The front page is supposed to lie at this place, use of the bruface template required.

Abstract

This is the abstract blablabla...
Keywords: Ultrawide Band, ...

Acknowledgements

I thank ...

Contents

Chapter 1

Introduction

But du mémoire, introduction, blablabla
Blablabla... [16]

Chapter 2

State of the art

This chapter outlines the state of the art. The first part focus on the implementation of a locating system, presented after a brief introduction to the Ultra-Wideband (UWB) and Real-time locating systems (RTLS). Next, the concept of virtual anchors and multi-path aided locating systems is discussed.

2.1 Ultra-Wideband Technology

UWB is a communication technology using, as the name states, a large bandwidth. This is not a new technology as it is the one used by Guglielmo Marconi for the first transatlantic communication using radio waves [19]. As define by the International Telecommunication Union Radiocommunication Sector (ITU-R) to be considered as UWB, the bandwidth of communication must be at least 20 % of the arithmetic center frequency [20].

One interesting feature of UWB is the possible coexistence with other radio waves already present in the environment such as Wireless Fidelity (Wi-Fi). As it can be seen on Fig. 2.1, the extension of the UWB in the spectral domain is quite large.

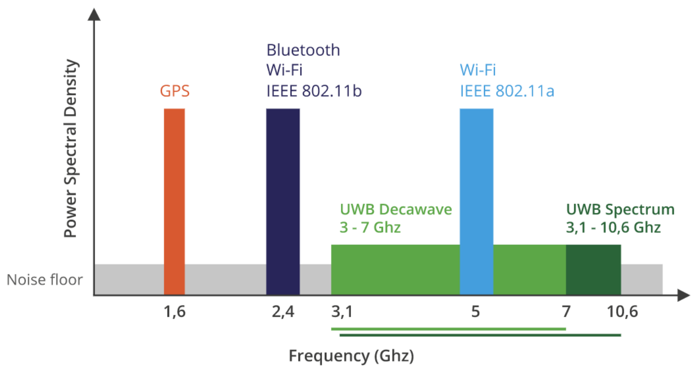


Figure 2.1: UWB spectrum compared to Wi-Fi and other wireless technology. Taken from [20].

Knowing this and based on the time-frequency duality reminded in eq. 2.1, one can see that the extension in the time domain will be quite small compared to other signals type. This is a direct effect of the Fourier transform, the extension of a function and it's extension of it's Fourier transform are inversely proportional. This follows the principle of uncertainty assessing that a trade-off needs to be done between the precision reached in the time domain and the one in the frequency domain [8].

$$x(at) \longleftrightarrow \frac{1}{|a|} * X(\frac{f}{a}) \tag{2.1}$$

The Fig. 2.2 shows the theoretical duration of an impulse of the UWB. One can see that the time extension of the significant part of the pulse is from 0.3 to 0.7 nanoseconds, leading to a pulse duration of 0.4 nanoseconds.

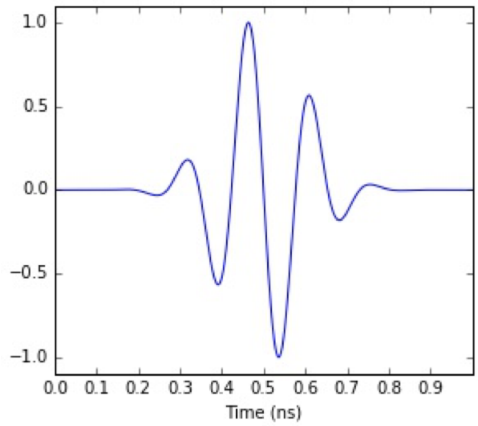


Figure 2.2: Theoritical duration of an UWB pulse. Taken from [7].

An advantage of the UWB is its robustness in regard of the Multipath Channels (MPCs) EXPLIQUER CE QUE SONT LES MPCS . This can be understood by looking at Fig. 2.3, where several peaks can be distinguished, each corresponding to a different path travelled by the wave. Indeed, the probability to have a collision depends on the size of the pulse sent. From this, the interest of the UWB in confined area appears as a lot of MPC are present due to the reflections coming from all walls of a room.

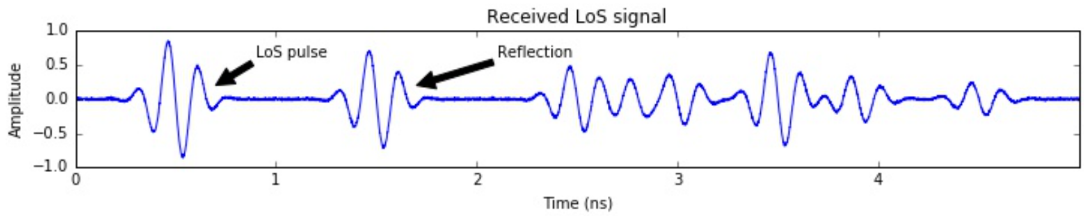


Figure 2.3: Example of an MPC with a Line of Sight (LoS) and a Non Line of Sight (NLoS). Taken from [7]DEFINIR LOS ET NLOS

2.2 Real Time Locating Systems

RTLS are systems used to track and identify the location of objects in real time. This is a rather vague definition since nothing is specified concerning the means employed to achieve the localization. The RTLS that will be presented in this section will all have in common the use of wireless communications, between devices called "anchor" and "tag", the tag being associated with the object to locate while the anchor is at a fixed and known location.

Those RTLS can be separated in two categories : "Relative localization" and "Absolute localization". The relative localization algorithm presented in section 2.2.1 is the Time of Flight (ToF) method that is used in this project to compute the distance between an anchor and a tag. This choice has been made and explained in [10], [14] alongside a presentation of several approaches to determine the relative position of a tag relatively to an anchor. Detailler plus cette partie

2.2.1 Symmetric double sided two-way ranging

Symmetric double-sided two-way ranging (SDS-TWR) consists in an exchange of three messages between two Ranging-capable DEVices (RDEVs), respectively '$RDEV_1$' initiating the communication and '$RDEV_2$'. Each device needs to save the Time of Emission (ToE) or Time of Arrival (ToA) of every message. Those times being respectively t_0, t_1 for the first message, t_2, t_3 for the second message and t_4, t_5 for the last message.

Each message contains the different timestamps previously computed, meaning that at the end of this exchange $RDEV_2$ possess all the informations about the timestamps, while $RDEV_1$ misses the last one. If one wants $RDEV_1$ to be able to compute the ToF then a last message containing t_5 should be exchanged.

A schematic of the exchanges between $RDEV_1$ and $RDEV_2$ that occurs in SDS-TWR is shown in Fig. 2.4.

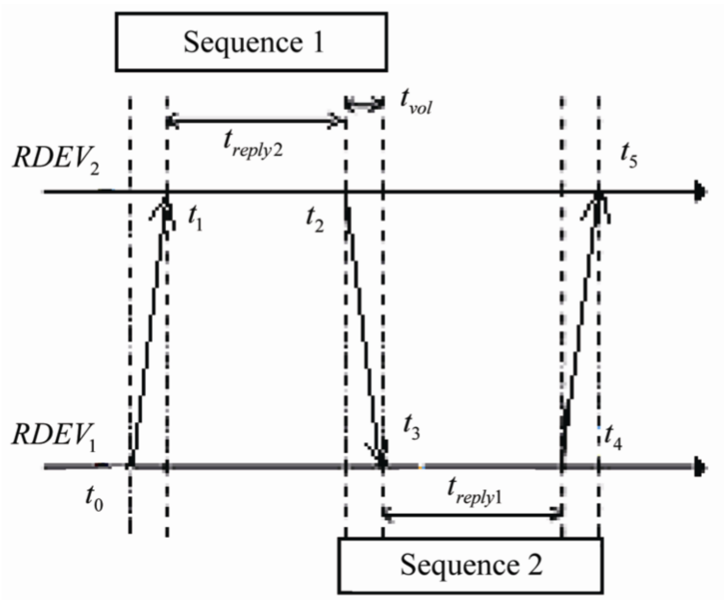


Figure 2.4: Symmetric double-sided two-way ranging. Taken from [3].

