, the respectively amplitude is presented by y_1, y_2, y_3 and y_4, and the corresponding power value of the first harmonic of the signal determined by the method described in Section 2.3.1, represented respectively by x_1, x_2, x_3 and x_4.

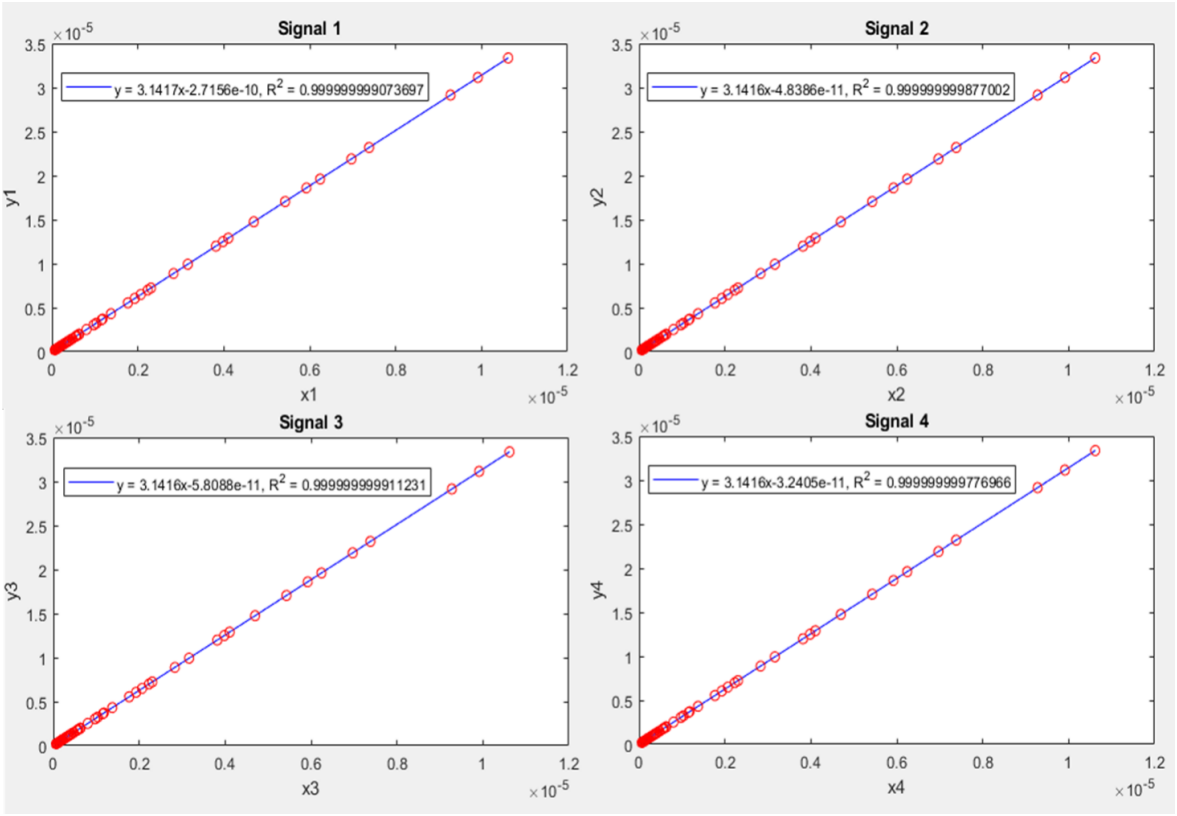

Figure 3.5: Evolution of determined power with received power.

So, based on these graphs in Figure 3.5, create the power models, $model_1$, $model_2$, $model_3$ and $model_4$, which best fit the data used for each signal 1, 2, 3 and 4, respectively, as shown below,

$$
\begin{aligned}
model_1 &: y_1 = 3.1417 \times x_1 - 2.7156 \times 10^{-10}, R_1^2 = 0.999999999073697 \\
model_2 &: y_2 = 3.1416 \times x_2 - 4.8386 \times 10^{-11}, R_2^2 = 0.999999999877002 \\
model_3 &: y_3 = 3.1416 \times x_3 - 3.2405 \times 10^{-11}, R_3^2 = 0.999999999911231 \\
model_4 &: y_4 = 3.1416 \times x_4 - 5.8088 \times 10^{-11}, R_4^2 = 0.999999999776966
\end{aligned}
$$

It should be noted that the relationship between the quantities shown, in Figure 3.5, are not perfectly linear, since R_i^2, $i = \{1, 2, 3, 4\}$ is not of unitary value. R^2 is the coefficient of determination that defines a measure of fit of a generalized linear statistical model, such as simple linear regression, to the observed values of a random variable. R^2 varies between 0 and 1. Expresses the amount of data variance that is explained by the linear regression model. Thus, the higher the value of R^2, the more explanatory is the linear regression model, that is, the better it fits the sample. R^2 is calculated for each of the models according to Eq. 3.1.1, presented below.

$$
R^2 = 1 - \frac{\sum_{j=1}^{m \times n} (y_j - \hat{y}_j)^2}{\sum_{j=1}^{m \times n} (y_j - \bar{y})^2} \tag{3.1}
$$

Where, $m \times n$ is equal to the number of points on the Fingerprint map. For this simulation, the Fingerprint map used has dimensions of 10x10, so it has a data of 100 values. More, y_j represents all signal amplitude values i that are received in the photodiode receiver along the different points of the room that belong to the Fingerprint 10x10 map, such that, $j = \{1, 2, \ldots, 100\}$. The variable $\hat{y}_j$ is the estimated value of y_j and $\bar{y}$ is the average value of all the values of y_j, $j = \{1, 2, \ldots, 100\}$.

In this Figure 3.5, it is observed that as the signal amplitude value i received at the photodiode receiver increases, y_i, the power value determined by the method described in 2.3.1 also increases, x_i. However, this relationship is not perfectly linear.

In the four corners of the room, for the different signals that are received by the photodiode receiver, we have similar amplitude/power values. This happens because the transmitters send signals with the same value of power, of unitary value, and they are placed of symmetrical form in the space of the room. That is, as there is symmetry in the positioning of the system components, then the sum of the 4 power values associated with the signals that are received at the photodiode receiver is the same when the photodiode receiver is placed in each of the four corners of the room. The 4 power values associated with the signals that are received at the receiver, positioned in each of the 4 corners of the room, are the same.

This non-linear models, in the realization of this simulation, appears as an error of the FFT. The FFT error refers to the maximum value of difference between the sum of the powers in the four corners of the room, since, this error is theoretically 0. The error derived from using the FFT function of MATLAB was tested as a function of the number of harmonics used to calculate the power value in the frequency domain of each four signals received by the receiver, as shown in Figure 3.6.

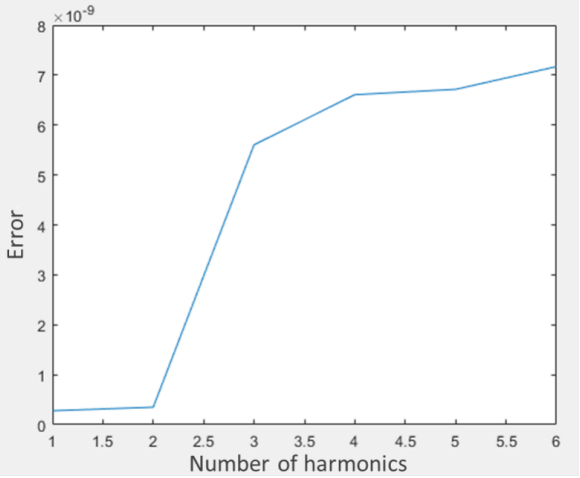

Figure 3.6: Evolution of fast Fourier transform error with number of peaks.

Figure 3.6, shows the maximum error value, the maximum value of difference between the sum of the powers in the four corners of the room, for each harmonics 1, 2, 3, 4, 5 and 6. It can be seen from Figure 3.6 that the error value increases with the number of peaks used in the calculation of the signal power value i at the frequency domain until it stabilizes.

Thus, the power values related to the first harmonic of each signal i are used to create the linear regression models, that best fits the data used, power model.

Furthermore, the error found in the FFT, only for the first harmonic, can be minimized by approximating and adjusting the frequency value used for each signal, as shown below in Figure 3.7 and Table 3.1.

set	f1	f2	f3	f4	error
1	2030	2360	2700	3090	2.7816×10^{-10}
2	2030	2370	2680	2990	8.8879×10^{-09}
3	2030	2390	2760	3130	2.7033×10^{-09}
4	2030	2340	2700	3070	1.4579×10^{-09}
5	400	800	1600	3200	1.5794×10^{-09}

Table 3.1: Frequencies tested.

set	R_1^2	R_2^2	R_3^2	R_4^2
1	0.999999999073697	0.999999999877002	0.999999999911231	0.999999999776966
2	0.999999999698967	0.999998623108083	0.999999999807942	0.999999999854494
3	0.999999974977298	0.999999998670059	0.999999999858658	0.999999928909171
4	0.999999999637438	0.999999999230718	0.999999940751949	0.999999998187383
5	0.999999999971644	0.999999999997796	0.999993853831260	0.999999999755225

Table 3.2: Determination coefficient R_i^2.

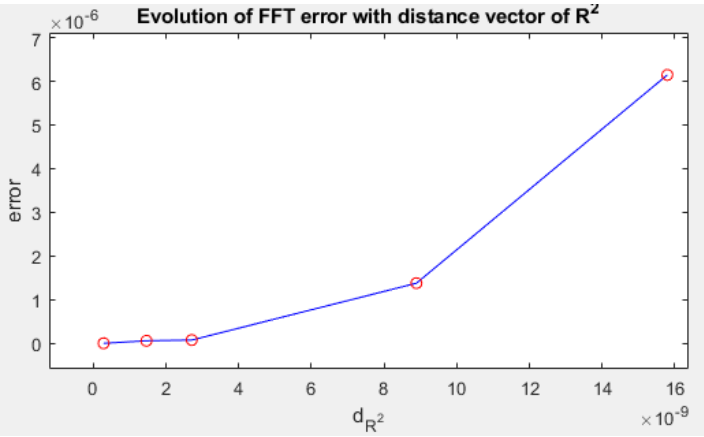

Figure 3.7: Evolution of fast Fourier transform error with distance vector of R^2.

All 5 sets, 1, 2, 3, 4 and 5, of 4 frequencies, f1, f2, f3 and f4 associated with each transmitter 1, 2, 3 and 4, shown in Tables 3.1 and 3.2, are used to test the error that exists in estimating the amplitude value of each of the signals that are received at the receiver. This error evolution is shown in Figure 3.7. All frequency sets presented were chosen iteratively. The variables R_1^2, R_2^2, R_3^2 and R_4^2 are the coefficient of determination of the power models $model_1$, $model_2$, $model_3$ and $model_4$, respectively.

Figure 3.7, shows the evolution between the FFT error, *error*, and the distance, d_{R^2}, of the vector $< R^2 >= (R_1^2, R_2^2, R_3^2, R_4^2)$ with the unit vector, for each set, 1, 2, 3, 4 and 5, calculated using Eq. 3.2, presented below.

$$d_{R^2} = sqrt((1 - R_1^2)^2 + (1 - R_2^2)^2 + (1 - R_3^2)^2 + (1 - R_4^2)^2) \tag{3.2}$$

In Figure 3.7, it is observed that the error of the FFT increases the greater the value of d_{R^2}. This is due to the fact that the higher the value of d_{R^2}, greater is the distance from R_i^2, $i = \{1, 2, 3, 4\}$ with the ideal unit value of coefficient of determination, so the power model is less explanatory, that is, the worse it fits the sample. Therefore, the greater the error of the FFT, the greater the value of d_{R^2}.

In Figure 3.7, it is observed that for set 1 the error value is smaller and that for set 5 the error value is greater. When trying to approximate the numerical value of the frequencies of the four signals, it was possible to reduce the error, however, if the frequencies of the 4 signals are very close, the four signals began to influence each other in the frequency domain, thus increasing the error value in the estimation of the amplitude value of these four signals.

Therefore, frequencies from set 1 are used to continue the simulation of the TSM, described in 2.3.1. Based on these frequencies, the value of the difference between each estimated power value, according to this SPR algorithm, and the respective real power value, determined according to Eq. 2.6, for each room position, as shown in Figures 3.8, 3.9, 3.10,3.11.

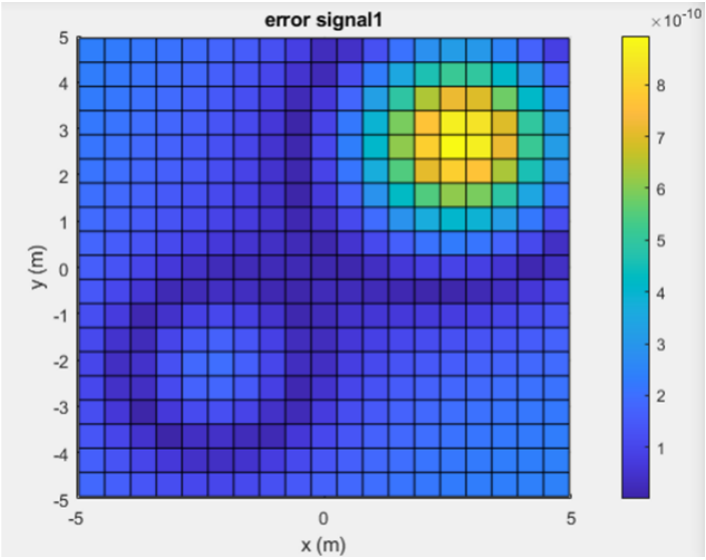

Figure 3.8: Signal power estimation error for signal 1.

For signals 1, 2, 3 and 4, the estimated maximum power error values are 0.89443 μ m, 0.22504 μm, 0.23443 μm and 0.33640 μm, respectively.

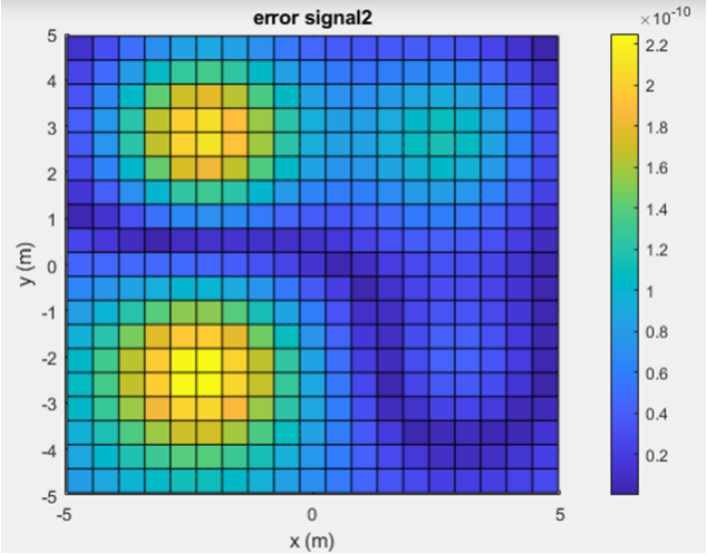

Figure 3.9: Signal power estimation error for signal 2.

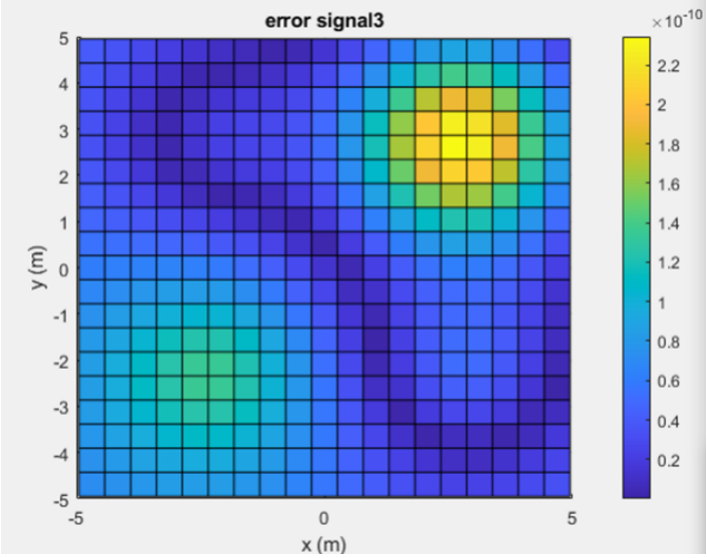

Figure 3.10: Signal power estimation error for signal 3.

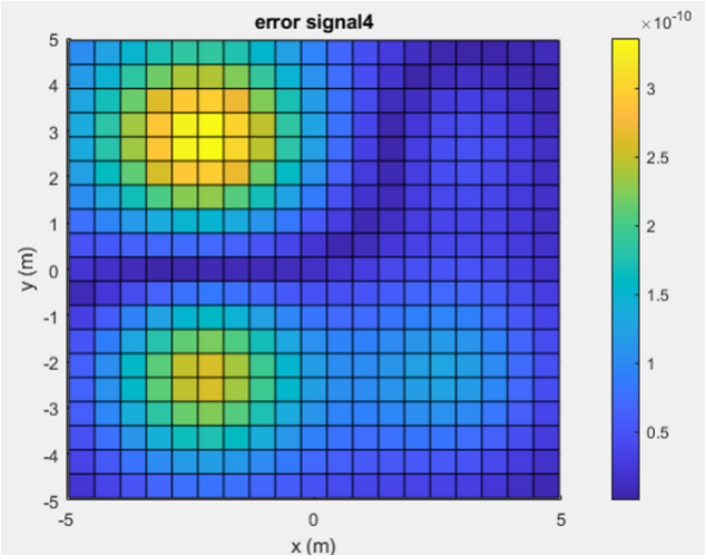

Figure 3.11: Signal power estimation error for signal 4.

3.2 Position Algorithm Methods

In this Section is presented the simulation of the position algorithm methods that estimate the coordinates of the receiver's position on the floor of the room. This position algorithms perform this position estimation using the DC gain information, of each of the signals that compose the mixed signal, given by the SPR method. Here, the methods TSM, Trilateration with transmitters and methods implemented with ML, are presented.

3.2.1 Three Steps Method

At this point, the simulation results of the implementation of the TSM are presented. The results obtained by simulating the Fingerprint Search algorithm, and the interpolation and trilateration algorithms are presented.

Fingerprint Search

In practice, all points belonging to the Fingerprint map correspond to a position $(x, y, 0)$ in the room, so there is no repetition of points, and contain a vector with the four practical values of power measured at each point in the room where the photodiode receiver is placed. Each point on the Fingerprint map contains the information of the four power values determined by the linear model, explained in 3.1.1, of each signal received by the receiver when it is placed in a position $(x, y, 0)$. Thus, in a simulation environment for each of the points, at a position $(x, y, 0)$, belonging to the Fingerprint map, the amplitude values of the signals that are received at the photodiode receiver are first determined, based on Eq. 2.6. Then, linear models are created based on these amplitude values of the received signals and the respective power values of the first peak of each signal in the frequency domain, as explained previously in 3.1.1. So, each point on the Fingerprint map corresponds to a position $(x, y, 0)$ in the room and contains the four power values that represent the estimation of the amplitudes of the signals received at the receiver at that position $(x, y, 0)$, using the linear model created. This is how the Fingerprint map is created.

In this simulation, a room with 400 positions was created, evenly distributed throughout the room, where the receiver is placed and the system tested. Of these 400 positions, it was chosen to use a Fingerprint map with 100 points. As seen in Figure 3.12, the blue dots represent the 400 positions that define the simulation environment of the test area of the room with $10m^2$ and the green dots are the 100 positions that belong to the Fingerprint map.

