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... [15]

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 [18]. 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 [19].

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 huge.

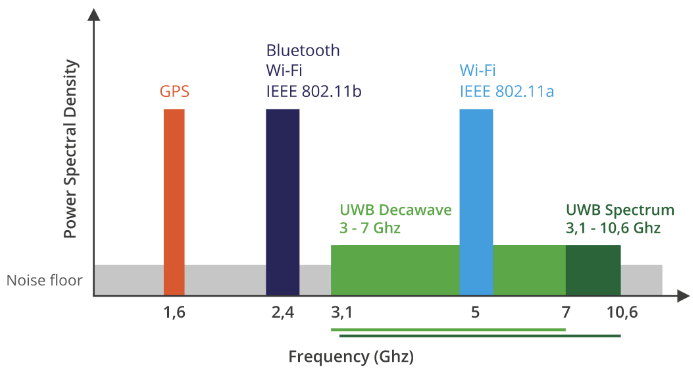

Figure 2.1: UWB spectrum compared to Wi-Fi and other wireless technology. Taken from [19]

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.

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

The Fig. 2.2 shows the theoretical duration of an impulse of the UWB.

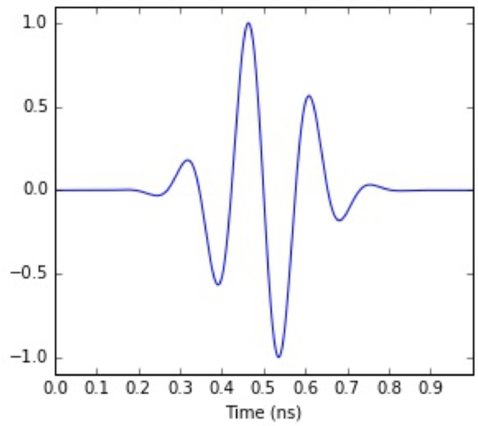

Figure 2.2: Theoritical duration of an UWB pulse. Taken from [7]

An advantage of the UWB is its robustness in regard of the Multipath Channels (MPC). This can be understood by looking at Fig. 2.3, where several peaks can be distinguished, each corresponding either 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 to all the wall of a room.

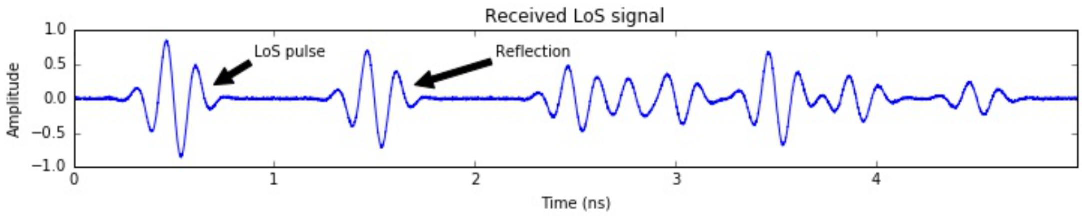

Figure 2.3: Example of an MPC with a Line of Sight (LoS) and a Non Line of Sight (NLoS). Taken from [7]

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 being called in this paper "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 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 [9], [13] alongside a presentation of several approach to determine the relative position of a tag relatively to an anchor.

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 devices, respectively $RDEV_1$ initiating the communication and

‘$RDEV_2$. Each device need to save the Time of Emission (ToE) or Time of Arrival (ToA) of every message. Those time 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 with that t_5 in it should be exchanged.

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

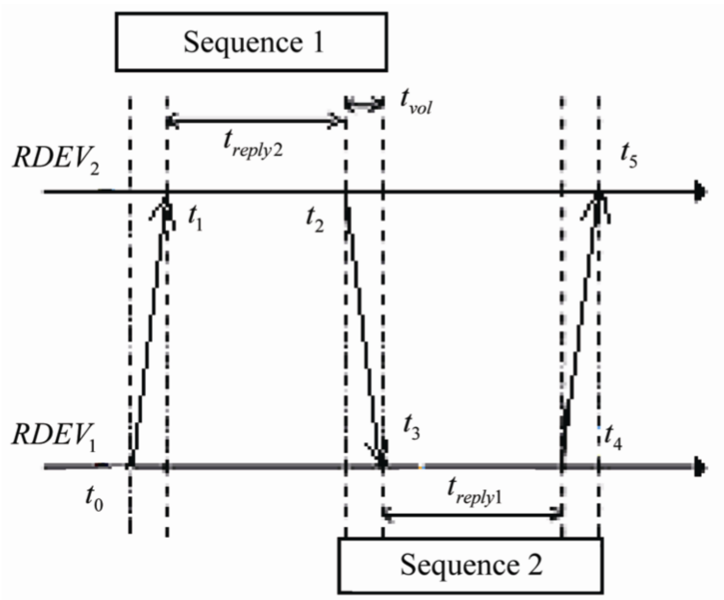

Figure 2.4: Symmetric double-sided two-way ranging. Taken from [3]

Based on those timestamps, the computation of the ToF can be observed in eq. 2.2.