Based on those timestamps, the computation of the ToF can be observed in eq. 2.2. This estimation is the arithmetic means of the ToF from '$RDEV_1$' to '$RDEV_2$' and vice-versa.

$$t_{est} = \frac{((t_3 - t_0) - (t_2 - t_1)) + ((t_5 - t_2) - (t_4 - t_3))}{4} \tag{2.2}$$

Since that ToF computed remains an estimation, it is important to know the magnitude of the error as well as its evolution in parallel of the true value of the ToF.

$$t_{true} - t_{est} = \frac{1}{4} * ((t_2 - t_1) - (t_4 - t_3)) * (e_1 - e_2) \tag{2.3}$$

The term $e_1 - e_2$ being the difference between the internal clocks of both devices. [3]

2.2.2 Trilateration

The SDS-TWR allowing us to compute the ToF, the relative distance can be computed using the light celerity. If the relative distance between a tag and three different anchors is known, it is possible to compute the intersection of three circle having as center the position of the anchor and radius corresponding to the ToF associated to this anchor. A scheme displaying that solution can be seen on Fig. 2.5.

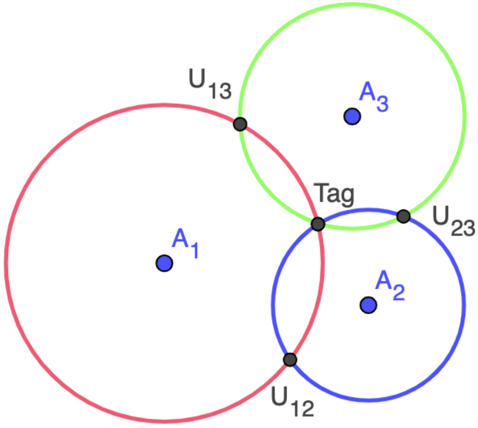


Figure 2.5: Trilateration

As one can deduce, in a two dimensional plan, three anchors are needed to have an intersection of only one point, removing the uncertainty on the solution. If only two anchors where used, one would obtain two possible solution : Tag and U_{XY}. In a three dimensional space, four anchors would be needed. This actually corresponds to the following system of equations :

$$\begin{cases} (x_1 - x_0)^2 + (y_1 - y_0)^2 = d_1^2 \\ (x_2 - x_0)^2 + (y_2 - y_0)^2 = d_2^2 \\ (x_3 - x_0)^2 + (y_3 - y_0)^2 = d_3^2 \end{cases} \tag{2.4}$$

Where (x_i, y_i) corresponds to the position of the anchor i and d_i corresponds to the distance between this anchor and the tag, (x_0, y_0) being the position of the tag.

Uncertainties

Due to the inaccuracy of the computed distances, the system 2.4 can not be solved. There is a lot of probability that there is not a single point (x_0, y_0) that solves it. To avoid this problem, the estimator $S(\vec{p})$ developed in [21] is used.

$$\begin{aligned} S(\vec{p_0}) &= \sum_i^N (\|\vec{p_i} - \vec{p_0}\|^2 - d_i^2)^2 \\ \vec{p_0} &= \underset{\vec{p}}{\operatorname{argmin}}\ S(\vec{p}) \end{aligned} \tag{2.5}$$

Where $\vec{p_k} = (x_k, y_k) \forall k \in 0, ..., N$ in the two dimensional cases, N being the number of anchors[1]. The objective of this estimator is to find the estimate $\vec{p_0}$ minimizing the value of $S(\vec{p_0})$.

[1]Which can be superior to 3, even in a 2D plan.

2.3 Implementation of a locating system

Using the technology briefly presented in sections 2.1 and 2.2, a locating system has been developed by Quentin Fesler and Cédric Hannotier in [10], [14]. This system is able to retrieve a localization with an error oscillating between twenty and fifty centimetres inside of a building [13].

This locating system is composed of fixed antennas called anchors, based on ESP8266 as micro-controller and a Decawave DWM1000 UWB transceiver[5]. The tag are built using an Android cellphone, a PSoC[2] and also a DWM1000 module.

2.3.1 DWM1000

The DWM1000 is the antenna chosen to operate the wireless communication, it will be needed for the tag as well as for the different anchors. The configuration of those antennas and the Serial Protocol Interface (SPI) communication are both explained in this section.

Figure 2.6: DWM1000 module. Taken from [5].

Configuration

Before using the DWM1000, tests have been conducted to choose a configuration minimizing the error rate and the consumption while maximizing the communication speed. This leads to the following choices[3] :

- Channel number : 5
- Bitrate : 6.8 Mbits^{-1}
- Pulse Repetition Frequency (PRF) : 16MHz
- Preamble length : 128 bits

The chosen channel number is the number 5 partly due to the European Union (EU) regulations that are more strict in the frequencies bounds from 3.1 to 4.8 GHz than in the frequencies bounds from 6 to 9 GHz[4]. The other channel that is in those more boundaries in the 7^{th} one. The difference being a bandwidth being twice as largei[4]. The channel 7 also lies within this region of the electromagnetic spectrum, but was not chosen. The difference between those channel resides in the size of the bandwidth.

The choices of the bitrate are restricted between 110 kbits^{-1}, 850 kbits^{-1} or 6800 kbits^{-1}. The reasons behind the choice of the bitrate at 6.8 Mbits^{-1} are detailed in [14]. This

[2]The exact model is the : CY8C5888LTI-LP097 [13].

[3]A more detailed discussion on the choice of those parameters can be found in [14].

[4]The bandwidth of the 5^{th} one is 499.2MHz while the one from the 7^{th} is 1081.6 MHz.

principal reason being the existence of a "Smart Transmit Power Control" allowing to increase the power of transmission under certain conditions, one of them being that the bitrate must be fixed at 6.8 Mbits^{-1}.

The PRF can be chosen between 16 MHz and 64 MHz, an higher one increasing the operating range while consuming more power.

The preamble length is used to estimate the channel estimation, which describes how a signal propagates from the transmitter to the receiver, in order to perform the equalization to flatten the frequency response. The precision of the estimation increases with the size of the preamble. Unfortunately, a longer preamble means that a larger proportion of the energy consumed will not be used effectively, thus decreasing the energy efficiency of the setup, and that a shorter time window will be available to transmit actual data. Recommended bitrate in function of the channel are proposed in [6].

Control

The DWM1000 is piloted via an SPI bus, this communication follows a master-slave scheme where the master, which is the micro-controller, controls the communication [2]. On Fig. 2.7, the four needed signals are displayed. Master Input, Slave Output (MISO) and Master Output, Slave Input (MOSI) are the connections used to transmit the data between the master and its slave. The Serial Clock (SCKL), generated by the master fixes the speed at which the MISO and MOSI exchanges data. Since the SPI allows different slaves for only one master, the Slave Select (SS) is used by the master to select with which specific slave it communicates.

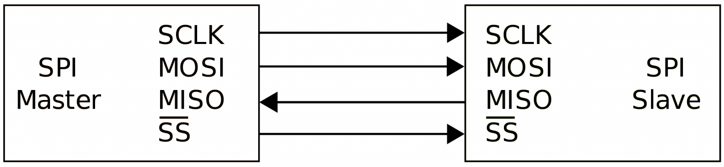


Figure 2.7: SPI Schematic.