In a random floor position $(x, y)_T$ in the room, the photodiode receiver is placed and the power of each signal is determined once again using the linear model created. Where the position $(x, y)_T$ represents the "real" position, in simulation, of the photodiode receiver. These four determined power values are defined as the four practical power values, determined in simulation, characteristic of the position $(x, y)_T$ where the receiver is located.

The Fingerprint map will then serve to create the data for training the NN model, $modelD$, as well as a research database in order to find the four points on the Fingerprint map that best delimit the area where the receiver is located. This NN model, $modelD$, estimates the distance, in meters, between two power vectors through the difference of two power vectors.

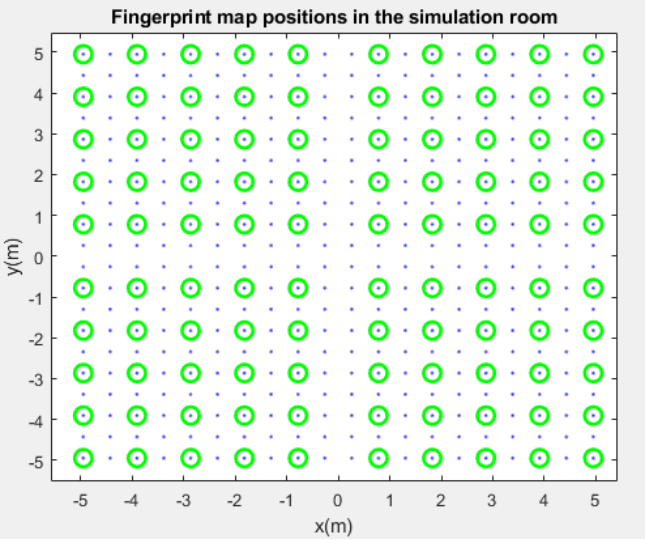

Figure 3.12: Fingerprint map 10x10 positions in the simulation room.

The Fingerprint map is made up of 100 points, that is, 100 power vectors, so the data used for training the NN model, $modelD$, is made up of all combinations of difference between these 100 power vectors. Thus, the data used for training the model has 10000 elements that represent the relationship between the difference and the distance between two power vectors. The floor of the room was defined with 400 possible positions to place the receiver, so the data used for testing the model is constituted by the combinations of the difference between all 400 power vectors, removing only the data used for training. Thus, the model is tested with a data of 150000 elements.

Firstly, the Table 3.3 shows a sample of the training data used, where the 13 features used for training the NN model are represented.

Index	0	1	2	3	4
V_{11} (μJ)	6.4964	6.4964	6.4964	6.4964	6.4964
V_{12} (μJ)	0.5768	0.5768	0.5768	0.5768	0.5768
V_{13} (μJ)	0.5768	0.5768	0.5768	0.5768	0.5768
V_{14} (μJ)	0.1992	0.1992	0.1992	0.1992	0.1992
V_{21} (μJ)	6.4964	9.9351	12.509	11.990	8.8890
V_{22} (μJ)	0.5768	0.9118	1.4942	2.5238	4.3149
V_{23} (μJ)	0.57685	0.6487	0.6863	0.6795	0.6302
V_{24} (μJ)	0.1992	0.2574	0.3295	0.4144	0.5062
dif1 (μJ)	0	-3.4387	-6.0123	-5.4941	-2.3926
dif2 (μJ)	0	-0.3349	-0.9173	-1.9469	-3.7381
dif3 (μJ)	0	-0.0719	-0.1094	-0.1026	-0.0533
dif4 (μJ)	0	-0.0581	-0.1303	-0.2151	-0.3069
d (m)	0	1.0421	2.0842	3.1263	4.1684

Table 3.3: Sample of the training data used for distance neural networks model.

Of the 13 features, each power vector is characterized by 4 features represented as V_{ij}, where $i = \{1, 2\}$ representing one of the two power vectors and $j = \{1, 2, 3, 4\}$ representing

each element of the power vector i. The difference between the vectors is characterized by 4 features, "dif1","dif2","dif3" and "dif4", and 1 features characterizes the distance between the vectors, "d".

Firstly, the relationship between the difference and the distance, d, between each element in the power vectors is presented by the Figure 3.13, for the training data.

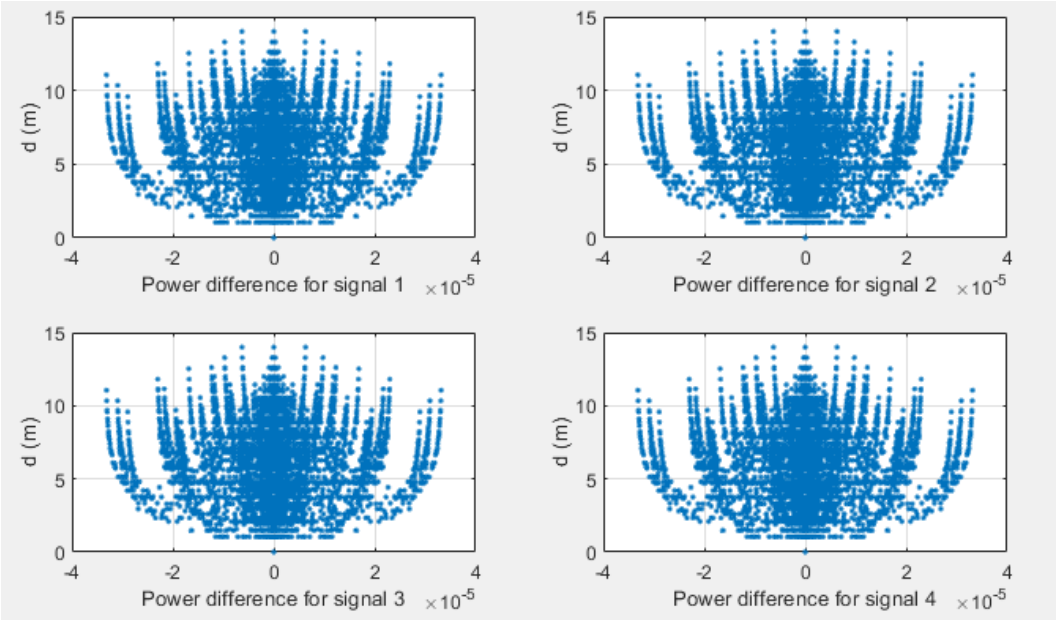

Figure 3.13: Relationship between the difference and the distance between the power vectors.

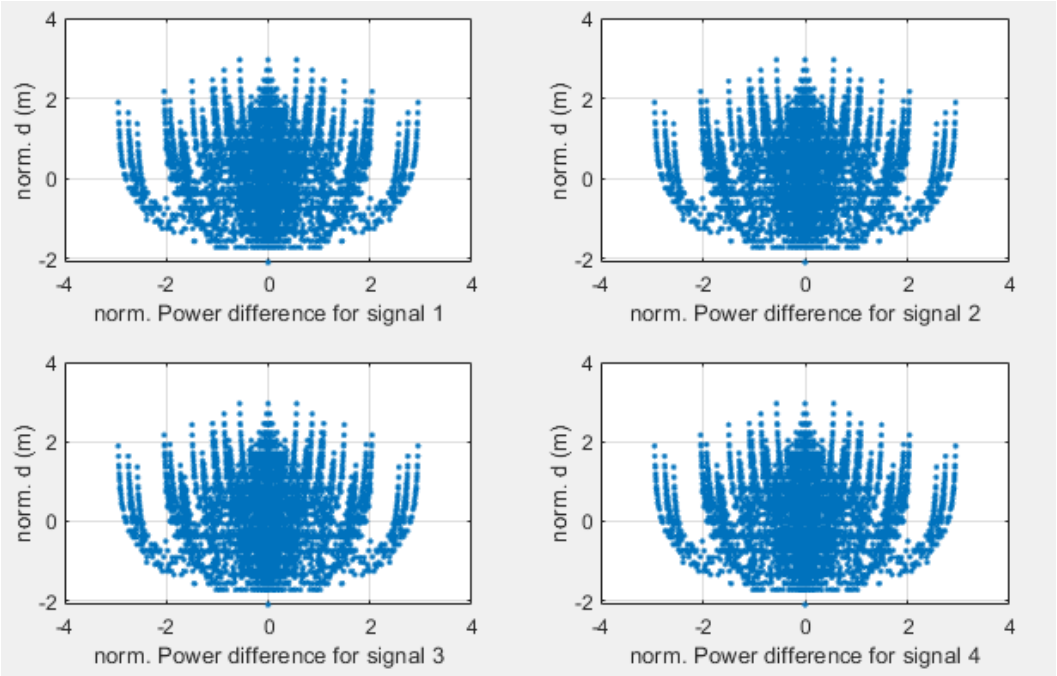

Figure 3.14: Relationship between the standardized difference and the distance between the power vectors.

The values of the presented database are standardized, according to 2.44, before starting

the training of the NN model, as shown in the Figure 3.14.

The implemented NN model has a structure with, 12 neurons in the input layer related to the difference between the power vectors and the values of the power vectors, 1 neuron in the output layer, 3 hidden layers each with 100 neurons. The activation function for the calculations inside each neuron is performed by the ReLU function. The hyper-parameters defined for the training of the *modelD* model were for a batch of size 20 and for 1000 epoch. Therefore, training this *modelD* produces a model with a test RMSE value of 0.075919.

In the Figure 3.15, it can be seen represented in orange, for the test data, the predictions of the value of the distance, d, in meters, related to the difference between two power vectors, made by *modelD*, as well, we have the original test data values represented in blue.

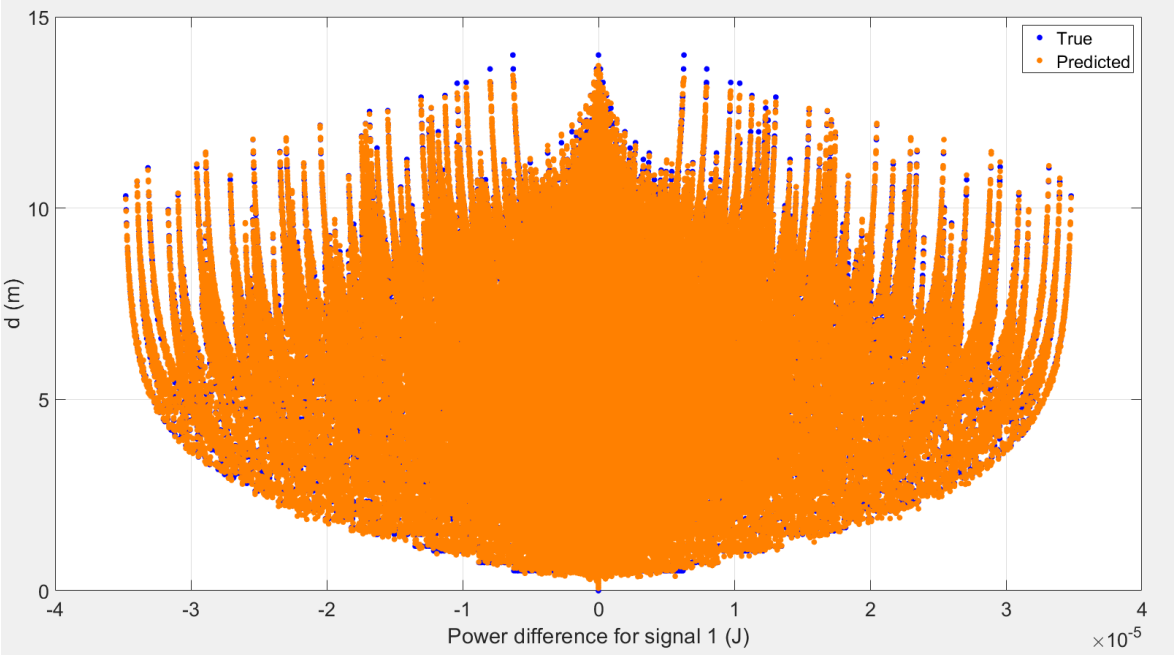

Figure 3.15: Neural network model D predictions for test data.

Now, having trained the model NN, *modelD*, the search algorithm is executed on the Fingerprint map in order to obtain the four points that define the smallest square region where the receiver is located.

It should be noted that the four positions that delimit the area where the receiver is located are only determined if the distance of the vectors mentioned above is not null. When the distance between the practical power vector, characteristic of the position $(x, y)_T$, and the vector of one of the points in the Fingerprint map is null, it means that the position we are estimating is the same as the position where the Fingerprint map point is located. Therefore, the estimated receiver position value takes the value of the position to which the Fingerprint map point is associated, thus ending the simulation for this position where the receiver was placed.

Thus, for the Fingerprint map with 100 points and for each of the 400 possible positions to place the receiver, for each of the positions where the receiver is located, the four points of the Fingerprint map that best fit that position of the receiver based on the explained Fingerprint

Search technique. Merely as a way of testing in simulation the execution of this technique, Fingerprint Search, the estimation of the position of the photodiode receiver corresponds to the position of the first point of the chosen map, that is, the one whose distance between its power vector and the power vector practices is lower. The Figure 3.16 shows the results of the position error in meters of the position estimate provided by the Fingerprint Search technique for all 400 positions where the receiver was placed in a simulation environment.

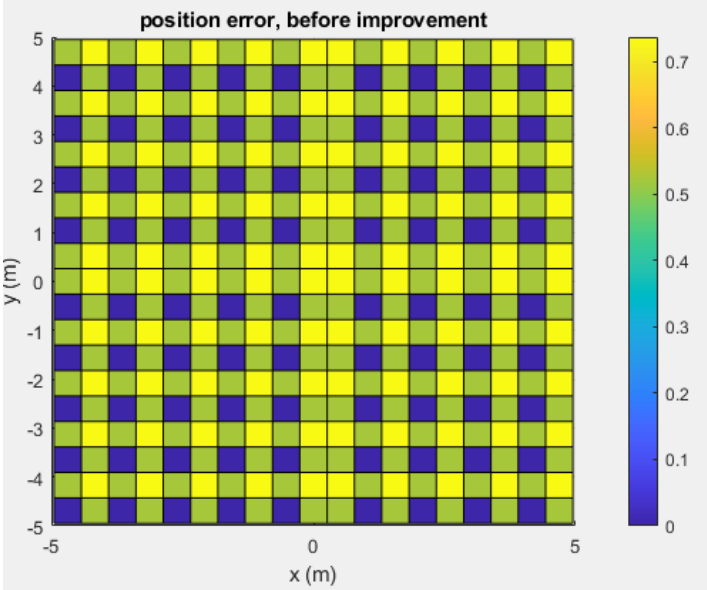

Figure 3.16: Results fingerprint search 10x10.

In Figure 3.16, it is observed that the maximum error obtained with the Fingerprint search algorithm is 0.7443 m. This maximum error value, corresponds to the distance between the receiver's position, in the room, and the first point of the vertice chosen by the Fingerprint search algorithm, within the 4 points provided by the Fingerprint map, which best fit that receiver's position.

In this way, we get an error that follows a pattern across the different positions in the room. It can also be seen that the error of the estimated position value is null when the receiver is placed on one of the positions that are registered in one of the points of the Fingerprint map. This means that the first point chosen by the search algorithm is the point on the Fingerprint map that coincides with the coordinate point where the receiver is positioned. Therefore, the search algorithm correctly chooses the first point on the Fingerprint map that is closest to the receiver.

The Fingerprint Search algorithm is tested again, but now, for a smaller map of points, 5x5, that is, 25 points represented in Figure 3.17. The results obtained are shown in Figure 3.18.

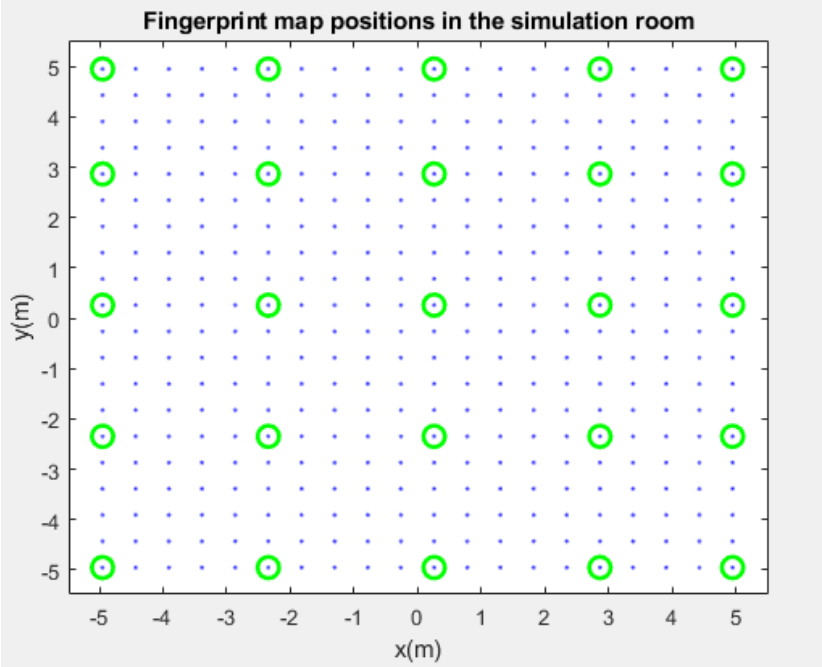

Figure 3.17: Fingerprint map 5x5 positions in the simulation room.

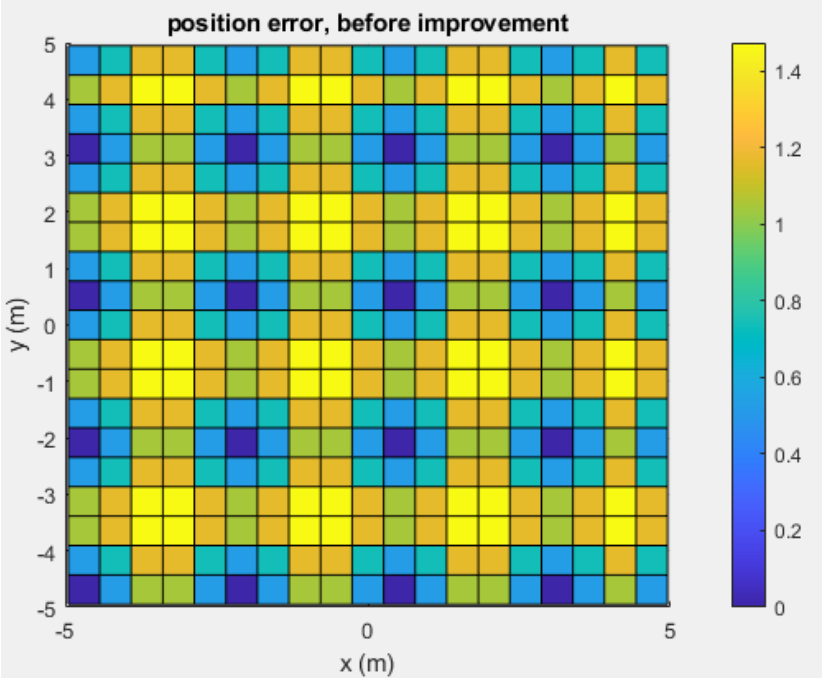

Figure 3.18: Results fingerprint 5x5.

In Figure 3.18, it can be seen that the maximum error obtained with the Fingerprint Search algorithm was approximately 1.488 m.

Positioning Algorithms

In this simulation step we will test the weighted average and trilateration positioning algorithms. The photodiode receiver position is now estimated through these algorithms, thus

discarding the receiver position estimate as the position of the Fingerprint map point whose norm of distance between its power vector and the practical power vector, at the receiver position, it is lower. The Fingerprint search algorithm predates these algorithms.