2.3.2 Anchor

The anchors are fixed antennas composed of a DWM1000 and an ESP8266 [9]. They are placed at known position in the room and are used to compute the ToF between them and the tag using the SDS-TWR, as explained in section 2.2.1.

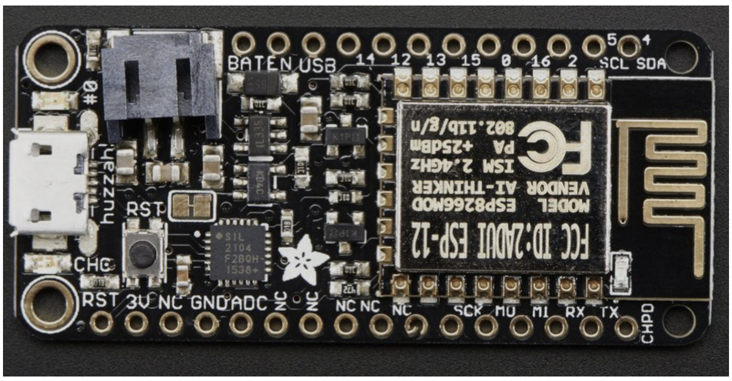


Figure 2.8: ESP8266 mounted on a Feather Huzzah board. Taken from [1]

The micro-controller has been combined with the development board Feather Huzzah from Adafruit [1]. This module is represented in Fig. 2.8. This board can be flashed using a USB serie connection, allowing a easy deployment of the code. It also has the advantage of being light and small, a feature which is useful when one wants to deploy several anchors in a room.

2.3.3 Tag

The tag is the object whose localization has to be determined. It is composed of a DWM1000 antenna, a PSoC[5] and an Android application. The Fig. 2.9 shows the PSoC used as well as its custom board made by the electronic BEAMS service of the ULB.

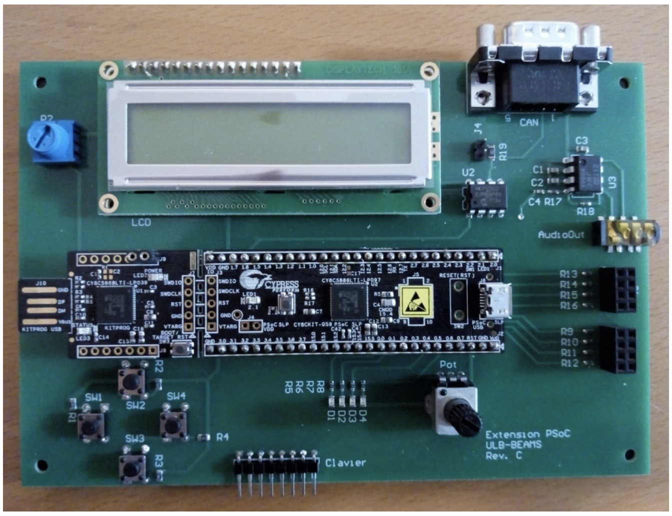


Figure 2.9: PSoC micro-controller mounted on a custom board made by the BEAMS service.

The communication between the DWM1000 an the PSoC is performed using a SPI bus as for the anchors, the PSoC being the master. As for the ESP8266, the PSoC can be flashed through an USB bus. The micro-controller receives instructions from the application on the cellphone and controls the communications of the DWM1000 with the different anchors. It then transmits the received data from the DWM1000 to the application through an USB bus.

2.3.4 Android Application

To control the PSoC, an Android application has been developed. A screen-shot of the main window can be seen in the appendix. A. It exhibits four buttons, whose functionalities are presented in what follows.

Navigation

The Navigation button opens a map of an environment[6] and an arrow is displayed at the estimated location of the tag, the orientation being the estimated orientation of the cellphone. The coordinates are also shown, computed from the bottom left corner of the map. The used map is shown in Fig. 2.10.

[5]The exact model is the CY8C5888LTI-LP097 [13].
[6]The room UA5.214 in this case, which is one of the electronic lab at the ULB.

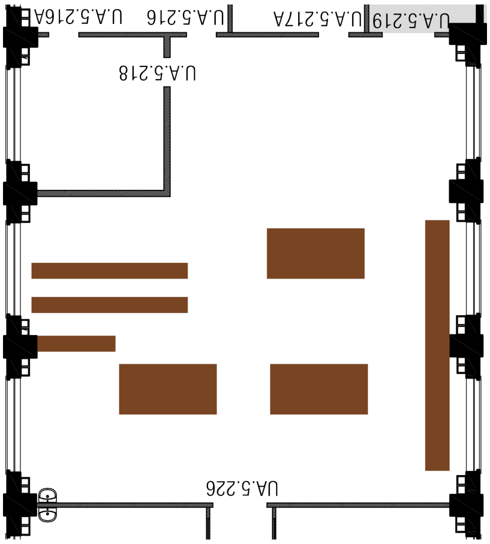


Figure 2.10: Map of the UA5.214

When this function runs, the application enters a loop that continuously requests the PSoC to perform a SDS-TWR with each anchor. The data containing the different timestamps are transmitted towards the application for each anchor separately. From those data, the distance between each anchor and the tag is computed on the cellphone as well as the trilateration algorithm described in section 2.2.2, leading to an estimate of the location. The biggest computations are done on the cellphone because of the bigger computational power available in comparison to the PSoC.

A detailed state machine representing the interactions between the tag and an anchor can be found in appendix make a link to an annex and put the state machine in annex [14]. While the anchor only performs one SDS-TWR at a time, the tag needs to keep an history of the anchors contacted to perform the trilateration afterwards.

Test USB connection, Test orientation

The Test USB connection allows to test that the USB communication bus used to communicate with the PSoC is fully operational. The test procedure consists of sending a 16 bits long messages to the PSoC, this message being : `0x0406`. If the PSoC is well connected, the application is supposed to receive a 32 bits long message : `0x02034637`. This feature allows to quickly debug the USB communication.

The "Test orientation" has been designed to test the detection of the orientation of the cellphone. The three angles necessary to characterize the orientation of the device are respectively the Azimuth, the Pitch and the Roll.

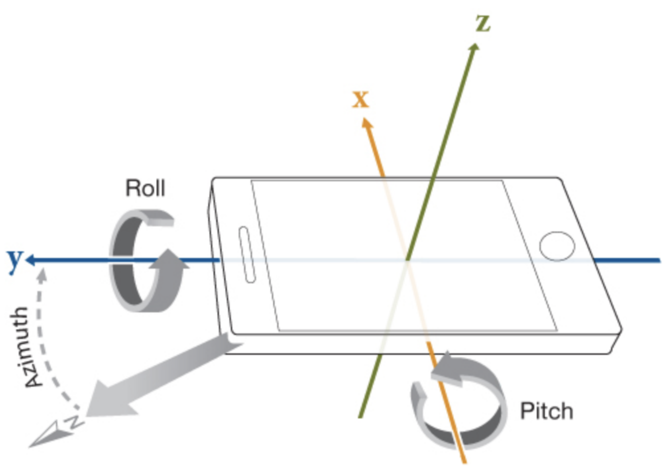


Figure 2.11: Representation of the Roll, Pitch and Azimuth orientation parameters. Taken from [**mathworks**].

Calibration

The "Calibration" has been designed to enhance the precision on the timestamps. Indeed, there is a difference between the moment when the packet is received at the antenna of the DWM1000 and the moment of its detection, which corresponds to the timestamp. The same phenomena appears when the UWB transceiver transmits a packet. Those errors on the timestamps are called the transmit/receive antenna delay and must be configured to match the actual antenna delay.

Expliquer l'origine des delays

2.3.5 Precision obtained

The precision obtained with this locating system depends on several factors, some of which being the location of the anchors in the room, the distance of the tag to those anchors, the clutter of the room, etc...

Nonetheless, a statistical study has been conducted to assess the performances of the locating system. The tag has been placed at several location that can be observed in Fig. 2.12.

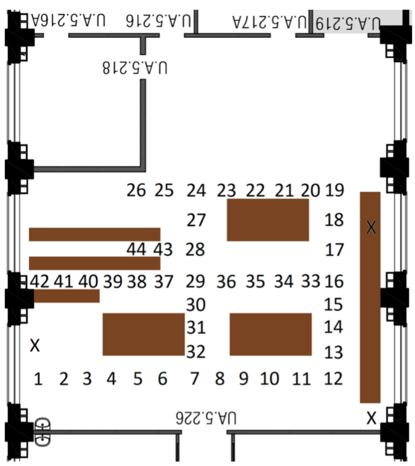


Figure 2.12: Locations corresponding to the measures represented in Fig. 2.13. Taken from [14].

For each location, the measurement was repeated a hundred times. The error remains below 45 cm for 80% of the measurements but reaches up to 85 cm in the worst case. Such deviations can be explained by many different factors, among which the non-trivial geometry of the room, the walls that are not parallel, the presence of windows, ... [14]. The extremity of the vertical segment represent the minimal and maximal errors for each locations.

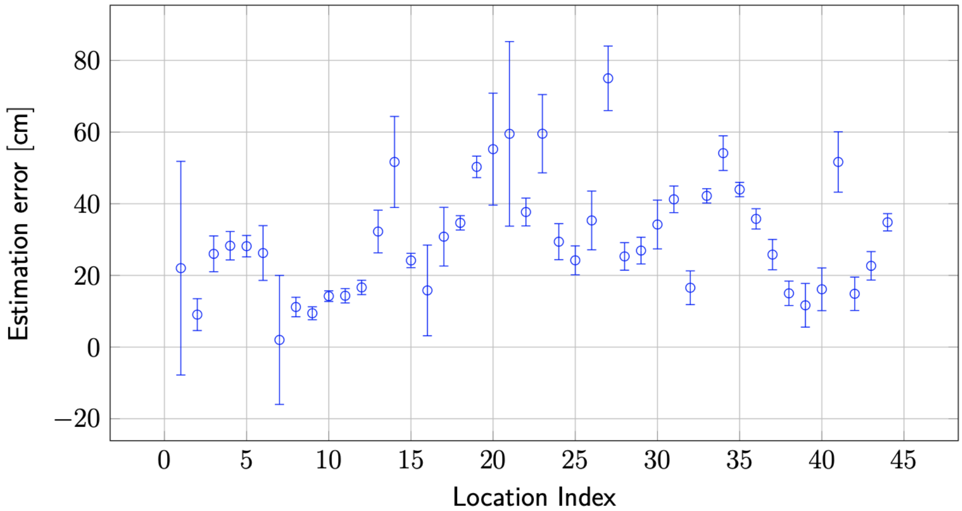


Figure 2.13: Error on the estimated location in function of the position of the tag. Taken from [14].

To achieve a better localization, anchors can be added. It has the advantage to add an equation to the system shown in eq. 2.5, which, if the new anchor gives some coherent measure, improves the estimation of the localization. The drawback is that it will slow done the actualization of the position, since the tag has to communicate with more anchors. Such an algorithm dealing with four anchors is presented in [13].

2.4 Multi-path aided locating system

Rajouter un fil rouge du fonctionnement

2.4.1 Channel Impulse Response

The Channel Impulse Response (CIR) has already been mentioned in the section 2.1. The CIR is not only used in telecommunications systems but also in control theory for example, to characterize the behaviour of a system[7] [12]. As the name states, the goal of this CIR is to characterize the reaction of a system to a stimulus in the form of a pulse.

An example is displayed on the Fig. 2.14. A ray tracing has been performed on the left image. The LoS rays and their first reflection can be observed. If a Dirac pulse is sent from the transmitter (TX) to the receiver (RX), since the propagation time will depend on the distance travelled, different peaks will appear, each corresponding to a different ray.

[7] In control theory, the transfer function which corresponds to the step response is commonly used.

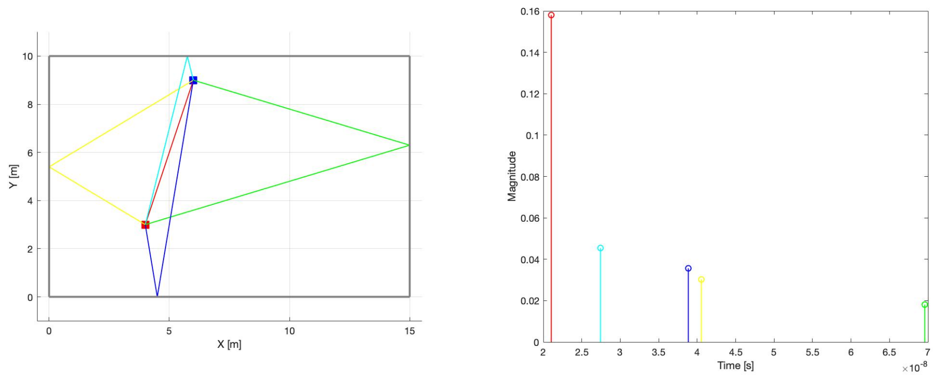


Figure 2.14: (Left) Direct and simple reflected rays between an anchor $(4, 3)$ and a tag $(6, 9)$. (Right) Theoretical CIR associated to each ray displayed in the map.

As one can see, those different arrival times for each ray correspond to the different peaks on the right figure of the Fig. 2.14. The heights of the peaks depend on the attenuation due to the reflections on the walls. The equations used to compute those attenuations are detailed in the section $ref\{SectionNotWrittenYet\}$.

2.4.2 Virtual Anchors

The concept of Virtual Anchor (VA) has been introduced in [17]. While the anchors are some actual devices, the VA are not physically implemented. In order to create them, one needs to know the location of an anchor in a room as well as the exact geometry of the room. In Fig. 2.15, the VAs have been created using the method of images. To find the location of the VA 1, the left wall act as the symmetry axis for an axial symmetry. From this, it is possible to extend this method to two reflections, two axial symmetries on two different walls would be needed in that case.

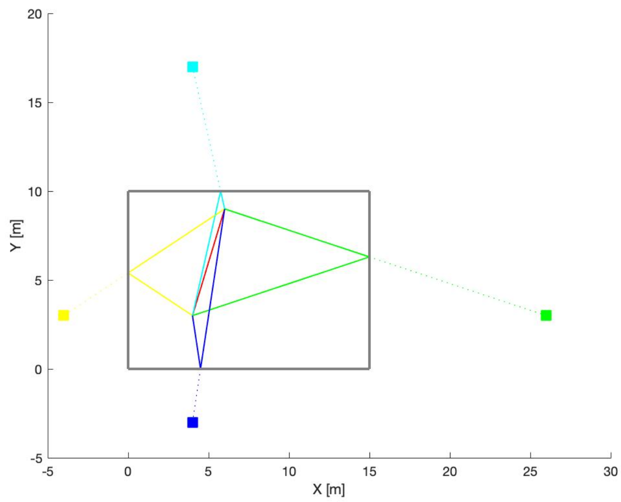


Figure 2.15: VAs associated with its corresponding ray

Using the theoretical CIR from Fig. 2.14, each peak which is associated to a reflected ray can be considered as being the LoS ray from a VA of Fig. 2.15. The colors have been kept for the sake of clarity. The advantages of the methods of images is that the travelled distance by each ray as well as the angular distribution is kept unchanged.