Interpolation With Weighted Averages

This positioning technique, as explained in 2.3.1, uses the four positions provided by the Fingerprint search algorithm that delimit the region of the room where the receiver is located, and provides an estimate of the coordinate receiver's position value (X, Y) based on Eq. 2.15 and 2.16, whose weight is determined by the cosine function.

Keeping the 100 points on the 10x10 Fingerprint map referred to in Figure 3.12, the interpolation with weighted average algorithm was tested for the 400 positions in the room as shown in Figure 3.19.

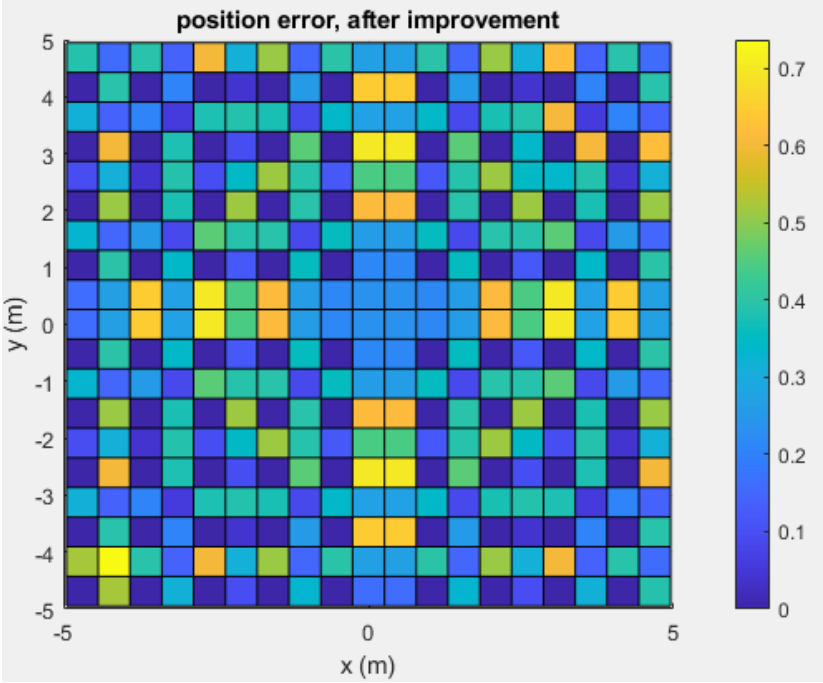

Figure 3.19: Results weighed average with cosine function weight.

As can be seen in Figure 3.19, the weighted average algorithm with weights defined by the cosine function does not produce significant improvements to the system. The receiver position estimation error, of value 0.741 m, decreased for most points in the room, however, the maximum error value of the system has not significantly improved since the results obtained with the Fingerprint map, Figure 3.16.

The Cumulative Distribution Function (CDF) of the position error variable, is a function that explains how values or ranges of values are associated with a probability. That is, it allows the statistical analysis of the position error value at each point in the room. The CDF that describes the probability of position error values for the weighted average interpolation algorithm with weight defined by the cosine function, is presented in Figure 3.20.

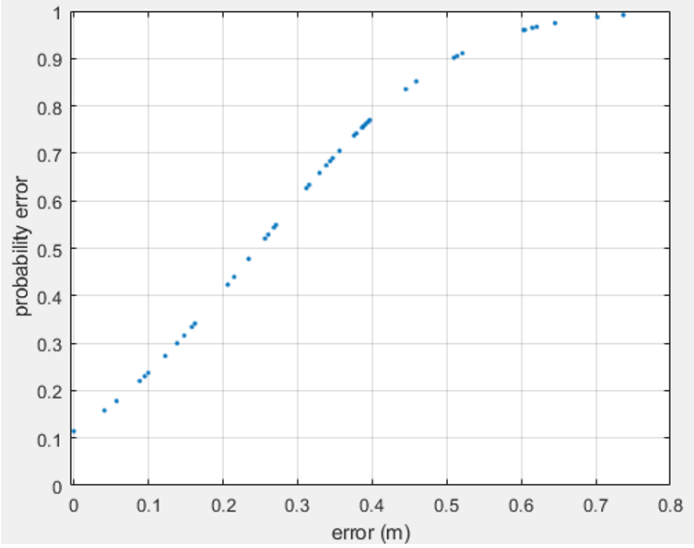

Figure 3.20: Cumulative distribution function for room position error values of weighted average with cosine function as weight.

By the Figure 3.20, the probability that the position error is zero is 11.44%. Note that, for the receiver positions, when it is placed over one of the points in the room that coincides with one of the points on the Fingerprint map, the position error is null. Plus, the probability that the error is less than 0.509 m is 90%.

Now, the estimation of the position value of the coordinate receiver (X, Y) will be determined based on the Eq. 2.18 and 2.19, whose weight is determined by the function exponential. First, we estimated the value of n for which the maximum estimated position error is smaller, for all 400 positions in the room, represented in the Figure 3.12. The Figure 3.21, shows the evolution of the maximum position error value as a function of the value of n. Where it can be seen in this Figure that as the value of n increases, the greater the value of the maximum position error, tending to the value 1.16511 m.

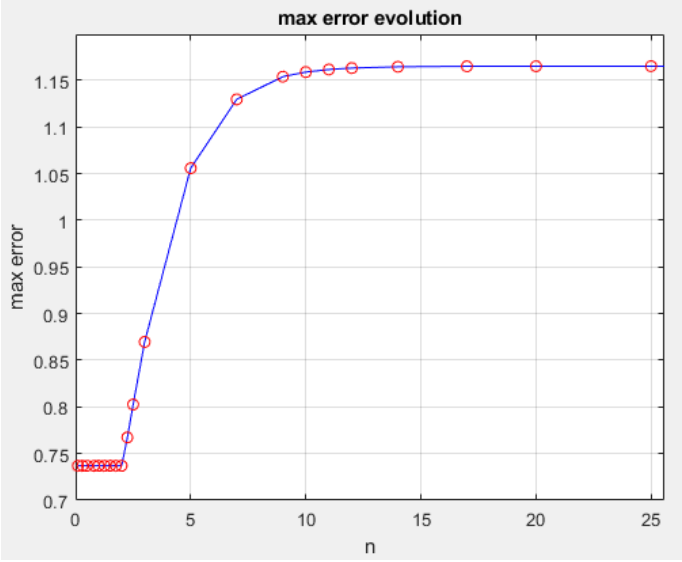

Figure 3.21: Evolution of max position error with n.

Using the CDF, position error values are calculated, with a probability of 95%, depending on the values of n, as shown in the Figure 3.22. From this Figure it is observed that the error value is minimal, 0.5306 m, occurs for $n = 2$.

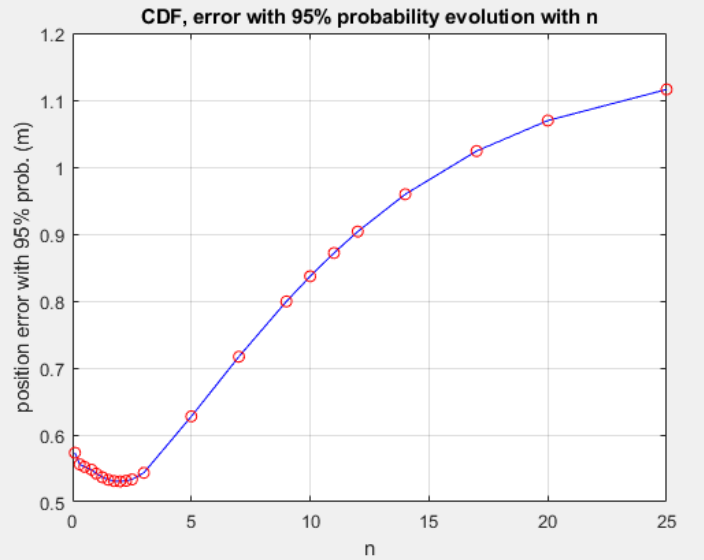

Figure 3.22: Evolution of cumulative distribution function position error with n.

From the Figure 3.22, when we increase the value of n, the greater the range of values that the weight takes, calculated by the exponential $e^{-\frac{d_k}{d_{dmax}} \times n}$, so the error goes down. However, there is a concavity in the graph as the value of n increases. The greater the range of values that the weight takes, the greater the difference between the value of the weights assigned to each coordinate, in the Eq. 2.18 and Eq. 2.19, until, the weight with a higher value becomes predominant in the determination of the coordinates X and Y, by the Eq. 2.18 and Eq. 2.19, with the remaining weights taking values very close to from zero. Therefore, the greater the predominance of a weight associated with a coordinate, which mitigates the influence that other weights have in determining the (X, Y) coordinates of the receiver, the greater the calculated position error.

Keeping the 100 points of the Fingerprint map referred to in Figure 3.12, the interpolation with weighted averages algorithm was tested, using the exponential as weight and $n = 2$, for the 400 room positions as shown in Figure 3.23.

As can be seen in Figure 3.23, the weighted average algorithm, defined by the exponential function, does not produce significant improvements to the system. The receiver position estimation error decreased for most points in the room, however, the maximum error value of the system has not improved since the results obtained with the Fingerprint map, 3.16.

The CDF that describes the probability of the position error values for the weighted average interpolation algorithm with weight defined by the exponential function with $n = 2$, is presented in the Figure 3.24. In this Figure 3.24, the probability that the position error is zero is 11.44%. Note that, for the positions of the receiver, when it is placed on one of the points in the room that coincides with one of the points on the Fingerprint map, the position error is null. Furthermore, the probability value of the error being less than 0.48 m is 90%.

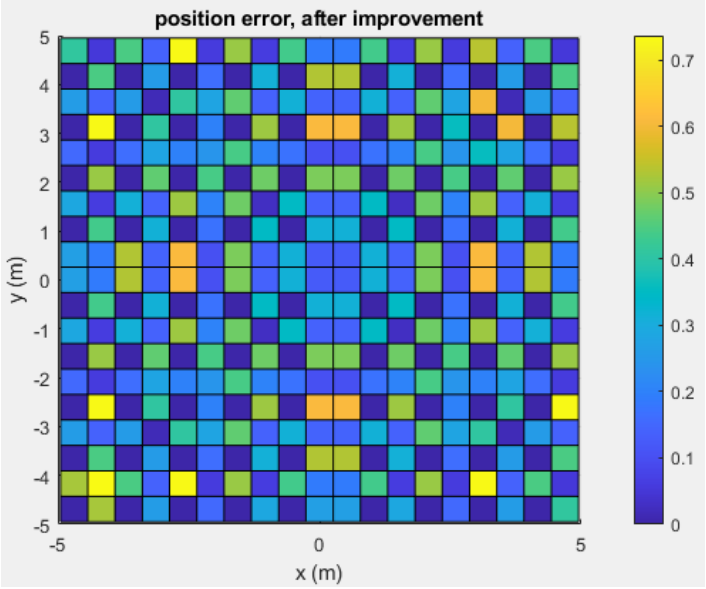

Figure 3.23: Results weighted averages with exponential function weight.

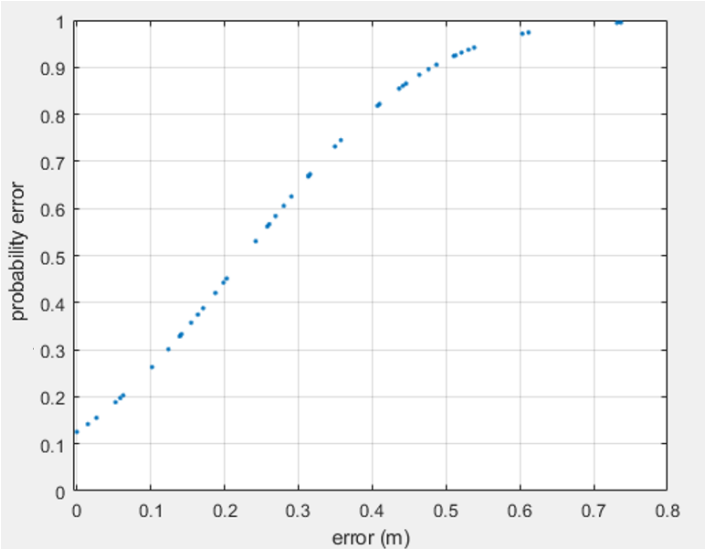

Figure 3.24: Cumulative distribution function for room position error values of weighted average with exponential function as weight.

Trilateration

Now, instead of using the interpolation with weighted average algorithm mentioned above, is used the trilateration algorithm mentioned in Section 2.3.1. This positioning algorithm uses the four positions provided by the Fingerprint search algorithm that delimit the region of the room where the receiver is located, and provides an estimate of the receiver's position value based on Eq. 2.36.

This simulation was divided into two simulations, each corresponding to different manipulations of Eq. 2.27.

In the 1^{st} simulation, the estimation of the positions $Q(X, Y)$ according to Eq. 2.36 whose manipulation to Eq. 2.27 was the subtraction by Eq. 2.27 with $k = 4$. Therefore, matrix A

has dimensions 3x2 and matrix B has dimensions 3x1.

In the 2^{nd} simulation, four manipulations are made to the Eq. 2.27, $i = k$. The Eq. 2.27 was subtracted by the Eq. 2.27 for the parameters $k = \{1, 2, 3, 4\}$, respectively. Hence, matrix A is 4 times larger having dimensions 12x2 and matrix B has dimensions 12x1.

Keeping the 100 points on the Fingerprint map referred to in the Figure 3.12, the trilateration algorithm was tested, from the 1^{st} and 2^{nd} simulations, for the 400 positions of the room, as shown in Figures 3.25 and 3.26, respectively.

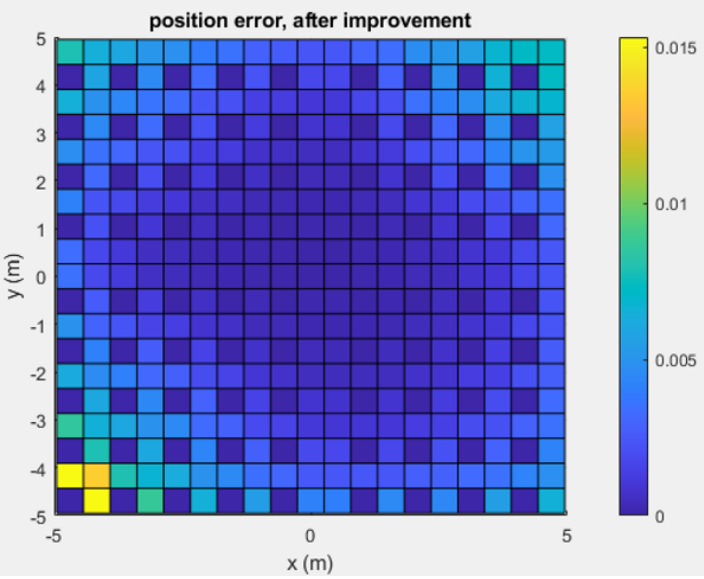

Figure 3.25: Trilateration results, first simulation.

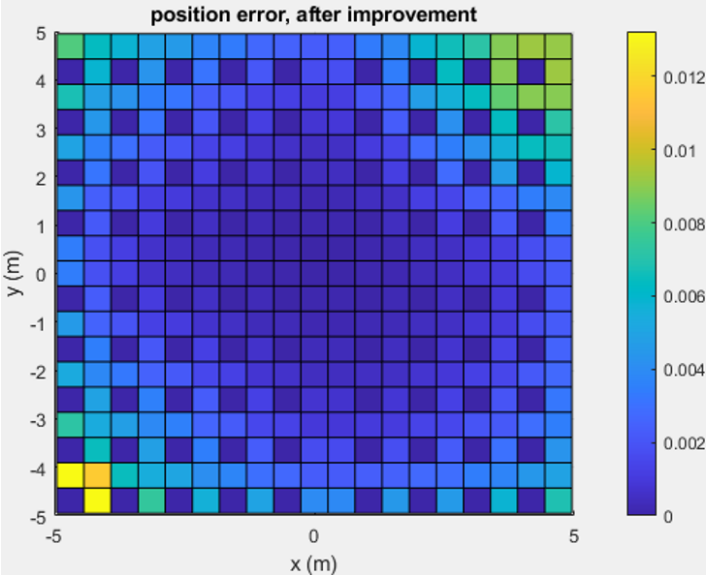

Figure 3.26: Trilateration results, second simulation.

It is observed that, with the increase in the number of equations that characterize the position of the receiver in the room, the maximum position error obtained decreases, being in

the 2^{nd} simulation, of value 1.26 cm, as seen in Figure 3.26. In Figure 3.25, you can see the maximum error for the 1^{st} simulation with a value of 1.5 cm.

For the 1^{st} and 2^{nd} simulations, the CDFs that describe the probability of the position error values are shown in Figures 3.27 and 3.28, respectively. For the 1^{st} and 2^{nd} simulations, 20% of the values have zero position error, since they are the error values for the positions of the receiver, when it is placed on one of the points in the room that coincides with one of the points on the Fingerprint map. However, for the 1^{st} and 2^{nd} simulations 90% of the values have less than 5.29 mm and5.41 mm of position error, respectively.

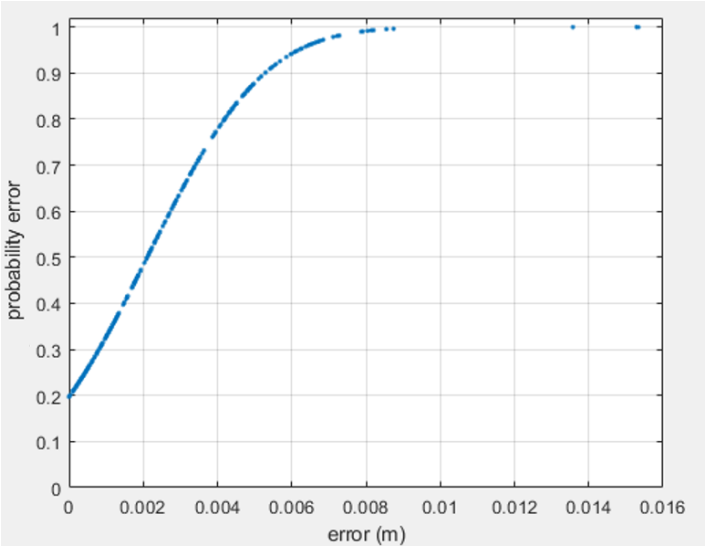

Figure 3.27: Cumulative distribution function for room position error values of trilateration, first simulation.

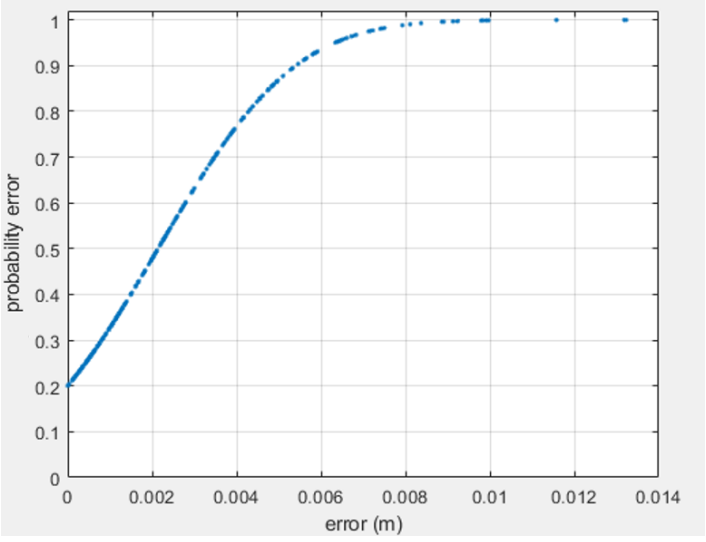

Figure 3.28: Cumulative distribution function for room position error values of trilateration, second simulation.