2.4.3 Locating system

Based on the concept of VAs and CIR, several methods are proposed to find the localization of a tag in a room of known geometry using only one anchor.

In [18], the localization of a moving tag is proposed. This method uses the MPC detected as well as the history of the position of the tag which allows to perform a cross-correlation with the newly detected position, therefore maximizing the coherence of the behaviour.

In [11], the Cooperative Multipath-Assisted Indoor Navigation and Tracking (Co-MINT) uses other tags in the room as anchor to perform the location. Each tag, acting as a transceiver and a receiver, communicates with the other tags to compute its distance relatively to those tag. An algorithm is presented in this paper that combines those different informations to achieve the localization.

A refaire d'ici
In [15], a method providing the tag location without using a history of the previous locations is proposed. Using the CIR, several pulses are detected, being equivalent to the ToF computed in the multiple anchor locating system previously presented. Nonetheless, there is a significant difference between both cases. In this method, each ToF is not associated with an anchor (virtual or not). Without knowing the position of the tag, one can not attribute each peak to an anchor[8].

Due to this difference, the system 2.5 needs to be solved multiple times. Depending on the number of peaks detected, permutations between each VA needs to be done to test all the possibilities. The objective is still to find the p_0 value that minimize the equation.

[8] An exception is done for the first peak, since the LoS that originates from the physical anchor is assumed to be the shortest path.

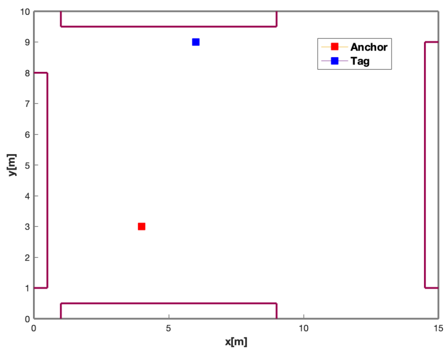
Chapter 3

Algorithms

This chapter present the two locating method implemented and investigated in the simulation. In order to perform the localization, as seen in 2.4, the CIR obtained needs to be treated in order to extract some usefull data. The first section presents such methods. Then, using the extracted peaks, two different localization algorithm are presented, a hard one based on the trilateration and a soft one based on the comparison of theoretical and realistic CIR.

3.1 Peaks extraction

A CIR as presented in 2.14 unfortunately does not exist in the real world, at least, not using the material presented in section 2.3. First, extra peaks will appears, due to the double, triple, etc... reflection on the walls, on the furnitures of the room. Some peaks caused by diffraction may also appears. People walking or standing into the room will also modify the channel and so the CIR. Second, as previously stated in 2.3.1, the bandwidth of the DWM1000 is not infinite, occasioning a spatial extension of the different peaks. The evolution from a theoretical "perfect" case to a "real" one can be observed in Fig. 3.1.



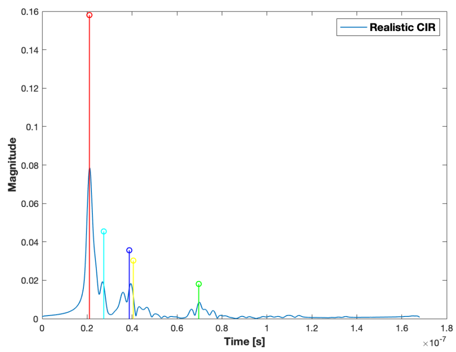


Figure 3.1: (Left) Room used to generate the realistic CIR using up to three reflected rays. (Right) Superposition of the CIR in Fig. 2.3 and the realistic CIR.

The simulation used to generate the realistic CIR is described in chapter 4. As one can observe, the peaks that originates from the theoretical case matches peaks in the realistic one. But one can also observe that some new peaks arises in the realistic ones, such as the ones around $0.4 * 10^{-7}s$.

The goal of the peak extraction function is, as the name states, to extract the peaks matching the needs of the locating algorithm. As it will be seen in section 3.2, 3.3, the two methods do not achieve their objective in the same way, the optimal peaks will be slightly different. Hence the two different peaks extraction methods presented beneath.

3.1.1 Soft case

For the soft locating system, the algorithm of the peak extraction is detailed in algo. 1. The algorithm takes as input a vector made of tuples, respectively the time and the amplitude of each point of the CIR. The `n` parameter states the number of peaks that the algorithm needs to extract.

Algorithm 1: Peaks Extraction - Soft case

Inputs:

$\mathtt{CIR} = \{\mathtt{cir}\} = \{(\mathtt{t_i}, \mathtt{A_i}) \text{ s.t. } \mathtt{i} \in \mathbb{N}_0\} \in \mathbb{R} \times \mathbb{R},\ \mathtt{n} \in \mathbb{N}_0$;

Initialize:

$\mathtt{Peaks} \longleftarrow \{(\mathtt{t_i}, \mathtt{A_i}) \text{ s.t. } \mathtt{i} \in \{1, ..., n\}\}$;

$\mathtt{ratio} \longleftarrow 100$;

$\mathtt{i} \longleftarrow 1$;

$\mathtt{CIR_{max}} \longleftarrow$ `local_max(CIR)`;

$\mathtt{CIR_{max}} \longleftarrow$ `Order(CIR`$_{\mathtt{max}}$`, 'Descend', 'Prominence based')`;

for *{cir} in CIR_{max}* **do**

    if *prom({cir})* $>$ *max(CIR)/ratio* **then**

        `Peaks[i]` $\longleftarrow$ `{cir}`;

        $\mathtt{i} \longleftarrow \mathtt{i} + 1$;

        if $i > n$ **then**

            `break`;

Output:

`Peaks`;

During the initialization phase, first, the output `Peaks` is defined, this variable will be used to store the different peaks extracted. The variable `i` is the counter of the number of peaks found and stored into the `Peaks` vector and `ratio` is used to fix a limit to the relative size of the peaks being take[1]. The local maximum of the `CIR` are then saved in $\mathtt{CIR_{max}}$ and ordered in the decreasing order based on the prominence of each peak. The definition of the prominence is reminded in Fig. 3.2. Since the prominence depends on the local minima between two peaks, an higher amplitude does not always imply an higher prominence.

[1]This value of 100 is arbitrary, it avoids the program to take really smalls peaks produced by the noise.

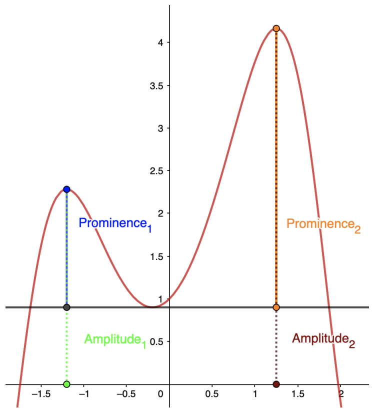


Figure 3.2: Difference between the prominence of a peak and its amplitude

The main loop then get the `n` first values of $\mathtt{CIR}_{\mathtt{max}}$ and store it into `Peaks`. The `If` condition is, as said before, used to removed the small fluctuation in the CIR due to the background, only keeping the most important peaks. If there is less than `n` peaks that satisfy the conditions, the algorithm returns as peaks as possible while still respecting the pre-established conditions.

3.1.2 Hard case

For the hard locating system, the same methodology is applied. The algorithm 2 slightly changes at several different places. First, the input is only `CIR`, the algorithm always returns three peaks in `Peaks`. Second, the ordering of $\mathtt{CIR}_{\mathtt{max}}$ is made based on the amplitude of the peaks. Finally, the amplitude, not the prominence, of the $\{\mathtt{cir}\}$ is compared to `max(CIR)/ratio`.