The trilateration algorithm, developed in 2.3.1, uses the Eq. 2.4, as a way to approximate the values of $(r^i)^2$ and $(r_k^i)^2$, presented in the Eq. 2.21. Referring that $(r^i)^2$ and $(r_k^i)^2$

are variables that represent the distance between the transmitter i, at height 0, and the receiver, and the distance between the point k of the Fingerprint map and the receiver, respectively. However, this approximate distance with the Eq. 2.4, describes the attenuation in the power value of the signal transmitted by the transmitter i in a LOS configuration. That is, disregarding reflections in the indoor environment. As this trilateration algorithm is tested in practice, in a simulation environment, where the power value of the signals detected by the receiver are influenced by replicas of the transmitted signals, NLOS, then, the trilateration algorithm developed, calculates only approximations of the values of $(r^i)^2$ and $(r_k^i)^2$, having an associated calculation error. Therefore, as there is this error associated with the value r_k^2 then this error will propagate in the algorithm leading to the receiver position error verified in the Figure 3.26.

Now, the system is tested when subject to a Gaussian error. Added a Gaussian error to the power value of the signals arriving at the receiver. The photodiode receiver now provides a mixed signal with a Gaussian error added to each of the signals that compose it and the sum of this noise is represented by the Signal to Noise Ratio (SNR) that affects the system. Since a sample of the signal i is sent by the transmitter i, the power value Pr_i of the sample of the signal i that arrives at the receiver is such that,

$$Pr_i = Pr_i + Pr_i \times 10^{(-SNR/10)} \times randn(1) \tag{3.3}$$

Where, $Pr_i \times 10^{(-SNR/10)}$ is the noise power that when multiplied by $randn(1)$, which returns a random scalar extracted from the standard normal distribution, we obtain the added Gaussian noise power to the i signal.

For each value of i, with $i = \{1, 2, 3.4\}$ representing each of the four transmitters, each signal detected by the receiver has a SNR of 5 dB, 10dB, 20 dB, 40 dB, 60 dB and 80 dB and the receiver position is estimated, based on the Trilateration algorithm, now affected by a Gaussian error, when the receiver is placed in all 400 points of the room already shown in blue in the Figure 3.12. Using the CDF, the position error values are calculated, with a probability of 95%, depending on the SNR values, so that, for the referred SNR values, the error values obtained are, respectively, 11.56 cm, 5.371 cm 8.001 mm, 5.23 mm, 5.23 mm, 5.23 mm. These values can be seen in the Figure 3.34.

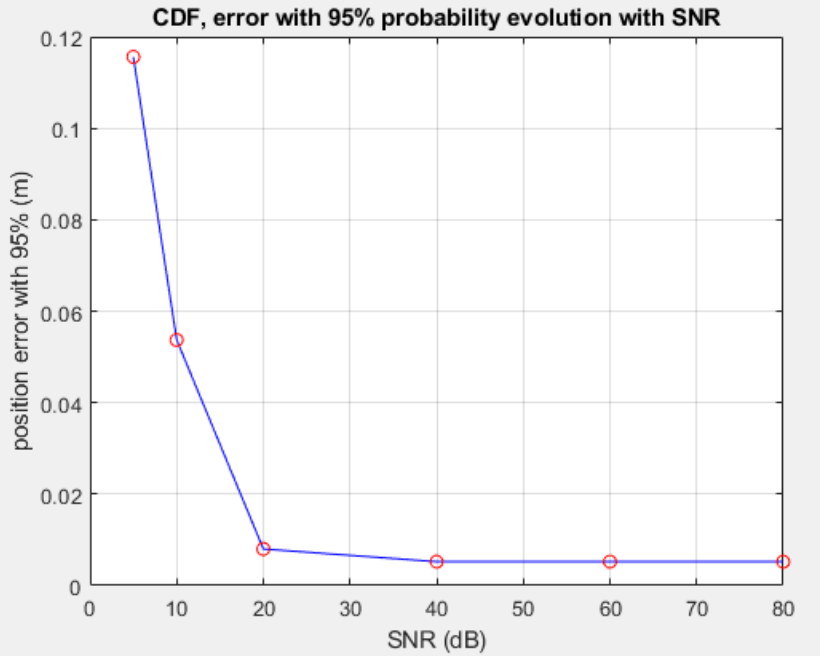

Figure 3.29: Results of trilateration when gaussian error is added to the system.

3.2.2 Trilateration With Transmitters

Now, the trilateration algorithm with transmitters referred to in 2.3.2 is used. This positioning algorithm uses the four positions of the transmitters and the estimated power value, using the SPR technique, at the position $Q(X, Y)$ and provides an estimate of the receiver position value. This simulation was divided into two simulations, each corresponding to different manipulations of Eq. 2.20.

In the 1^{st} simulation, the manipulation performed on the Eq. 2.20 is the subtraction by the Eq. 2.20 for $t = 4$. Therefore, matrix A has dimensions 3x2 and matrix B has dimensions 3x1 . In the 2^{nd} simulation, the manipulation performed on the Eq. 2.20 is the subtraction by the Eq. 2.20 with $t = \{1, 2, 3, 4\}$. Therefore, the matrix A has dimensions 12x2 and matrix B has dimensions 12x1.

Keeping the 100 points on the Fingerprint map referred to in the Figure 3.12, the trilateration algorithm with transmitters, from the 1^{st} and 2^{nd} simulations, was tested for the 400 positions of the room, and the results are shown in Figures 3.30 and 3.31, respectively.

It is observed that with the increase in the number of equations that characterize the position of the receiver in the room, the maximum error obtained was in the 2^{nd} simulation of approximate value 7.31 mm, as seen in Figure 3.31. In Figure 3.30, it can be seen the maximum error for the 1^{st} simulation of approximate value 8.44 mm.

The CDFs that describe the probability of the position error values, for the test performed for the 1^{st} and 2^{nd} simulations, are shown in Figures 3.32 and 3.33, respectively. For the 1^{st} and 2^{nd} simulations 15% of the values have less than 0.24 mm of position error. However, for the 1^{st} and 2^{nd} simulations 90% of the values have less than 4.08 mm and 4.11 mm position error, respectively.

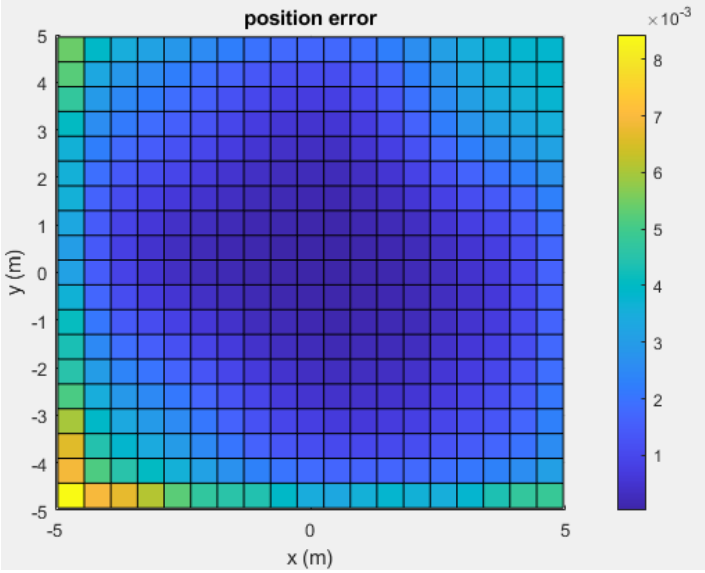

Figure 3.30: Results of trilateration with transmitters, first simulation.

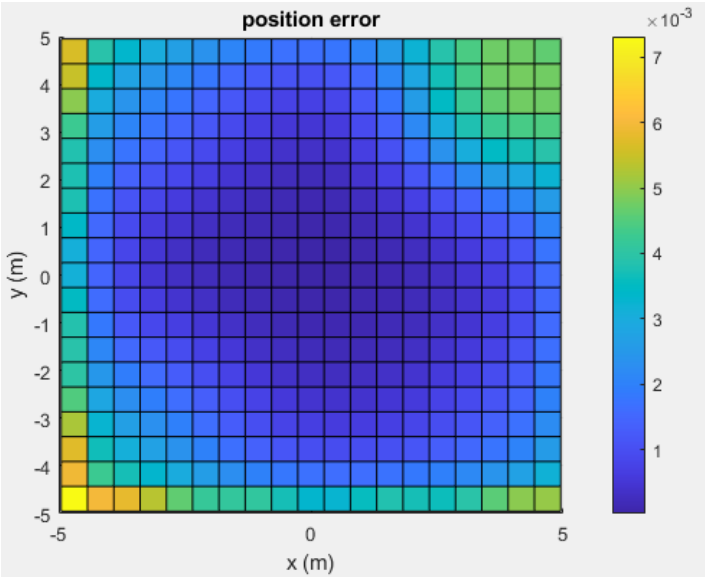

Figure 3.31: Results of trilateration with transmitters, second simulation.

The trilateration algorithm, developed 2.3.2, uses the Eq. 2.4, as a way to approximate the value of r_t^2, presented in the Eq. 2.38. r_t^2 represents the distance between the transmitter i, at height 0, and the receiver. However, this approximate distance with the Eq. 2.4, describes the attenuation in the power value of the signal transmitted by the transmitter i in a LOS configuration. That is, disregarding reflections in the indoor environment. As this trilateration algorithm is tested in practice, in a simulation environment, where the power value of the signals detected by the receiver are influenced by replicas of the transmitted signals, NLOS, then, the developed trilateration algorithm calculates the position (X, Y) through approximations of the values of r_t^2 thus having an associated calculation error. Therefore, as there is this error associated with the value r_t^2 then this error will propagate in the algorithm leading to the receiver position error verified in the Figure 3.31.

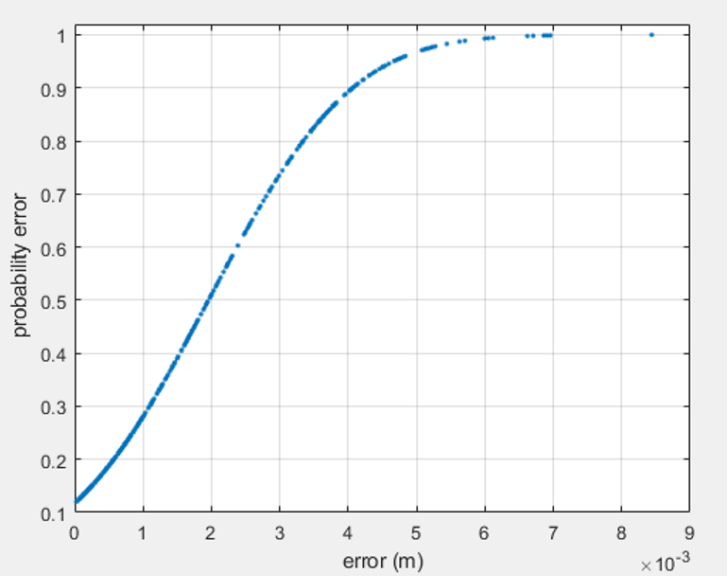

Figure 3.32: Room position error probability with cumulative distribution function of trilateration with transmitters, first simulation.

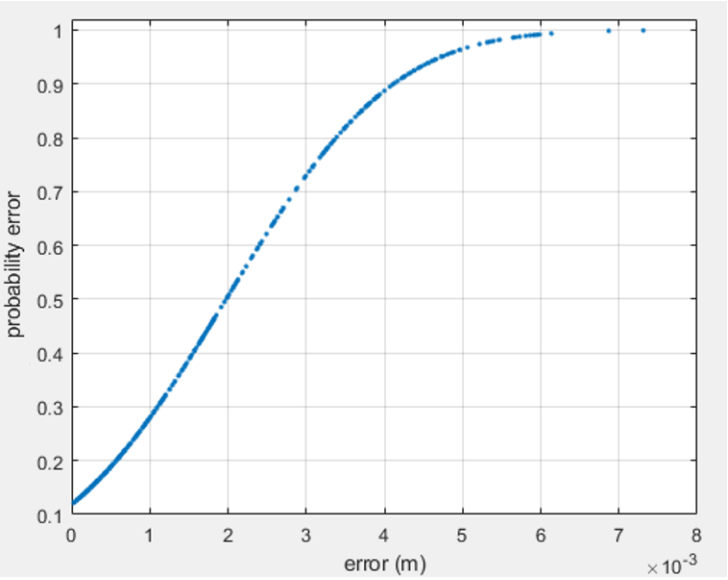

Figure 3.33: Room position error probability with cumulative distribution function of trilateration with transmitters algorithm, second simulation.

As previously performed, this system will now be evaluated on the Gaussian noise effect. For each value of i, with $i = \{1, 2, 3.4\}$ representing each of the four transmitters, each signal detected by the receiver has a SNR of 5 dB, 10 dB, 20 dB, 40 dB, 60 dB and 80 dB and the receiver position is estimated, based on the Trilateration algorithm with transmitters, now affected by a Gaussian error, when the receiver is placed in all 400 points of the room already shown in blue in the Figure 3.12. Using the CDF, the position error values are calculated, with a probability of 95%, depending on the SNR values, so that, for the referred SNR values, the error values obtained are, respectively, 7.831 cm, 2.943 cm, 5.498 mm, 4.721 mm, 4.726 mm, 4.726 mm. These values can be seen in the Figure 3.34.

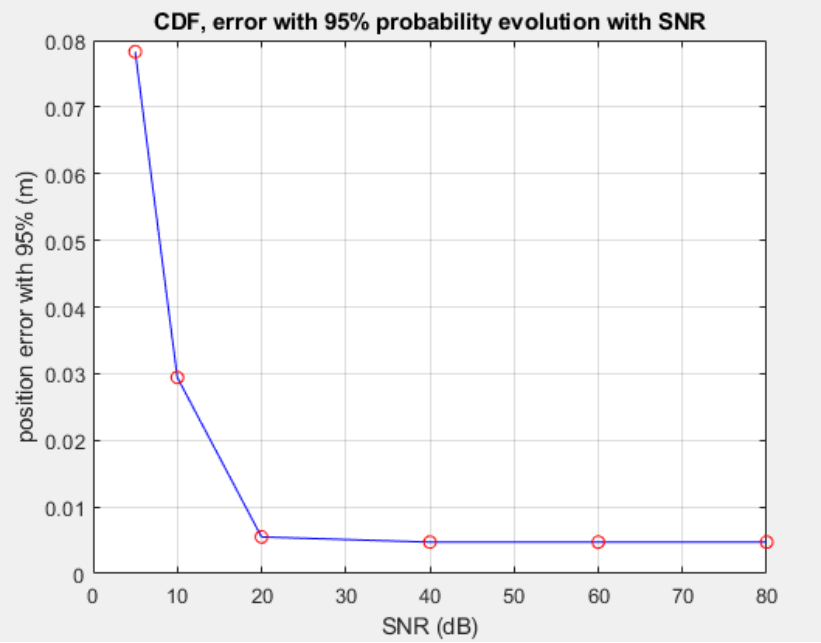

Figure 3.34: Results of trilateration with transmitters when gaussian error is added to the system.

Where, it can be seen that the position error value increases when the SNR value is smaller, presented in dB in the Figure 3.34, since the smaller the SNR the larger is the influence of noise on the signal. Furthermore, the error value tends to a maximum error value when the noise influence on the signal increases and the position error value tends to a minimum value when the noise influence on the signal decreases and becomes negligible.

Figures 3.35 and 3.36, show the results of the trilateration algorithm with transmitters for the 400 positions of the room when the SNR value, is 5 dB and 80 dB, whose value of maximum position error is 1.473×10^{-1} m and 7.312×10^{-3} m, respectively.

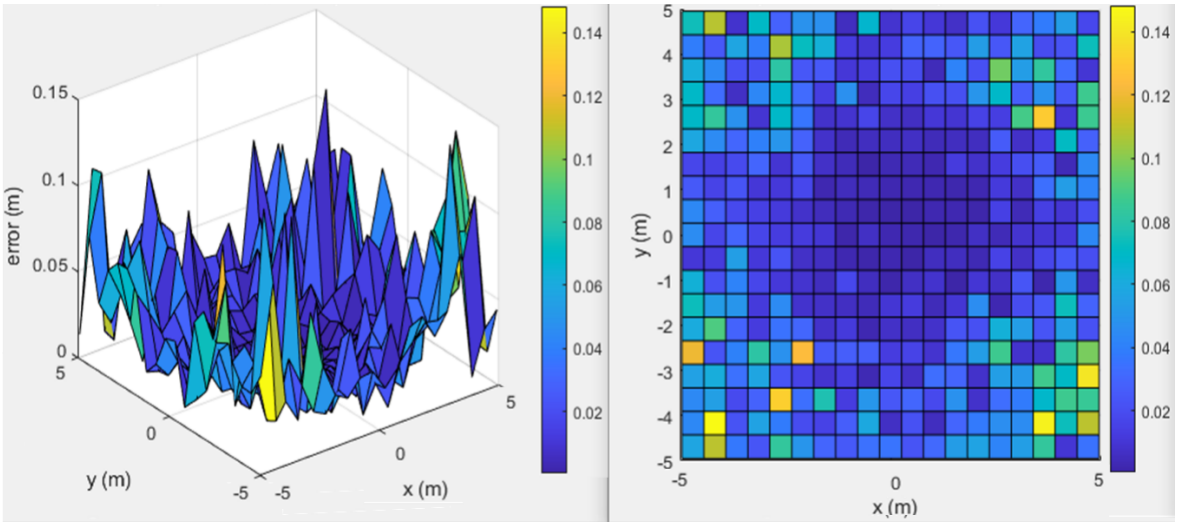

Figure 3.35: Results of trilateration with transmitters with signal to noise ratio 5 dB.

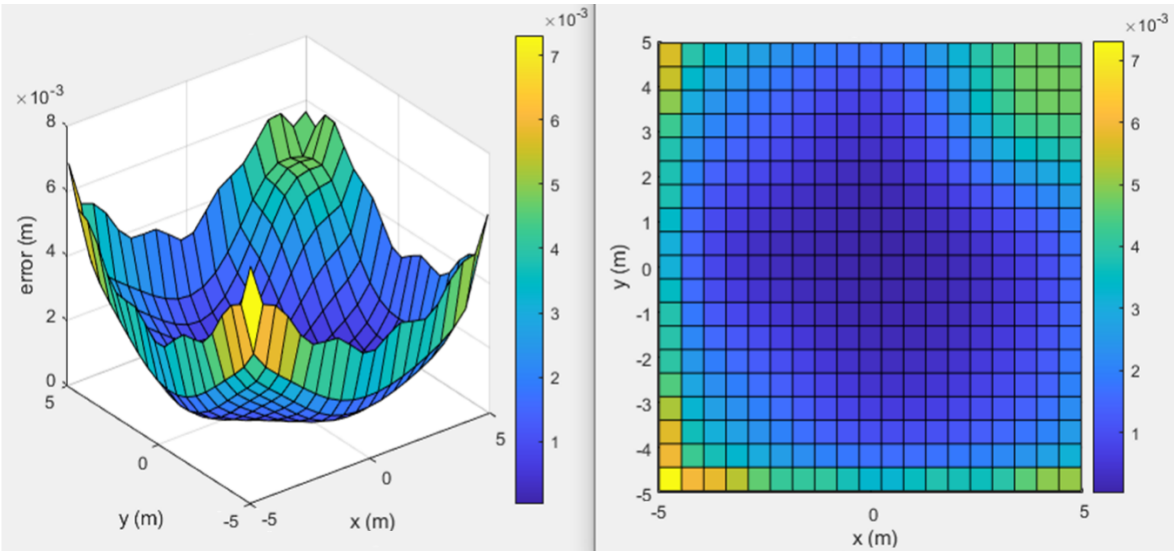

Figure 3.36: Results of trilateration with transmitters with signal to noise ratio 80 dB.

Comparing the values of position error verified in these two Figures 3.35 and 3.36 it can be seen that when the signal is subjected to a smaller SNR value, then the greater is the influence of the noise power added to the system, so the greater is the position error value for each position of the receiver in the room, in a simulation environment.

The CDF that describe the probability of position error values for SNR values of 5 dB and 80 dB are shown in Figures 3.37 and 3.38, respectively. In the Figure 3.37, for an SNR value of 5 dB, it is observable that 90% of the values have less than 6.797 cm of position error and that 15% of the values have less than 1.745 mm position error. In the Figure 3.38, for an SNR value of 80 dB, 90% of the values have less than 4.121 mm of position error and 15% of the values have less than 0.245 mm of position error.

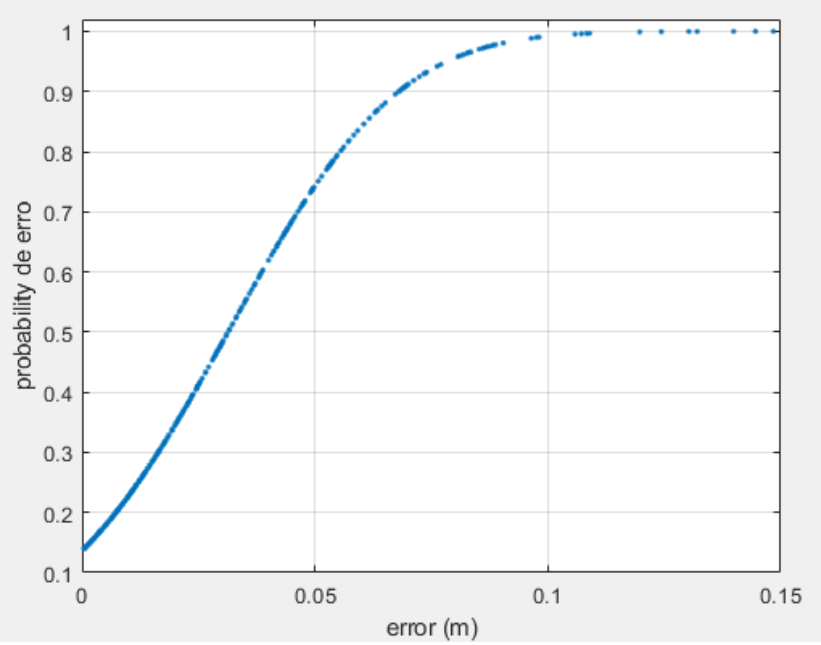

Figure 3.37: Cumulative distribution function for room position error values with an signal to noise ratio value of 5 dB.

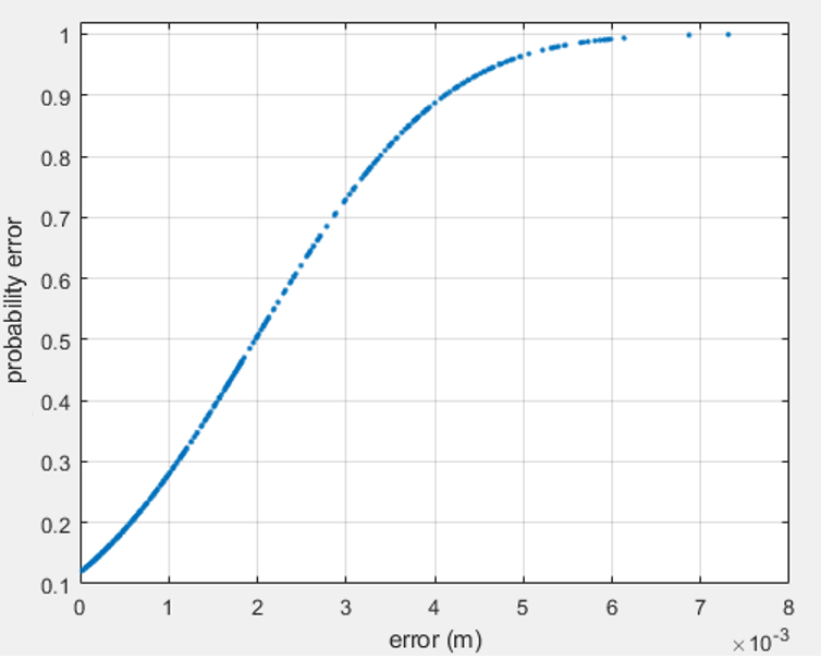

Figure 3.38: Cumulative distribution function for room position error values with an signal to noise ratio value of 80 dB.

3.2.3 Machine Learning

In this Section, are presented the simulation results of two position algorithms. The position algorithm developed and tested in simulation are the Trilateration With Transmitters And Neural Network Model and the Neural Network Position Model.

Trilateration With Transmitters And Neural Network Model

In simulation, in order to eliminate the effect of reflections associated with the developed trilateration algorithm, a NN model is used. The NN model, applied to the aforementioned trilateration algorithm with transmitters, predicts the value of the distance r_t^2, presented in the Eq. 2.38, based on the power values of the four signals detected in the receiver.

In this simulation, the NN model will be used to predict the value of the variable r_t^2 and, thus, estimate the position of the receiver, using the trilateration algorithm based on the values r_t^2 estimated by the NN model. For each receiver position in the room we have t distance values. Therefore, 4 NN models are created, each one, in order to predict the values of r_1^2, r_2^2, r_3^2, r_4^2.

In the Figure 3.39, we have the relations between the value of powers, 1, 2, 3 and 4, of the signals detected in the receiver, for the 400 positions in the room, represented in the Figure 3.1, and the distance values, r_1^2, r_2^2, r_3^2 and r_4^2, respectively.

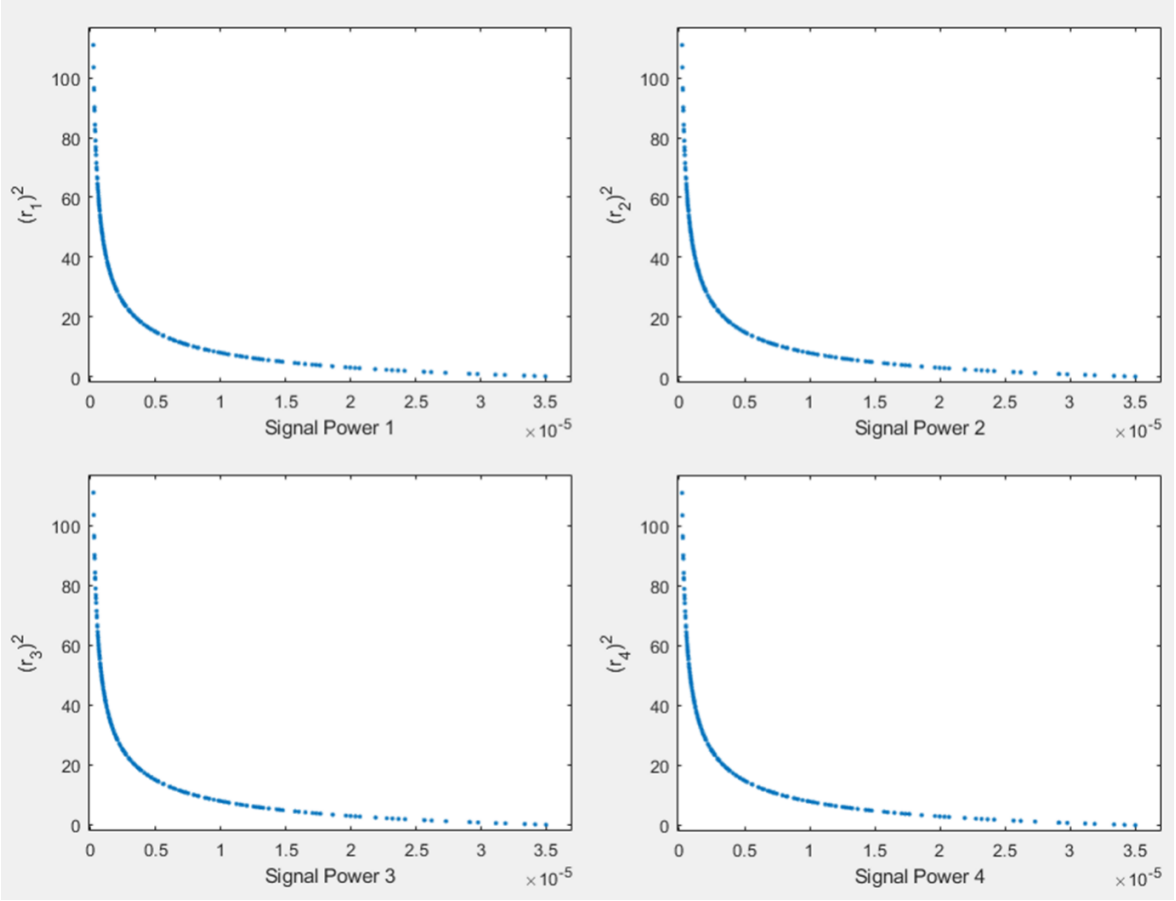

Figure 3.39: Relationship between power and quadratic distance.

The total data being 400 points and the data used for training and validation being the 100 points that will be measured to create the 10x10 Fingerprint map, in practice, represented in green in the Figure 3.12, then the remaining 300 points are used for testing.

The values of the presented database are standardized, according to 2.44, before starting the training of the NN model, as shown in the Figure 3.40.

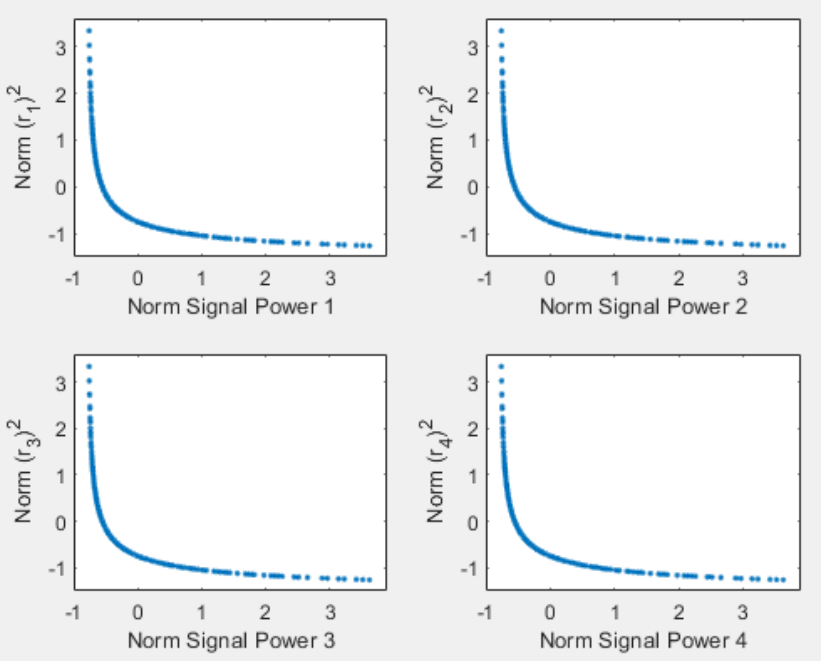

Figure 3.40: Relationship between power and quadratic distance normalized.

First, a NN model was created that predicts the distance value r_1^2 based on the input power value, *Signal Power* 1, of signal 1 received at the receiver. The implemented NN model has 3 hidden layers each consisting of 5 neurons, the activation function for the calculations inside each neuron is the ReLU function and the hyper-parameters of the gradient function, epochs and batch, are respectively 1000 and 20. In the Figures 3.41 and 3.42, we observe the predictions, in orange, of the distance value r_1^2, made by the NN model, as well as the original data values, in blue, for the training and test data, respectively.

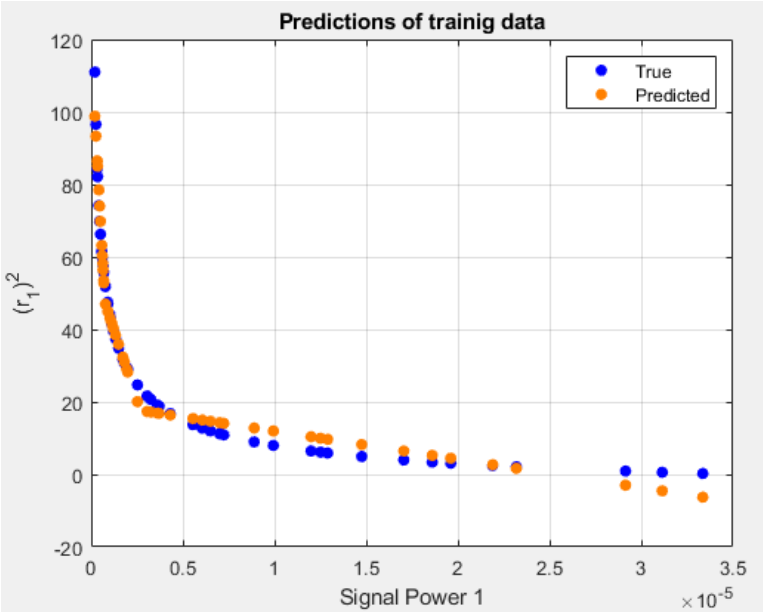

Figure 3.41: Neural networks model predictions for training data.

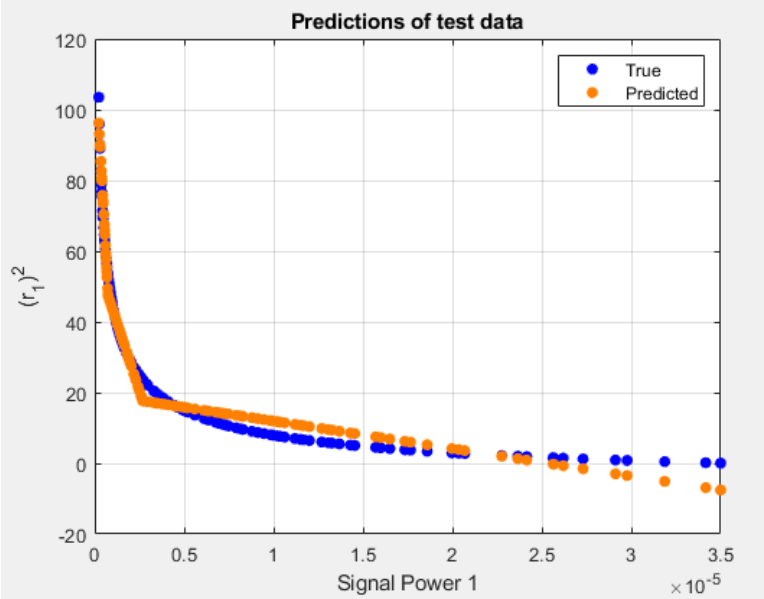

Figure 3.42: Neural networks model predictions for test data.

The forecast values obtained by the NN model and presented in these two Figures, 3.41 and 3.42, underwent denormalization according to the Eq. 2.45 based on the normalization parameters u and s determined in the data normalization before training the NN model.

It can be seen, from the results obtained in the Figures 3.41 and 3.42 that the NN model after training presents a high error in the prediction of the distance value r_1^2, having a value training RMSE value of 2.9791 and a test RMSE value of 3.0583, according to the Eq. 2.46. Since the created model presents a prediction of the distance value with a test RMSE value, 3.0583, then, in order to decrease the model's RMSE value, in order to improve the prediction of the same, the variation of the size of the hidden layers of the model. The number of neurons in each of the three hidden layers was varied for the values 5, 10, 25, 100, 200 and 300. In other words, 216 combinations were tested to structure the NN model.

In the Table 3.4, is shown a sample of the results obtained in the optimization of the NN model, where the test RMSE values are represented, in the column "$RMSE\_test$".

TrialNumber	Parameters	RMSE_test
1	5-5-5	7.6345
2	10-5-5	7.6176
3	25-5-5	2.0840
4	100-5-5	1.7287
5	200-5-5	1.3761
...	...	...
212	10-300-300	1.5116
213	25-300-300	0.8926
214	100-300-300	0.7455
215	200-300-300	0.8035
216	300-300-300	4.7920

Table 3.4: Sample of optimization results.

The RMSE values are associated with the number of neurons in the hidden layers. The first, second and third hidden layer and presented in the column "Parameters", separated by "-", respectively.

The results obtained by optimizing the model parameters allowed us to obtain a model with a test RMSE value of 0.1463 when the sizes of hidden layers 1, 2 and 3 are 10, 25 and 300, respectively. With these parameters, in the Figure 3.43, it is possible to observe the predictions, for the test data, of the distance value r_1^2 made by the model represented in orange, as well as, we have represented the values original dates in blue, the same as those represented in the Figure 3.39.

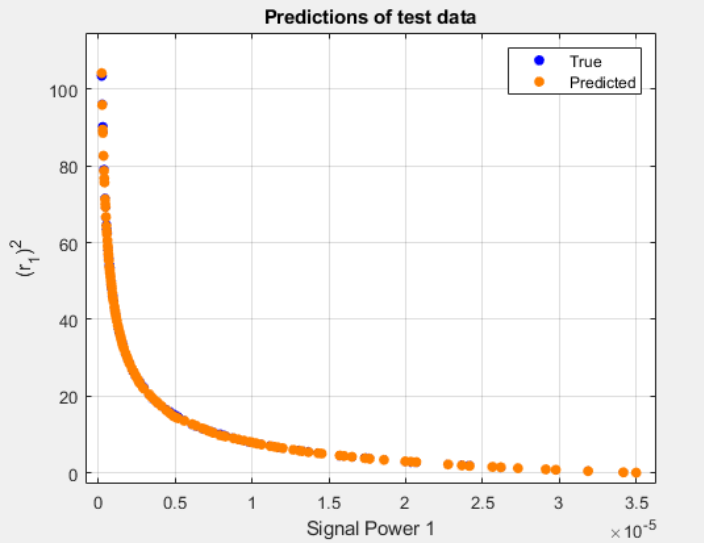

Figure 3.43: Optimized neural network model predictions for test data.

From the results obtained in the Figure 3.43, it can be seen that the predictions made by the optimized model predict the original distance values, r_1^2, with an error lower than the results obtained by the first implemented model, Figure 3.42.

Repeating the process, in order to predict the remaining three distance measures r_2^2, r_3^2 and r_4^2, three more NN models are created. The numbers of the three hidden layers of these three models are, respectively, 10, 25 and 300, equal to the optimized model. In this way, the results of the Figure 3.44 were obtained, where the predicted value of the distances is represented in orange, r_1^2, r_2^2, r_3^2 and r_4^2, of the four models created and the blue dots represent the original distance values of, r_1^2, r_2^2, r_3^2 and r_4^2, associated with each signal strength value, 1, 2, 3 and 4, received at the receiver, respectively, for the test data values. Associated with the prediction of the distance values r_1^2, r_2^2, r_3^2 and r_4^2, the RMSE test values of the models are, respectively, 0.1463, 0.1134, 0.1987 and 0.1120.

Using the four models created to provide a prediction of the distance values between each of the transmitters at zero height and the receiver, r_1^2, r_2^2, r_3^2 and r_4^2, from the estimated power value of each of the signals 1, 2, 3 and 4 received at the receiver, we recalculate the position values (X, Y) of the receiver from the trilateration algorithm with transmitters, described in 2.3.2. Thus, the results presented in the Figure 3.45 were obtained.