Algorithm 2: Peaks Extraction - Hard case

Inputs:

$\mathtt{CIR} = \{\mathtt{cir}\} = \{(\mathtt{t_i}, \mathtt{A_i}) \text{ s.t. } \mathtt{i} \in \mathbb{N}_0\} \in \mathbb{R} \times \mathbb{R}$;

Initialize:

$\mathtt{Peaks} \longleftarrow \{(\mathtt{t_i}, \mathtt{A_i}) \text{ s.t. } \mathtt{i} \in \{1, 2, 3\}\}$;
$\mathtt{ratio} \longleftarrow 100$;
$\mathtt{i} \longleftarrow 1$;
$\mathtt{CIR}_{\mathtt{max}} \longleftarrow \mathtt{local\_max}(\mathtt{CIR})$;
$\mathtt{CIR}_{\mathtt{max}} \longleftarrow$ `Order(`$\mathtt{CIR}_{\mathtt{max}}$`, 'Descend', 'Amplitude based')`;

for *{cir} in* CIR_{max} **do**
  if *amp({cir})* $>$ *max(CIR)/ratio* **then**
    `Peaks[i]` $\longleftarrow$ `{cir}`;
    $\mathtt{i} \longleftarrow \mathtt{i} + 1$;
    if $i > 3$ **then**
      `break`;

Output:

`Peaks`;

3.2 Hard localization algorithm

This locating system is based on the idea of trilateration and tries to mimic it. Using three peaks, it tries to match those with the anchor and two VAs to find an intersection point as in the Fig. 2.5. Those three peaks are extracted with the algorithm 2 from the received CIR at the tag. First, the number of systems that needs to be solved are presented, then a systematic method to solve them is proposed.

3.2.1 Virtual antennas combination

The hypothesis that the greatest peak, which is usually the first one, corresponds to the one from the LoS ray is made, this implies that there always exist a LoS in those rooms, meaning that this does not suffer of attenuation. Concerning the second and third peak, since each one can not be surely associated with a VA, the only available solution is to try every combination of VAs. The order being important[2], it is $P_4^2 = \frac{4!}{2!} = 12$ different systems to solve. This computation has been made for a room with a simple geometry, a rectangular, four walls room in this case, of course with a more complex geometry, the number of possible combination would increase.

On Fig. 3.3, a possible problem is shown. The CIR shown on the left side is obtained by computing only the LoS and the first reflections onto the different walls. In theory, five different peaks should be seen, but only four appears. In this particular case, the second peak is formed by the peaks from the blue VA and the green VA, which will be undistinguishable.

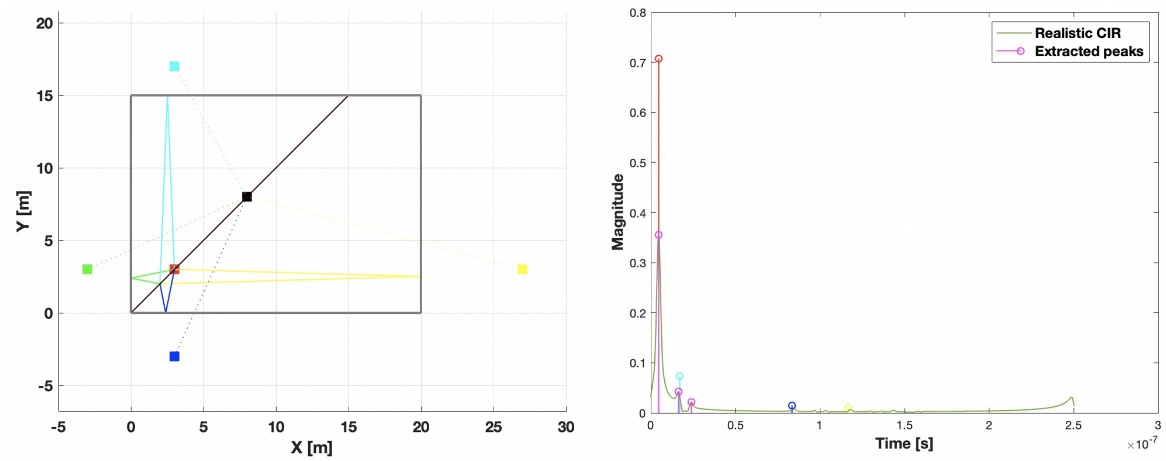


Figure 3.3: (Left) Room with the anchor in red and the tag in black. (Right) CIR corresponding to the room in the left image, the color of the peaks match the color of the associated VAs. The magenta peaks corresponds to the ones extracted using the algo. 2 from the realistic CIR in green.

A problem that may arise in such situation is that the peaks taken from the CIR could not correspond to some actual theoretical peaks[3]. Such an example can be seen on the right image in Fig. 3.3 where the third magenta peak does not correspond to any theoretical peak. To overcomes this problem, the proposed solution is to consider the cases peaks are mingled. Hence to the twelve possibles combinations discussed , one would have to consider that the second magenta peak originate from two different VAs, as the cyan peak on the right image of Fig 3.3.

This method has the drawback of requiring much computation since twelve more peaks - VAs needs to be checked, but it will solve some symmetry-related problem. This kind of

[2]Associating the tuple (d_1, d_2) to (va_1, va_2) is not equivalent to associate it to (va_2, va_1).
[3]The one that corresponds to one reflection.

problem mostly occurring when the tag is equidistant from two VAs, like on the brown line on the map, being the axial symmetry between the two bottom and left VAs. Since this problem only occurs in some specifics cases, one should first check the original antenna combination before checking those added one.

Another source of troubles for this algorithm is due to the finite bandwidth of the antennas. As it can be seen on Fig. 3.4, peaks that can be distinguished on the theoretical CIR are mingled in the realistic one. This phenomena can be observed with the blue and the cyan peaks for example, which are so close that there are considered as one in the finite band CIR.

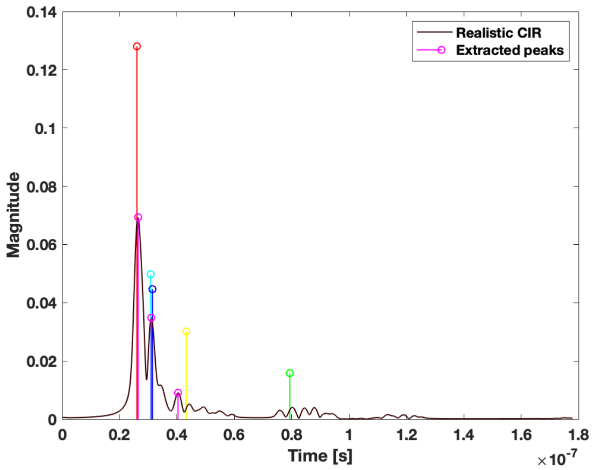


Figure 3.4: Infinite bandwidth CIR generated and the finite bandwidth associated in brown. The extracted peaks are shown in magenta.

To deal with this problem, the solution proposed is the same as the one dealing with the mingled peaks. The precision achieved on the localization in such case would be reduced in those cases. Such examples will be shown in chapter 4. Of course, the third detected peak could also originate from a VA after a simple reflection, this discussion only make sense when it does not.

3.2.2 System solver

Those systems are solved one at a time, starting with the twelve 'basic' ones, pursuing with the particular cases presented above if no suitable solution has been found. Since there is almost zero chances that the system leads into a perfect one point solution, the system 2.4 can not be simply resolved. The three equations are solved two by two, giving six real or complex solutions, if the six are real, this would correspond to U_{12}, U_{13}, U_{23} and the Tag 3 times in Fig. 2.5. From this point, the algorithm first exclude the solutions lying outside of the room, the solutions having a too big imaginary part are also discarded[4].

Using the remaining solutions, the algorithm needs to check if those can be combined to retrieve a suitable solution. A solution is considered as suitable if the algorithm can find three solutions being relatively close to each other. This notion of "relatively close" is subjective and is one of the parameter that one can use to tune the algorithm. If no solution is found using all the combinations, then no location is provided, hence the qualification of "hard".

[4]The notion of "too big" is completely subjective. It appears that when two circle are close to have an intersection, the imaginary part is smaller. NEED A PROOF

3.3 Soft localization algorithm

The locating system presented in this section achieve localization by comparing some theoretical CIR, computed knowing the geometry of the room with the CIR recovered at the tag. In order to make this comparison, the room is sampled, based on the desired precision that one will achieve. In this thesis, the sample will be made using square of one meter side.

For each peak extracted from the CIR, two informations are available, the time of arrival of this peak and its amplitude. One could use the amplitude of the peaks, this would probably work is a lot of cases in the simulation, but this approach has a major drawback, if a peak suffer from to much attenuation in the real case compared to the simulation due to losses coming from the transmission or the reflections on the different materials which is not perfectly simulated, if the antenna emission is not perfectly isotropic in the room plane, etc. Then, even if peaks would likely to be at the same place, nothing can ensure that the amplitude would be proportionally the same.

For those reasons, the comparison will be based on the time-of-arrival of each peak. This method also has some drawbacks, the transition to the finite band does not ensure that the peaks will remains at their exact location. When two peaks mingle, the new peak remaining in finite band is more likely to have an arrival in the middle of those two peaks. Some precautions have been taken to minimize the errors induced by this phenomena.

3.3.1 Space solution reduction

Since the algorithm needs to compare the theoretical CIR at every possible position of the tag, in order to speed-up the program, one could actually reduce the number of location to test.

Using the SDS-TWR, the ToF of the signal between the tag and the anchor can be computed[5]. Based on this ToF, a circle can be traced with the center on this anchor and the radius being the estimated distance deduced from the ToF. In theory, the tag is supposed to be located on this circle, but due to the discretization and errors on the ToF, a margin is taken to get the set of possible locations. This margin resides in the two orange circles, that can be observed on Fig. 3.5.

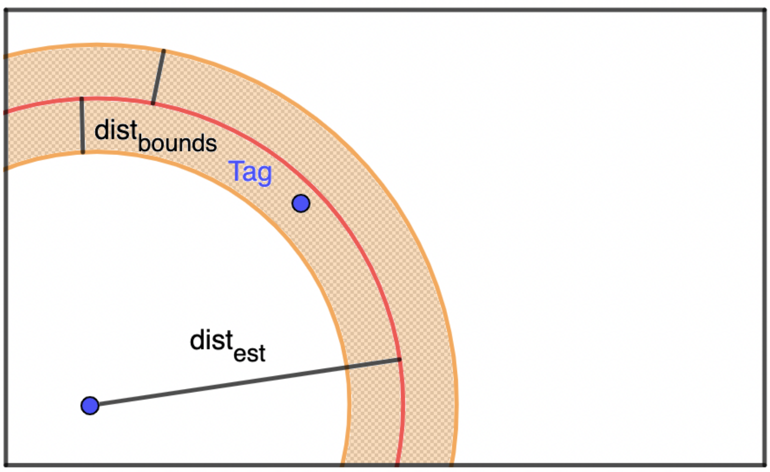


Figure 3.5: Refaire cette image

[5]In the simulation, it will be assumed to be extracted from the CIR

From the position inside those boundaries, a mask matrix representing all the position of the room is filled with ones for the position inside of the orange zone. The other positions are left to zero. Later, this mask is used to reduce the computations since only the values associated with a one will be tested.

3.3.2 Time MSE

For each remaining possible location, a finite band CIR is simulated using only the direct ray and the once reflected rays. Such example can be found in Fig. 3.6, in order to obtain something similar to the realistic case, the CIR has been passed in the finite band. The peaks are then extracted using algo. 1 from this finite band CIR.

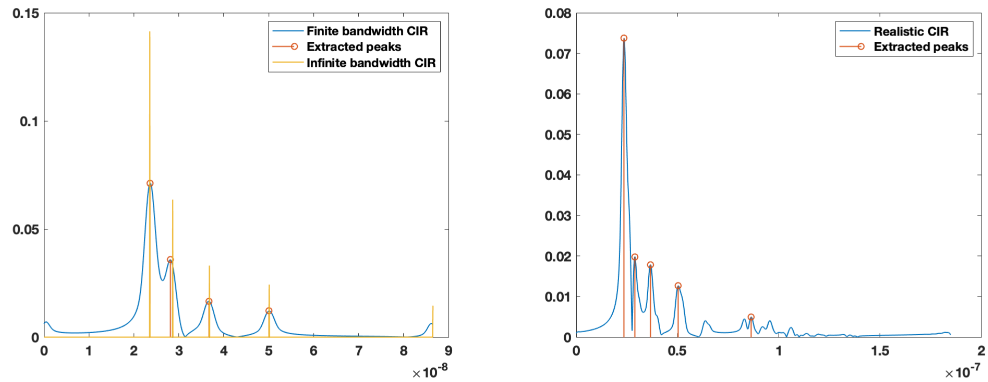


Figure 3.6: (Left) Theoretical CIR obtained using the simulation. The same room has been used as for the right image, without furnitures and using only the direct ray and single reflected rays. (Right) Realistic CIR obtained in a room full of furnitures using the simulation.

The extraction is also done once on the CIR. The using the different peaks extracted, the following problem is solved.

$$\begin{aligned} S(\vec{p_0}) &= \sum_{i}^{N}(\|\text{Peaks}(\vec{p_r})_i - \text{Peaks}(\vec{p_0})_i\|^2) \\ \vec{p_0} &= \underset{\vec{p}}{\text{argmin}}\ S(\vec{p}) \end{aligned} \tag{3.1}$$

The peaks extracted are sorted in the chronological order in the peak extraction. Therefore, the sum goes through the peaks of both CIR. The argument N is limited by the smallest list of peaks given by the algo. 1. While the $\vec{p_0}$ represents all the possible position tested, $\vec{p_r}$ represents the real case. $\text{Peaks}(\vec{p_0})_i$ represents the i-th peak extracted from the theoretical CIR computed by placing the tag at the position $\vec{p_0}$.

This algorithm will have as output a map of the room with the Mean Square Error (MSE) of the time arrival of the peaks computed at each possible location. The locations that were not estimated due to the space solution reduction occurring before the computation of the MSE are colored in black. An example of such output can be seen in Fig. 3.7, this heat-map has been generated using the MSE computed at each location evaluated. The estimator 3.1 will provide only one output, being the one with the lowest MSE. A solution will always be provided by this locating system.