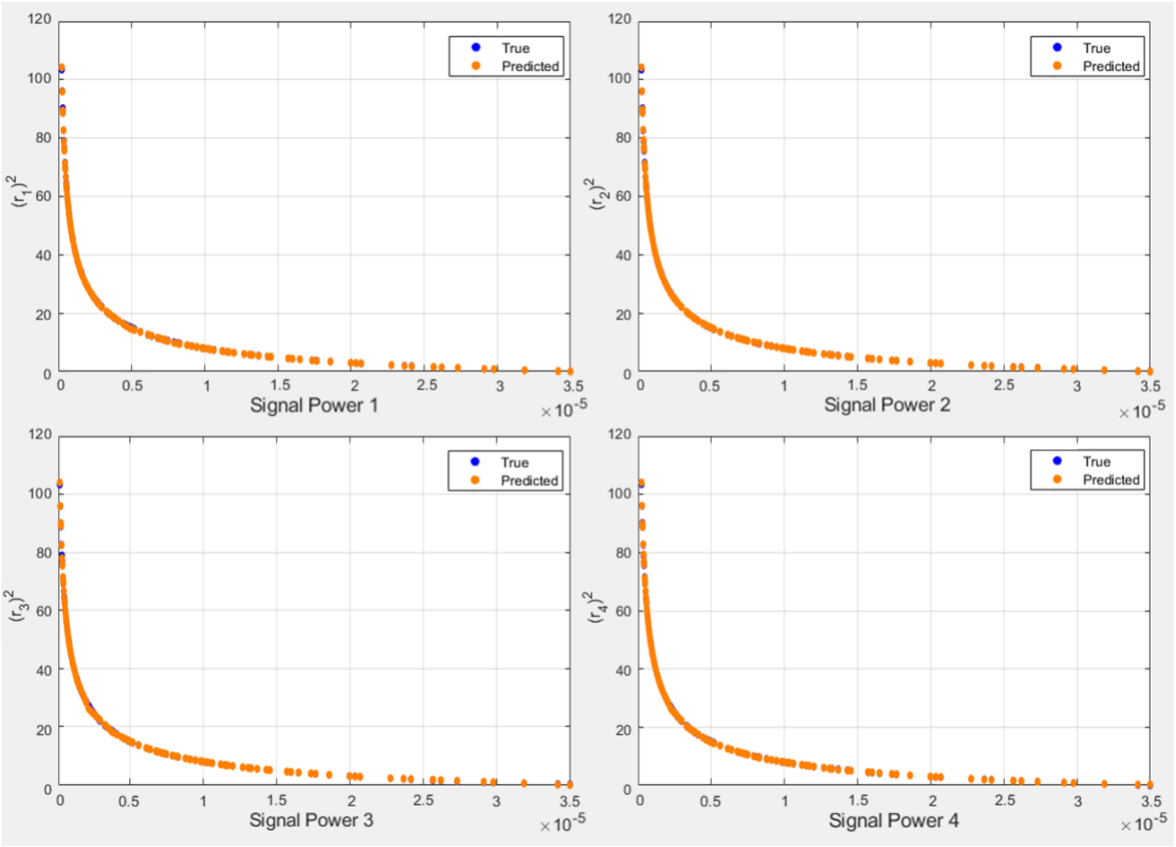

Figure 3.44: Four optimized neural network model predictions for test data.

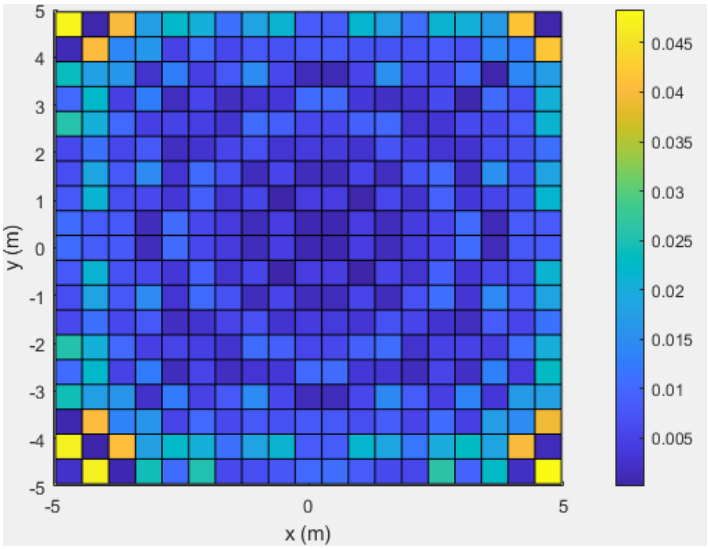

Figure 3.45: Trilateration with transmitters results using neural network models.

In the Figure 3.45, it is observed that the maximum position error obtained is 4.8 cm. The ML model approach where the maximum position error value is 4.8 cm, has a higher error value than the trilateration algorithm approach where the distance values r_1^2, r_2^2, r_3^2 and r_4^2 of the algorithm are estimated based on the Eq. 2.4 and where a maximum position error value of 7.31 mm is obtain.

The CDF describing the probability of position error values is shown in the Figure 3.46. From this Figure 3.46, it is observable that 90% of the values have less than 2.2 cm of position error and that 15% of the values have less than 0.24 mm of position error.

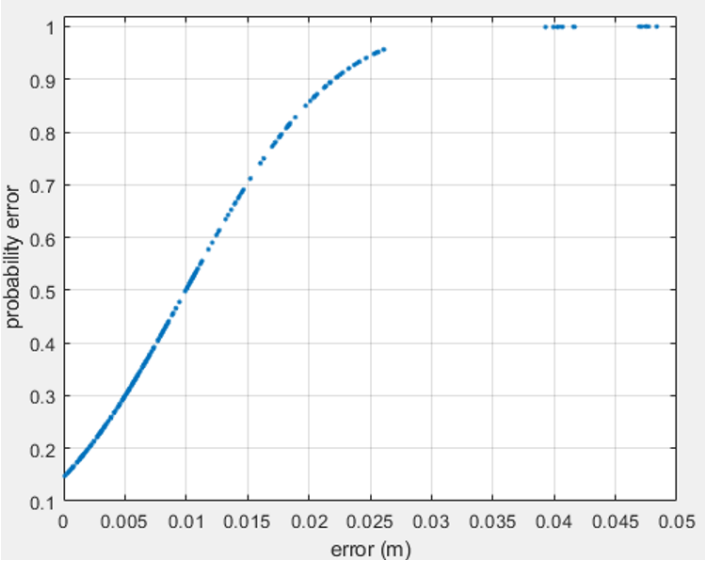

Figure 3.46: Cumulative distribution function for room position error values with neural network approach.

As previously performed, this system, is now, affected by Gaussian noise. Each signal detected by the receiver has a SNR of 5 dB, 10 dB, 20 dB, 40 dB and 80 dB and the receiver position is now estimated using the Trilateration with transmitters algorithm with the NN models help to predict the value of the variable r_t^2 when the system is now affected by a Gaussian noise. This system, is used to predict the receiver position when the receiver is placed in all 400 points of the room already shown in blue in the Figure 3.12. Using the CDF, the position error values are calculated, with a probability of 95%, depending on the SNR values, so that, for the referred SNR values, the position error values obtained are, respectively, 10.47 cm, 4.127 cm, 2.631 cm, 2.545 cm, 2.546 cm. These values can be seen in the Figure 3.47.

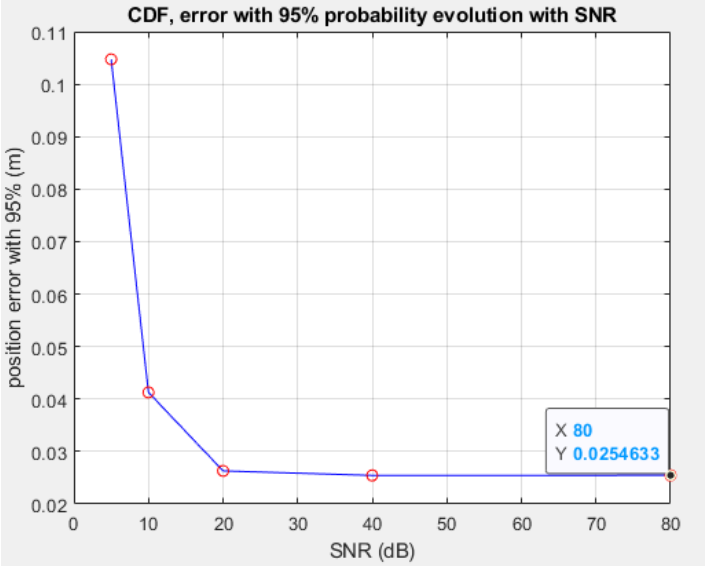

Figure 3.47: Results for trilateration neural network model when gaussian error is added to the system.

Neural Network Position Model

The developed SPR algorithm provides the distinction and estimation of each power value of each signal at each position in the room. Therefore, two NN models will be created, called $modelX$ and $modelY$, which provide, respectively, the coordinates X and Y, relative to the position of the photodiode receiver in the room, based on the four values of potency.

First, the relationship between the supplied powers of each of the four signals that arrive at the receiver with the coordinates X and Y are shown by Figures 3.48 and 3.49, respectively.

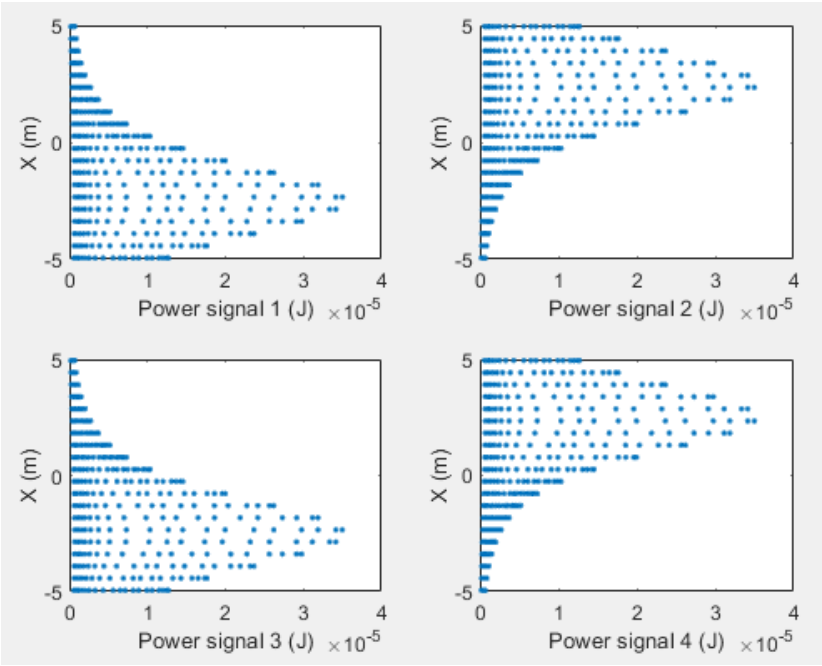

Figure 3.48: Relationship between the X coordinate and the power values of each signal, for the 400 positions of the room.

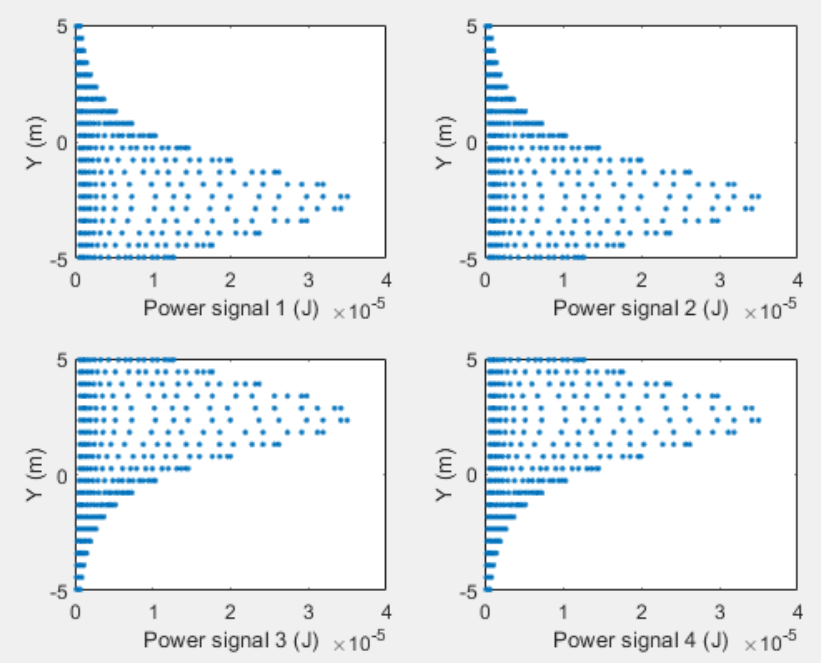

Figure 3.49: Relationship between the y coordinate and the power values of each signal, for the 400 positions of the room.

The total data being 400 points and the data used for training and validation being the 100 points that will be measured to create the 10x10 Fingerprint map, in practice, represented in green in the Figure 3.12, then the remaining 300 points are used for testing. In the Table 3.5, you can see a sample of the data used to train the NN models. Both models use the information on the power values of the four signals, however, the *modelX* uses the information related to the value of the coordinate X that characterizes the power value of the signals and the *modelY* uses the information relative to the value of the Y coordinate that characterizes the power value of the signals.

Power 1 (J)	Power 2 (J)	Power 3 (J)	Power 4 (J)	X (m)	Y (m)
6.4964×10^{-6}	5.7684×10^{-7}	5.7685×10^{-7}	1.9929×10^{-7}	-4.95	-4.9500
9.9351×10^{-6}	6.4878×10^{-7}	9.1185×10^{-7}	2.5748×10^{-7}	-4.95	-3.9079
1.2509×10^{-5}	6.8632×10^{-7}	1.4942×10^{-6}	3.2958×10^{-7}	-4.95	-2.8658
1.1990×10^{-5}	6.7953×10^{-7}	2.5239×10^{-6}	4.1441×10^{-7}	-4.95	-1.8237

Table 3.5: Data sample for neural network model training.

The values of the presented database are standardized, according to 2.44, before starting the training of the NN model, as shown in the Figures 3.50 and 3.51.

The implemented NN models have a structure such that they have 4 neurons in the input layer relative to the power value of each of the four signals, 1 neuron in the output layer, 3 hidden layers each consisting of 100 neurons and the function of activation for the calculations inside each neuron is the ReLU function.

In training the models, in order to reduce the value of RMSE, in order to improve their prediction and optimize the models, the hyper-parameters of the gradient, batch and epoch

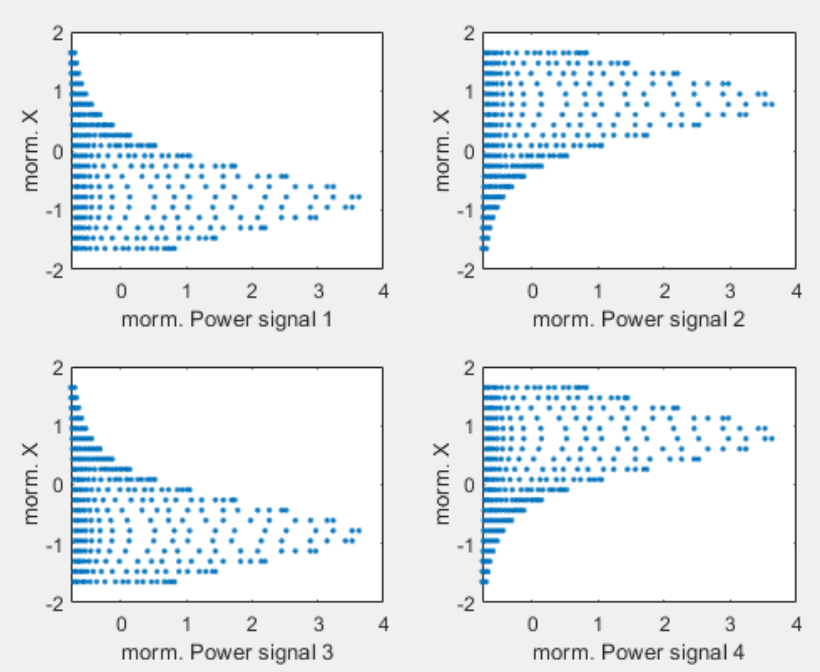

Figure 3.50: Relationship between the normalized values of x coordinate and the power values of each signal, for the 400 positions of the room.

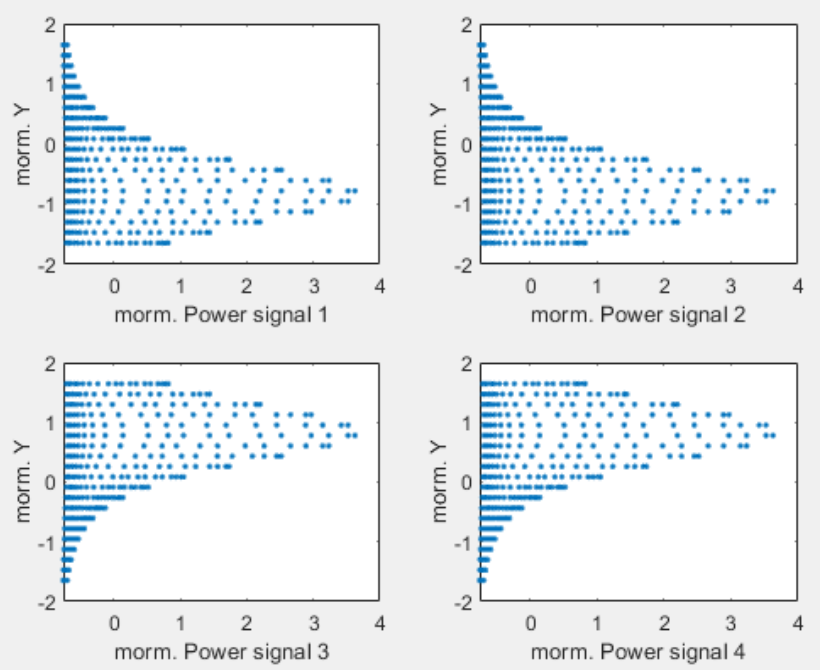

Figure 3.51: Relationship between the normalized values of y coordinate and the power values of each signal, for the 400 positions of the room.

functions are varied for the values $\{2, 5, 10, 15, 20, 25, 30\}$ and $\{10, 50, 100, 300, 500, 700, 1000\}$, respectively.

In the Figures 3.52 and 3.53, the values of the parameters *batch* and *epoch* are presented, such that, $batch - epoch$, for each RMSE value of the test data.

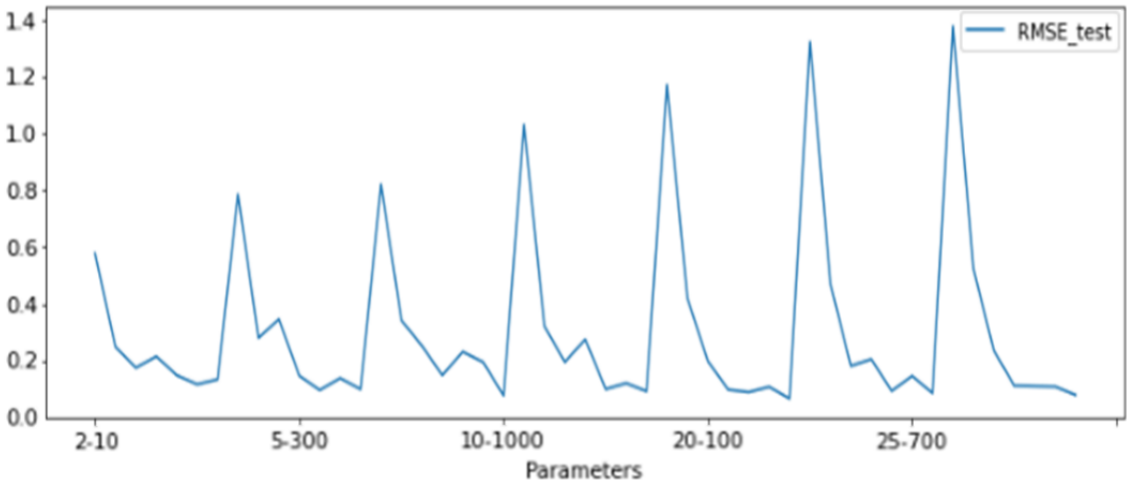

Figure 3.52: Root mean square deviation values for hyper-parameters variation, model X.