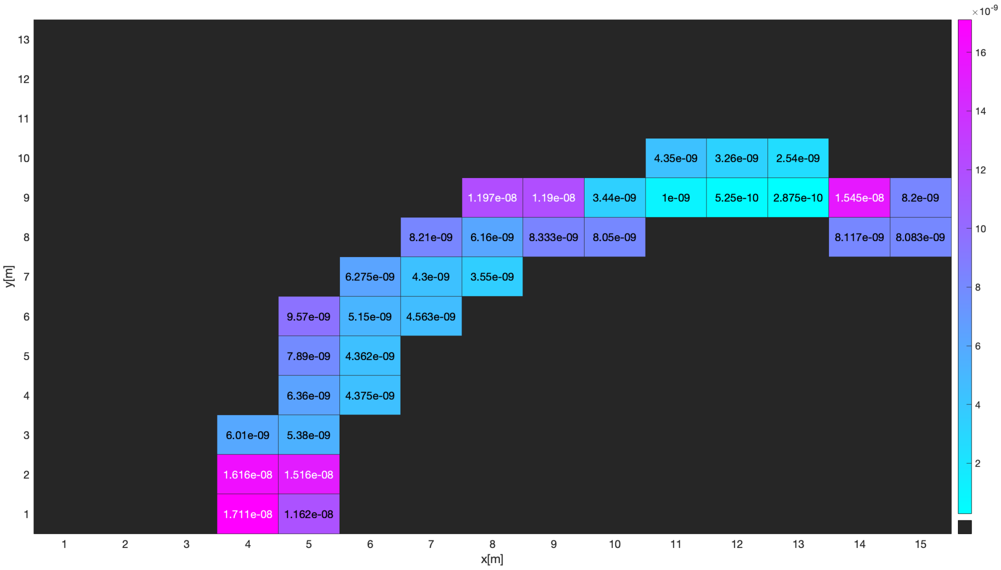


Figure 3.7: Example of an execution of the soft localization algorithm. Anchor at (12,2) and Tag at (13,9).

Chapter 4

Simulations

This chapter presents the simulation. To be continued .

4.1 Creation de la simulation

Bien expliquer la maniere dont les simulations sont générées
Dire que c'est un code orienté objet sous matlab.

4.1.1 Paramètres de la simulations, antennes, murs, etc ?

4.1.2 Configuration file, permet de générer les pièces qu'on veut, mettre les antennes où on veut etc

4.1.3 SImulation de la bande finie

4.1.4 Figures qu'on peut générer et donc les analyses qu'on peut faire

4.2 Resultats pour Soft Simulation

4.3 Résultats pour Hard Simulation

Explain the equation behind the simulation

Chapter 5

Experiences

This chapter A finir

5.1 Tag

Expliquer que le but était de ne pas influencer le fonctionnement des ancres.

5.1.1 DWM1000

Activation des bits nécéssaire, etc ...

5.1.2 PSoC

5.2 Android Application

Stockage de la cir, ajout des boutons, accès à la mémoire dans le téléphone, affichage de la cir dans le téléphone.

Expliquer comment ça à été implémenter

Les bagares avec DWM1000 pour obtenir la CIR, stockage de la CIR dans le telephone, modification du code pour tout coupler. Integration a la navigation de la recuperation des données.

Expliquer le protocole de test également.

Finir sur un mot qui dit que ce n'est pas fini à cause du coco.

Explain the configuration of the DWM1000

Chapter 6

Conclusion

Appendix A

Application view

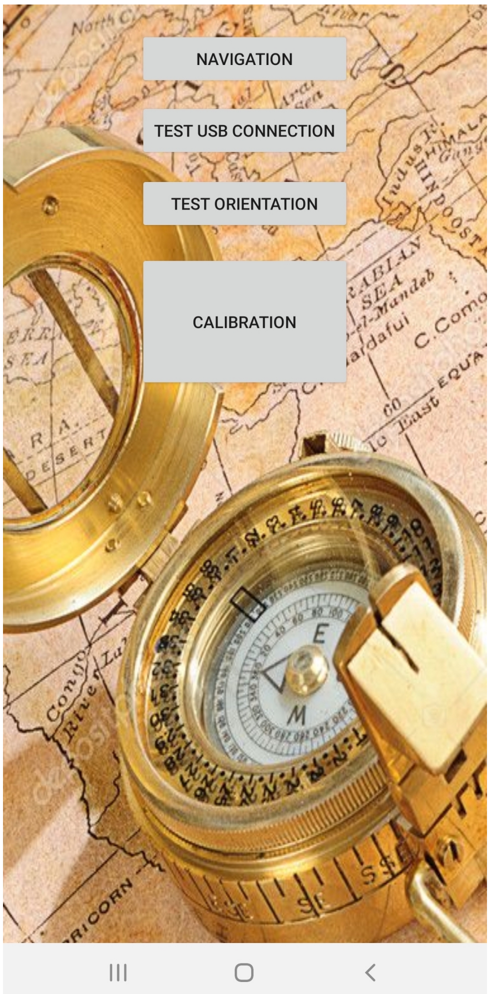


Figure A.1: Main window of the Android application.

Appendix B

State Machines

Bibliography

[1] *Adafruit Feather Huzzah datasheet.* `https://cdn-learn.adafruit.com/downloads/pdf/adafruit-feather-huzzah-esp8266.pdf`. 2020.

[2] *Bus SPI.* `http://projet.eu.org/pedago/sin/term/8-bus_SPI.pdf`.

[3] Rejane Dalce, Thierry Val, and Adrien V Bossche. "Comparison of indoor localization systems based on wireless communications". In: (2011).

[4] *Decawave Certification Guide Europe.* `https://www.decawave.com/wp-content/uploads/2018/10/APR003_Certification-Guide-Europe_v1.1.pdf`.

[5] *Decawave DWM1000 module.* `https://www.decawave.com/product/dwm1000-module/`. Accessed: 05-2020.

[6] *Decawave DWM1000 User Manual.* `https://www.decawave.com/sites/default/files/resources/dw1000_user_manual_2.11.pdf`. 2017.

[7] Jense Defraye. "Determining the position of sporters using ultra-wideband indoor localization". MA thesis. Belgium: UGent, 2017.

[8] David L Donoho and Philip B Stark. "Uncertainty principles and signal recovery". In: *SIAM Journal on Applied Mathematics* 49.3 (1989), pp. 906–931.

[9] *ESP8266 Datasheet.* `https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf`. 2020.

[10] Quentin Fesler. "High-accuracy localization of robotic platforms with ultra-wideband wireless transceivers". MA thesis. Belgium: ULB-VUB, 2018.

[11] Markus Froehle et al. "Cooperative multipath-assisted indoor navigation and tracking (Co-MINT) using UWB signals". In: *2013 IEEE International Conference on Communications Workshops (ICC)*. IEEE. 2013, pp. 16–21.

[12] Emmanuel Garonne. "Course of : Control System Design". Belgium, 2017.

[13] Léon Guyard. "Navigation et localisation en intérieur grâce à l'émission d'ondes à large bande fréquentielle". Belgium, 2019.

[14] Cedric Hannotier. "Indoor localization and navigation using ultra-wideband ranging". MA thesis. Belgium: ULB-VUB, 2019.

[15] Mads H Jespersen et al. "An indoor multipath-assisted single-anchor UWB localization method". In: *2018 IEEE MTT-S International Wireless Symposium (IWS)*. IEEE. 2018, pp. 1–3.

[16] Josef Kulmer et al. "Using DecaWave UWB transceivers for high-accuracy multipath-assisted indoor positioning". In: *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE. 2017, pp. 1239–1245.

[17] P. Meissner, C. Steiner, and K. Witrisal. "UWB positioning with virtual anchors and floor plan information". In: *2010 7th Workshop on Positioning, Navigation and Communication*. 2010, pp. 150–156.

[18] Paul Meissner, Thomas Gigl, and Klaus Witrisal. “UWB sequential Monte Carlo positioning using virtual anchors”. In: *2010 International Conference on Indoor Positioning and Indoor Navigation*. IEEE. 2010, pp. 1–10.

[19] Faranak Nekoogar. *Ultra-Wideband Communications: Fundamentals and Applications*. First. USA: Prentice Hall Press, 2005. ISBN: 0131463268.

[20] ITU-R SM.1755-0. *Characteristics of ultra-wideband technology*. 2006.

[21] Yu Zhou. “An efficient least-squares trilateration algorithm for mobile robot localization”. In: *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2009, pp. 3474–3479.