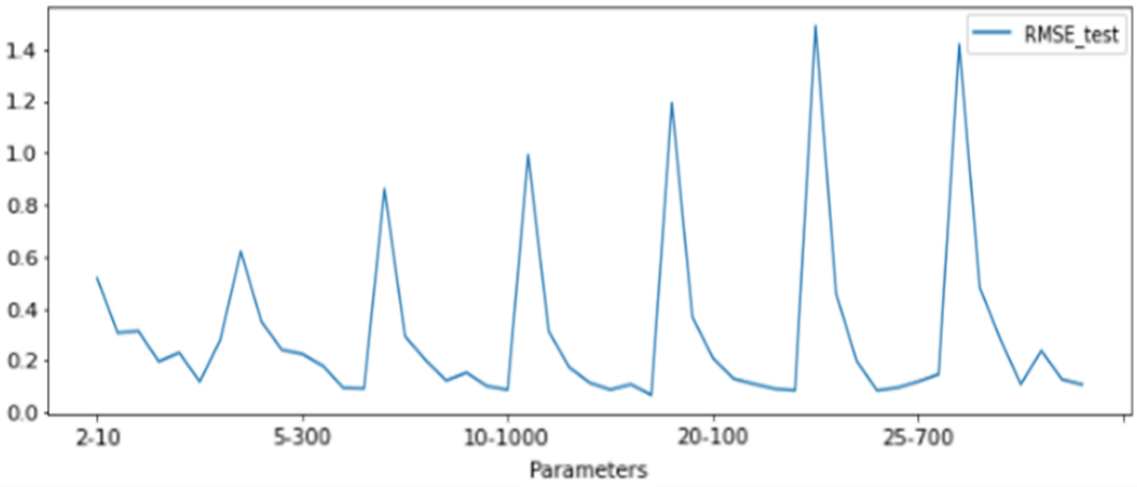

Figure 3.53: Root mean square deviation values for hyper-parameters variation, model Y.

From the Figures 3.52 and 3.53, for each batch the larger the epoch, the smaller the RMSE value of the test data and the greater the number of times the model is updated. It is also possible to observe that the maximum RMSE value verified for each batch increases as the batch size increases. In these Figures, for the chosen values the smallest test RMSE values obtained are of value 0.0686 and 0.0671 which occur for a batch size of 10 and 15, respectively, and an epoch number of value 1000.

Thus, we chose to train the $modelX$ and $modelY$ for a batch of sizes 10 and 15, respectively, and for 1000 epoch. The results obtained by optimizing the parameters of the models allowed to obtain the models $modelX$ and $modelY$ with a test RMSE value of 0.0673 and 0.0651, respectively. With these parameters, in the set of Figures 3.54, 3.55, 3.56 and 3.57 and in the set of Figures 3.58, 3.59, 3.60 and 3.61, it is possible to observe the predictions, for the test data, of the value of the coordinate X and Y, respectively. This set of Figures, represent the coordinates, relative to the position of the receiver in the room, made by the models $modelX$

and *modelY*, represented in orange, as well as, we have the original data values represented in blue, the same as those represented in the Figures 3.48 and 3.49.

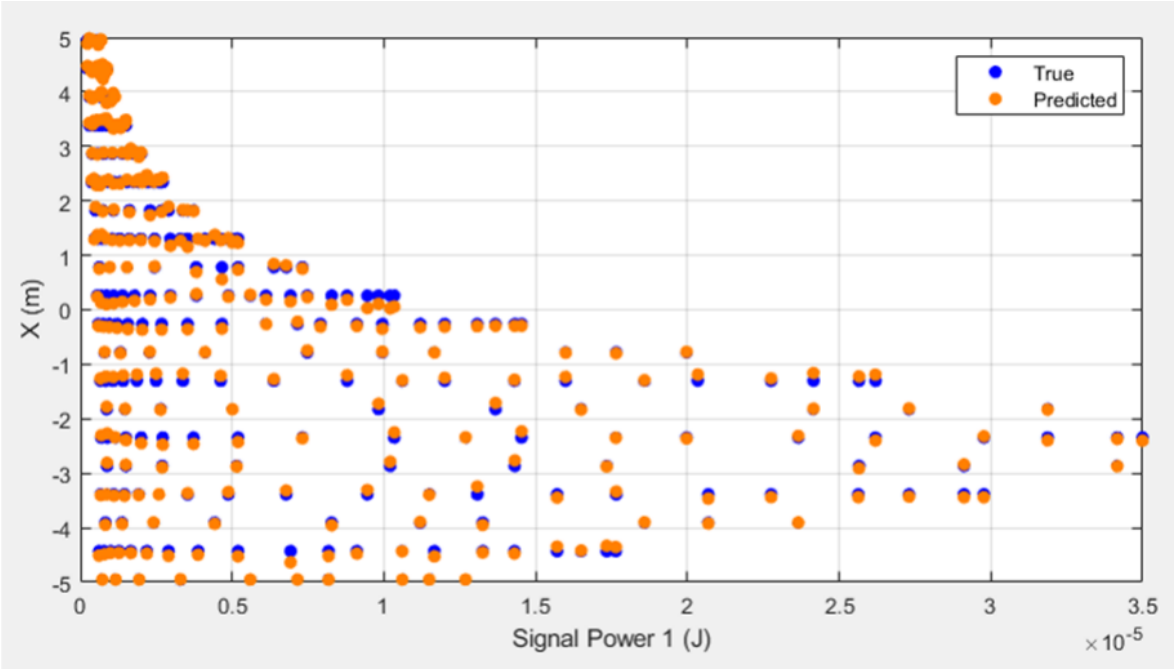

Figure 3.54: Optimized neural network model X predictions for signal 1 test data.

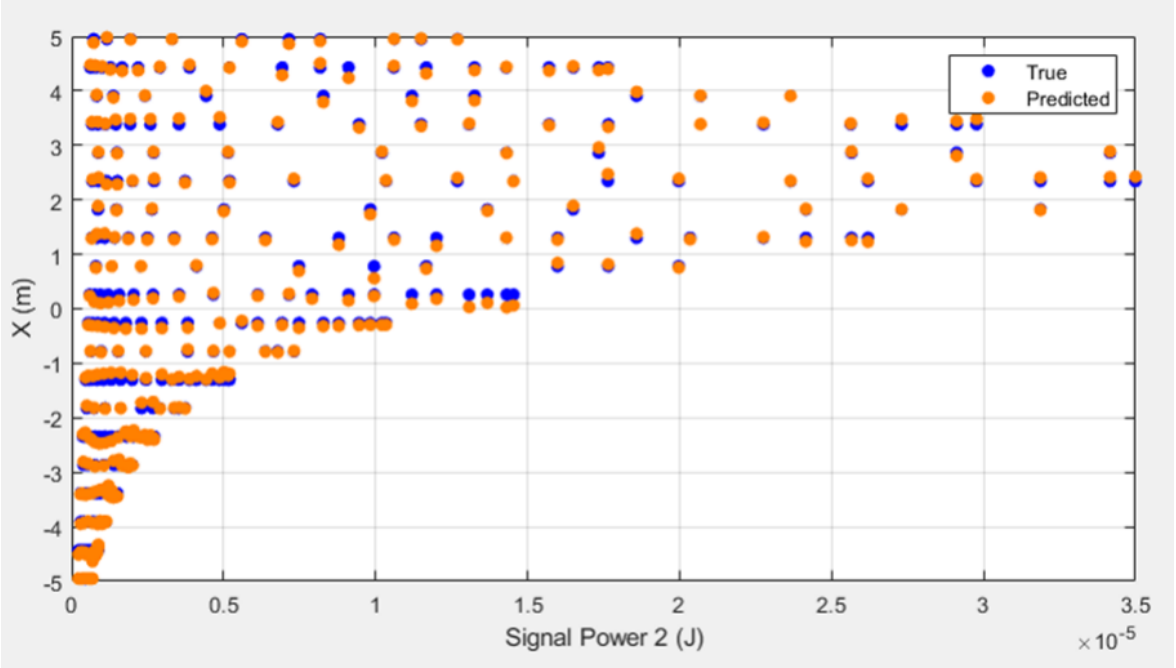

Figure 3.55: Optimized neural network model X predictions for signal 2 test data.

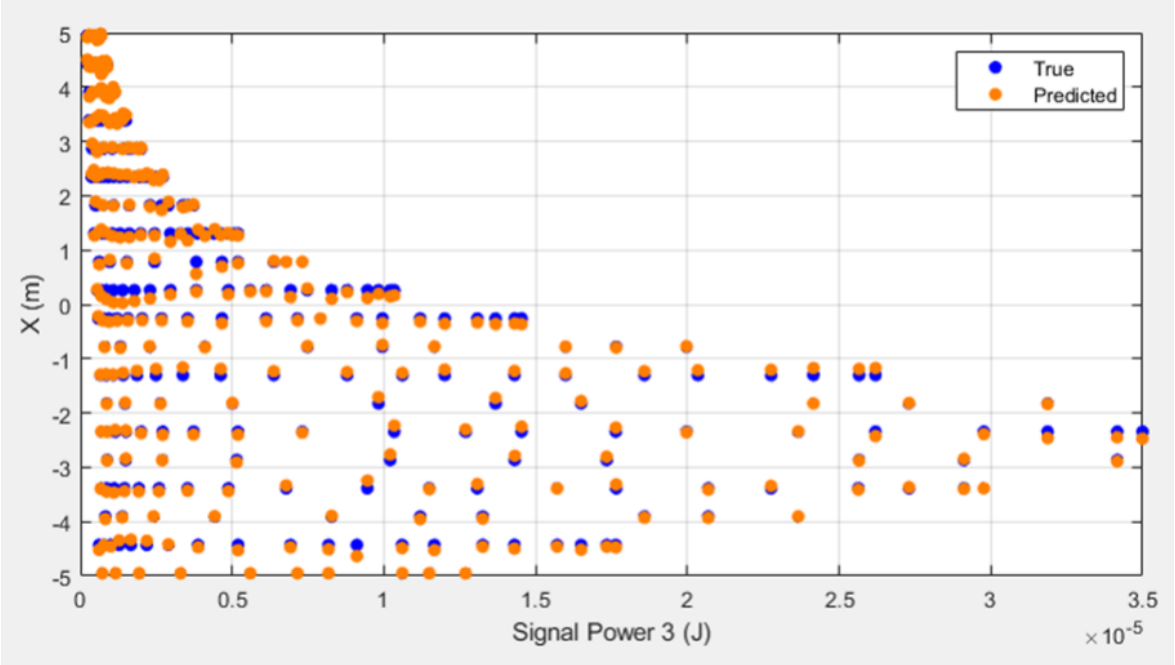

Figure 3.56: Optimized neural network model X predictions for signal 3 test data.

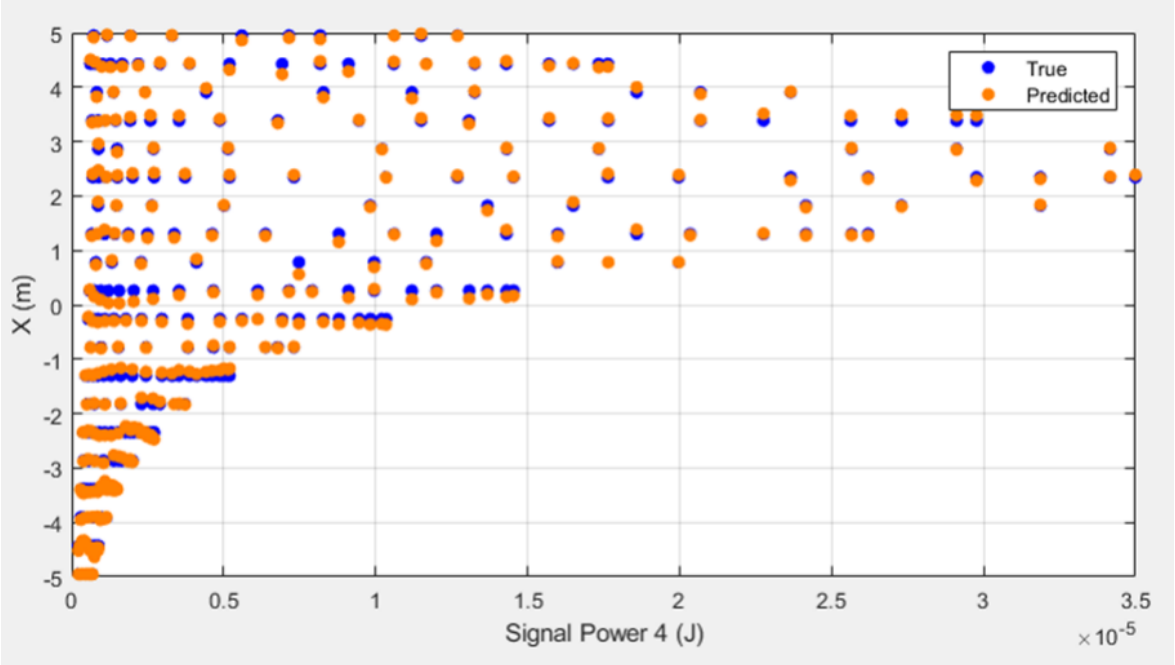

Figure 3.57: Optimized neural network model X predictions for signal 4 test data.

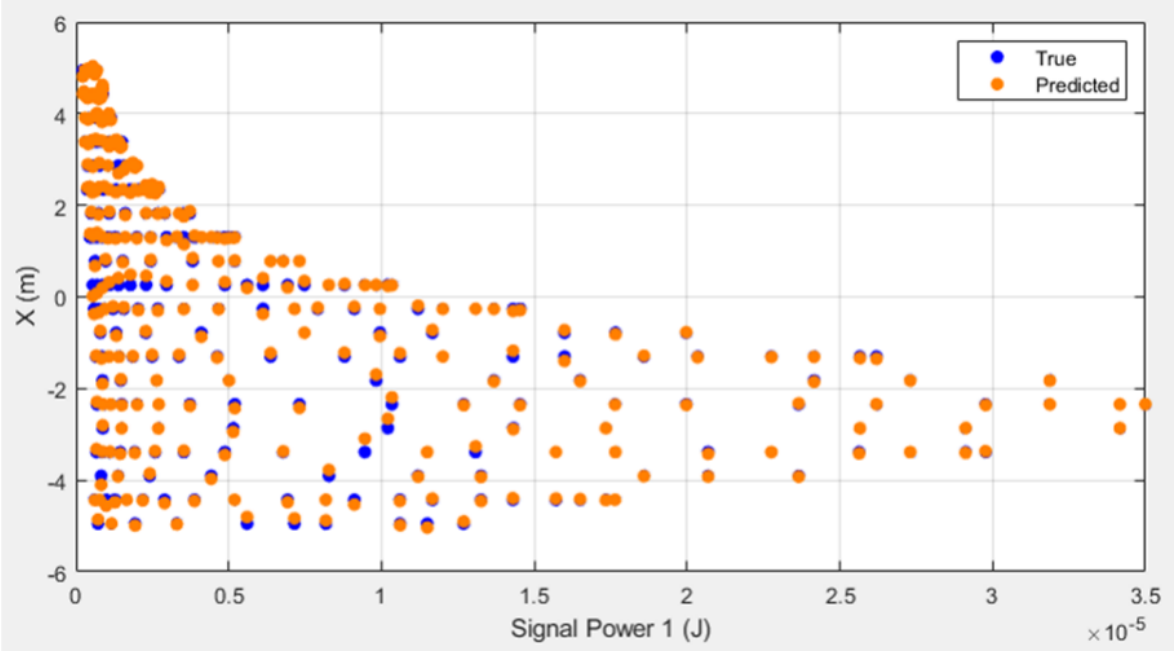

Figure 3.58: Optimized neural network model Y predictions for signal 1 test data.

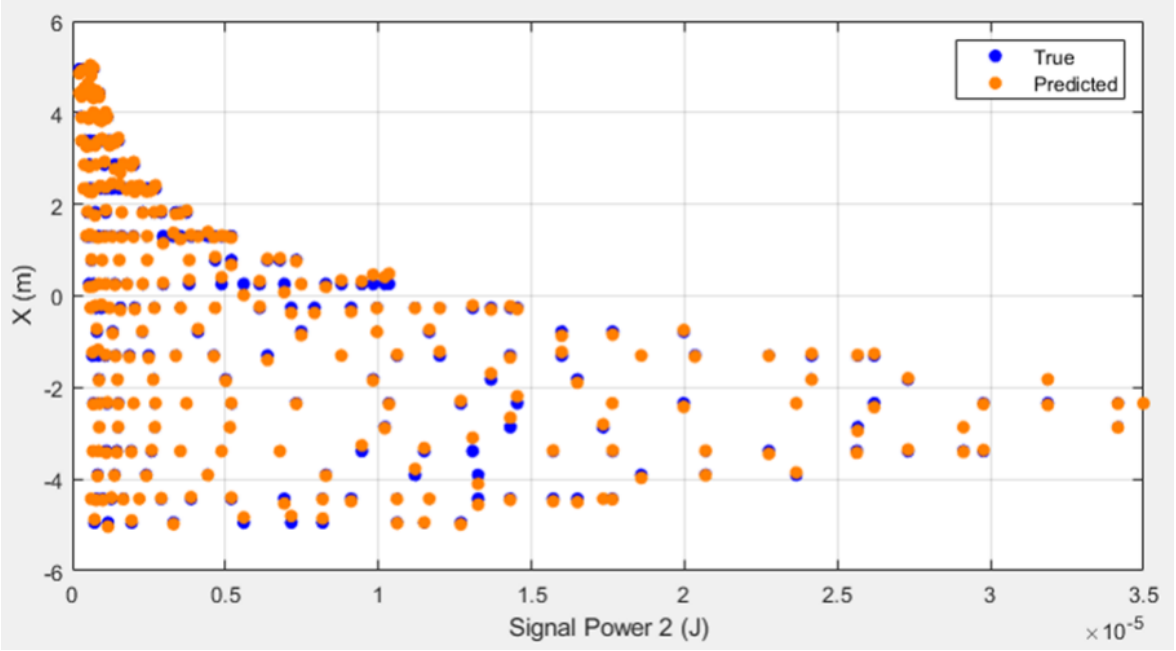

Figure 3.59: Optimized neural network model Y predictions for signal 2 test data.

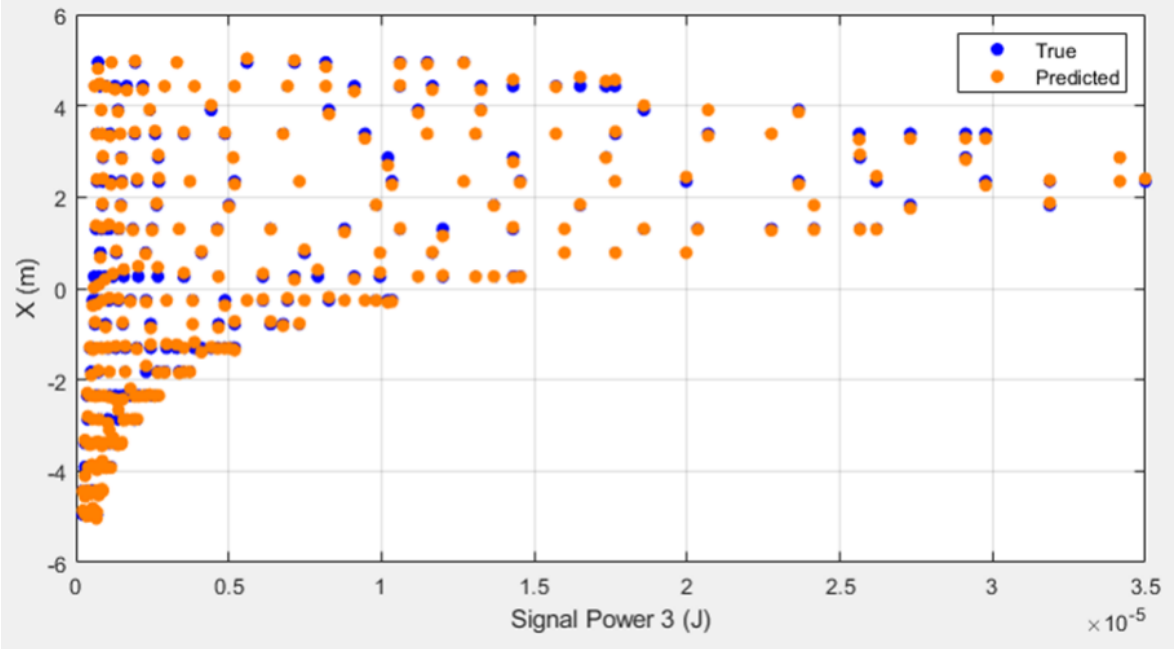

Figure 3.60: Optimized neural network model Y predictions for signal 3 test data.

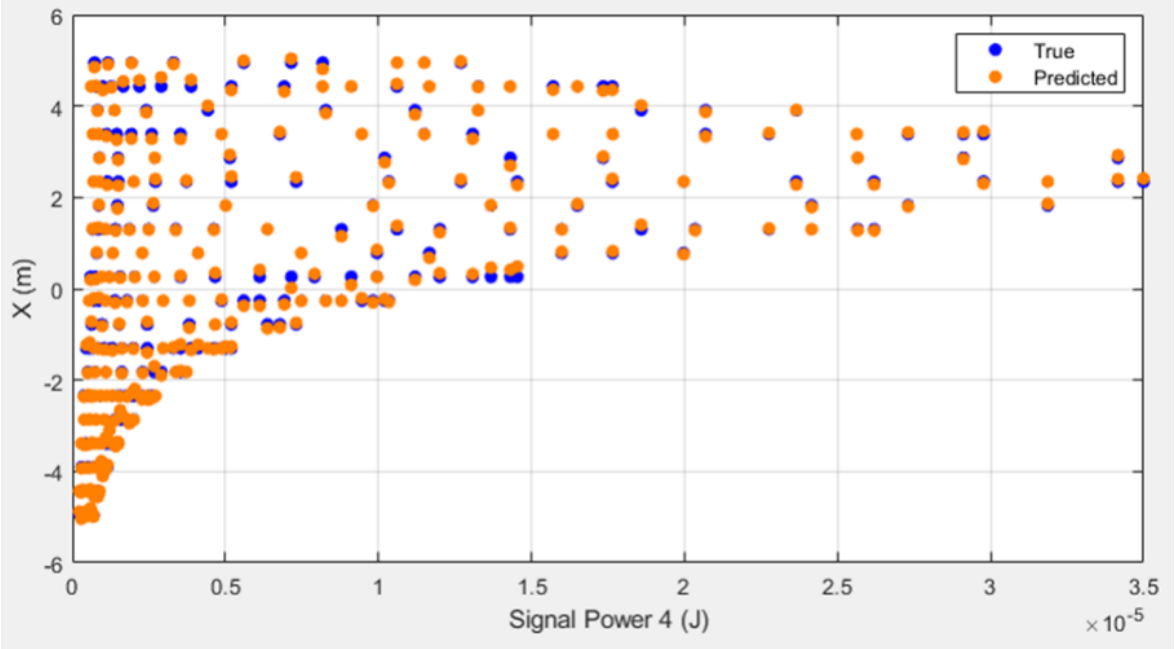

Figure 3.61: Optimized neural network model Y predictions for signal 4 test data.

Using the two models created to provide a prediction of the value of coordinates in the form (X, Y) referring to the photodiode receiver position in the room, from the estimated power value of each of the received signals 1, 2, 3 and 4 at the receiver, the position values (X, Y) of the receiver are estimated. Thus, the results presented in the Figure 3.62 were obtained.

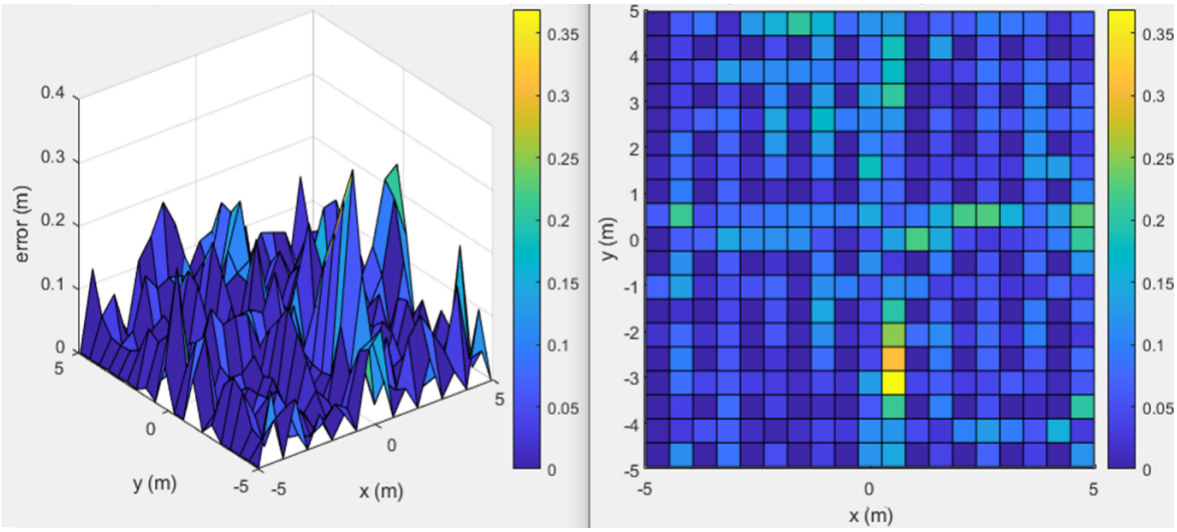

Figure 3.62: Results with neural network position model.

In the Figure 3.62, it is observed that the maximum position error obtained is 36.92 cm.

The CDF describing the probability of position error values is shown in the Figure 3.63. From this Figure 3.63, it can be seen that 90% of the values have less than 13.1 cm of position error and that 20% of the values have less than 8.132 mm of position error.

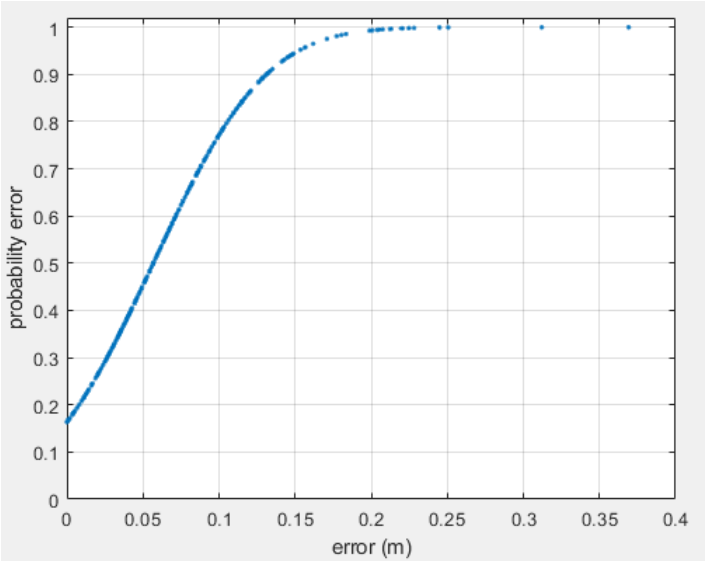

Figure 3.63: Cumulative distribution function for room position error values with neural network position model approach.

As previously performed, this system will now be evaluated on the Gaussian noise effect. Each signal detected by the receiver has a SNR of 5 dB, 10 dB, 20 dB, 40 dB, 60 dB and

80 dB and the receiver position is estimated, based on the NN Position Model, now affected by a Gaussian noise, when the receiver is placed in all 400 points of the room already shown in blue in the Figure 3.12. Using the CDF, the position error values are calculated, with a probability of 95%, depending on the SNR values, so that, for the referred SNR values, the position error values obtained are, respectively, 16.49 cm, 16.04 cm, 15.55 cm, 15.24 cm, 15.24 cm, 15.24 cm. These values can be seen in the Figure 3.64.

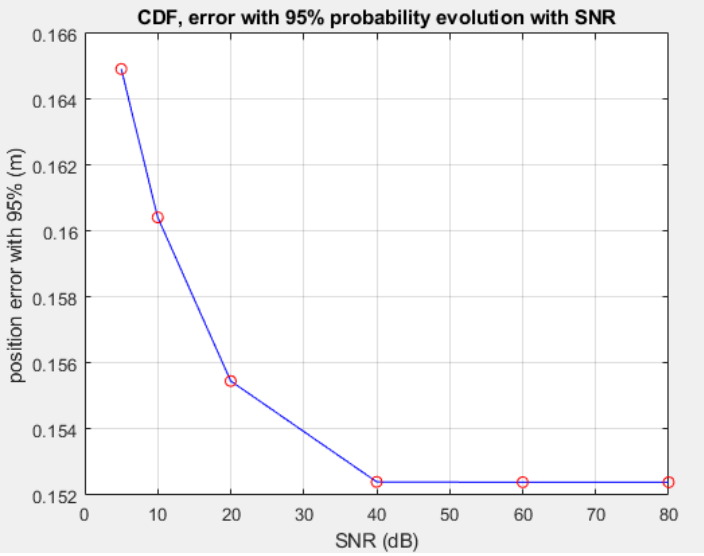

Figure 3.64: Results for neural networks Position Model when gaussian error is added to the system.

3.2.4 Results

The simulated results of the different position algorithm are added in this Section.

Position Algorithm		CDF at 95% error (m)		
		SNR 80 dB	SNR 10 dB	SNR 5 dB
TSM	Trilateration	5.23×10^{-3}	5.317×10^{-2}	11.56×10^{-2}
Trilateration with Transmitters	with NN model	2.546×10^{-2}	4.127×10^{-2}	10.47×10^{-2}
	without NN model	4.726×10^{-3}	2.943×10^{-2}	7.831×10^{-2}
NN position model		15.24×10^{-2}	16.04×10^{-2}	16.49×10^{-2}

Table 3.6: Position algorithm results.

In the Table 3.6, is presented the error evaluation of the implemented position algorithm when Gaussian noise affects the system and the trilateration with transmitters algorithm has the smaller error value for each of the SNR values. Through the Table 3.6, it can be seen that for the NN position model, as the noise power increases, the error value, which occurs with a probability of 95%, does not increase so sharply as in the other algorithms. This algorithm is little influenced by noise. However, the NN position model is the one with the larger error values for each of the SNR values.

The TSM with interpolation method did not present a very efficient solution because the probability value of the error being less than 0.48 m and 50.9 cm is 90% when the system is not affected with Gaussian error and when the weight function is implemented based on the exponential function and the cosine function.

CHAPTER 4

Conclusion

The system developed in this dissertation in an simulation environment, consisting of a photodiode receiver that receives the signal transmitted by 4 transmitters. The signal that is detected in the photodiode receiver is a mixture of the 4 signal codes transmitted by the 4 LEDs that are part of the system. This mixed signal that is provided by the receiver is composed of the sum of these 4 signals, which are detected by the receiver, after suffering propagation losses in the channel.To determine the position of the receiver is necessary to know the DC gain value of each of the 4 signal detected by the receiver.

Therefore, the SPR method is developed, which provides the information about the DC gain in the four signals that are detected in the receiver, taking the mixed signal as input. In simulation, the SPR algorithm determines the DC gain value of signals with a high accuracy having a maximum error of 8.9443×10^{-10} m, whose relative error value is approximately 0.0138%.

With the DC gain values provided by SPR are developed five position algorithms capable of estimating the coordinates of the receiver position using the DC gain value of each of the 4 signal detected by the receiver, being these, TSM with trilateration and interpolation, Trilateration with Transmitters with and without NN models adjustments and the NN position model.

In the Fingerprint Search method it is verified that the lower the resolution of the Fingerprint map, the greater the error associated with it. However, the higher the resolution, the greater the number of points and the time to create the Fingerprint map. Therefore, it is not practical to measure an infinite number of points inside the room in order to minimize the position error value of this algorithm as much as possible.

The Trilateration with Transmitters algorithm describes the position error caused by the reflection of the transmitted signals in the simulated system. Therefore, this algorithm does not consider the NLOS propagation channel to estimate the position of the photodiode receiver.

The NN model applied to this Trilateration with Transmitters algorithm combats the effect

that reflected signals have in determining the receiver's position. However, the error value of the verified Trilateration with Transmitters and NN model algorithm is higher than the error value of the Trilateration with Transmitters algorithm. The NN model developed was not able to reduce the influence of simulated reflections that affect this trilateration algorithm.

In this simulations, it was verified that the NN models are little affected by the increase of Gausian noise added to the system. However, these models also have a higher positioning error, caused by the low complexity of the developed NN models and the low amount of data for training these models.

CHAPTER 5

Future Work

In a future iteration of this work, the SPR algorithm can be improved, testing and studying the system further for new frequencies in order to minimize the error verified in the odd harmonics of the FFT. Also, the NN models could be tested with more training data in order to improve the prediction of the NN models. Finaly, the application of the algorithms developed in this dissertation could be tested with practical data and can be adjusted to estimate the height of the receiver as well, being able to determine the spatial coordinates of the position of the receiver.

CHAPTER 6

References

[1] F. R. Gfeller and U. Bapst, "Wireless in-house data communication via diffuse infrared radiation," *Proceedings of the IEEE*, vol. 67, no. 11, pp. 1474–1486, 1979.

[2] R. Ramirez-Iniguez, S. M. Idrus, and Z. Sun, *Optical wireless communications: IR for wireless connectivity*. Auerbach Publications, 2008.

[3] E. Ciaramella, Y. Arimoto, G. Contestabile, *et al.*, "1.28 terabit/s (32x40 gbit/s) wdm transmission system for free space optical communications," *IEEE Journal on selected areas in communications*, vol. 27, no. 9, pp. 1639–1645, 2009.

[4] V. Jungnickel, T. Haustein, A. Forck, and C. Von Helmolt, "155mbit/s wireless transmission with imaging infrared receiver," *ELECTRONICS LETTERS-IEE*, vol. 37, no. 5, pp. 314–314, 2001.

[5] J. Bellon, M. J. Sibley, D. Wisely, and S. Greaves, "Hub architecture for infra-red wireless networks in office environments," *IEE Proceedings-Optoelectronics*, vol. 146, no. 2, pp. 78–82, 1999.

[6] J. Kahn and J. Barry, "Wireless infrared communications," *Proceedings of the IEEE*, vol. 85, no. 2, pp. 265–298, 1997. DOI: 10.1109/5.554222.

[7] S. Rajbhandari, W. Popoola, and Z. Ghassemlooy, *Optical Wireless Communications: System and channel modelling with MATLAB®*. CRC Press, 2013. DOI: 10.1201/b12687.

[8] D. Kalman, "A singularly valuable decomposition: The svd of a matrix," *The college mathematics journal*, vol. 27, no. 1, pp. 2–23, 1996.

[9] A.-J. Van Der Veen, E. Deprettere, and A. Swindlehurst, "Subspace-based signal analysis using singular value decomposition," *Proceedings of the IEEE*, vol. 81, no. 9, pp. 1277–1308, 1993. DOI: 10.1109/5.237536.

[10] X. Lei, J. Guo, and C. Zhu, "Vision-based faint vibration extraction using singular value decomposition," *Mathematical Problems in Engineering*, vol. 2015, 2015.

[11] M. Heideman, D. Johnson, and C. Burrus, "Gauss and the history of the fast fourier transform," *IEEE ASSP Magazine*, vol. 1, no. 4, pp. 14–21, 1984. DOI: 10.1109/MASSP.1984.1162257.

[12] J. Herrnsdorf, M. D. Dawson, and M. J. Strain, "Positioning and data broadcasting using illumination pattern sequences displayed by led arrays," *IEEE Transactions on Communications*, vol. 66, no. 11, pp. 5582–5592, 2018. DOI: 10.1109/TCOMM.2018.2850892.

[13] J. Herrnsdorf, M. J. Strain, E. Gu, R. K. Henderson, and M. D. Dawson, "Positioning and space-division multiple access enabled by structured illumination with light-emitting diodes," *Journal of Lightwave Technology*, vol. 35, no. 12, pp. 2339–2345, 2017. DOI: 10.1109/JLT.2017.2672864.

[14] A. Griffiths, J. Herrnsdorf, J. McKendry, M. Strain, and M. Dawson, "Gallium nitride micro-light-emitting diode structured light sources for multi-modal optical wireless communications systems," *Philosophical Transactions of the Royal Society A*, vol. 378, no. 2169, p. 20 190 185, 2020.

[15] T. Maekawa, Y. Matsumoto, and K. Namiki, "Interpolation by geometric algorithm," *Computer-Aided Design*, vol. 39, no. 4, pp. 313–323, 2007, ISSN: 0010-4485. DOI: https://doi.org/10.1016/j.cad.2006.12.008.

[16] L. Asmaa, K. A. Hatim, and M. Abdelaaziz, "Localization algorithms research in wireless sensor network based on multilateration and trilateration techniques," in *2014 Third IEEE International Colloquium in Information Science and Technology (CIST)*, 2014, pp. 415–419. DOI: 10.1109/CIST.2014.7016656.

[17] S. Patil and M. Zaveri, "Mds and trilateration based localization in wireless sensor network," *Wireless Sensor Network*, vol. 03, no. 6, pp. 198–208, 2011.

[18] J. Padmanabhan and M. J. Johnson Premkumar, "Machine learning in automatic speech recognition: A survey," *IETE Technical Review*, vol. 32, no. 4, pp. 240–251, 2015.

[19] J. Lemley, S. Bazrafkan, and P. Corcoran, "Deep learning for consumer devices and services: Pushing the limits for machine learning, artificial intelligence, and computer vision.," *IEEE Consumer Electronics Magazine*, vol. 6, no. 2, pp. 48–56, 2017.

[20] B. He, K.-i. Shu, and H. Zhang, "Machine learning and data mining in diabetes diagnosis and treatment," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, vol. 490, 2019, p. 042 049.

[21] H. U. Dike, Y. Zhou, K. K. Deveerasetty, and Q. Wu, "Unsupervised learning based on artificial neural network: A review," in *2018 IEEE International Conference on Cyborg and Bionic Systems (CBS)*, IEEE, 2018, pp. 322–327.