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

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.

$$t_{true} - t_{est} = \frac{1}{4} * (t_{reply2} - t_{reply_1}) * (e_1 - e_2) \tag{2.3}$$

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

2.2.2 Trilateration

Based on the relative localization performed using SDS-TWR, a ToF can be computed. 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 the ToF associated to this anchor. A scheme displaying that solution can be seen on Fig. 2.5.

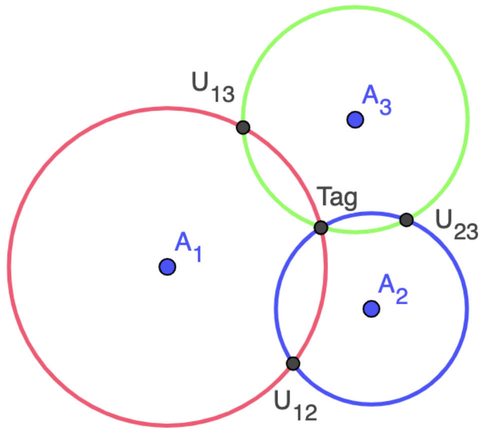

Figure 2.5: Triangulation

As one can deduce, in a two dimensional plan, three anchors are need 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 plan, four anchors would be needed. This actually correspond 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 exactly. There is not a single point as an output. As it stands, the system probably does not have a solution, each equations solved two-by-two would likely gives different answers. To avoid this problem, the trilateration estimator developed in [20] has been used.

$$S(p_0) = \sum_{i}^{N} [(p_i - p_0)^T (p_i - p_0) - d_i^2]^2 \tag{2.5}$$

Where $p_k = (x_k, y_k) \forall x \in 0, ..., N$ in the two dimensional case. N being the number of anchor[1]. The objective of this estimator is to find the value of p_o minimizing the value of $S(p_0)$.

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 [9], [13]. This system is able to retrieve a localization with an error oscillating between twenty and fifty centimetres inside of a building [12].

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

This locating system is composed of fixed antennas[2] made using an ESP8266 as micro-controller and an UWB transceiver being the DWM1000 produced by Decawave[5]. The tag are built using an Android cellphone, a PSoC[3] and also a DWM1000 module.

2.3.1 DWM1000

The DWM1000 is the antenna chosen to operate the wireless communication part, it will be needed for the tag as well as for the different anchors. The configuration of these antenna 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 the most suited configuration to have a low error rate, the best speed of the communication and the lowest power consumption possible. This leads to the following choices[4] :

- 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 $(3.1; 4.8)$GHz than in the frequencies bounds $(6; 9)$GHz[4]. The other channel that is in those more boundaries in the 7^{th} one. The difference being a bandwidth being twice as large[5].

The choices of the bitrate are restricted between 110kbits^{-1}, 850kbits^{-1} or 6800kbits^{-1}. The reasons behind the choice of the bitrate at 6.8Mbits^{-1} are explained in [13].

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

The preamble length is used for the channel estimation, the longer the more accurate. Unfortunately, a longer preamble means more power consumption unused to transmit "real" data and less time to transmit to "real" data. Recommended bitrate in function of the bitrate are proposed in [6].

[2]Called anchors
[3]The exact model is the : CY8C5888LTI-LP097 [12]
[4]A more detailed discussion on the choice of those parameters can be found in [13]
[5]The bandwidth of the 5^{th} one is 499.2MHz while the one from the 7^{th} is 1081.6MHz.

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 transmission speed happening on the MISO and MOSI. Since the SPI allows different slaves for only one master, the Slave Select (SS) is used by the master to select a specific slave to communicate with.

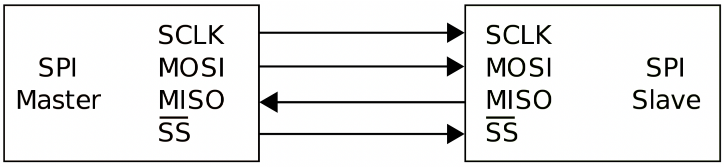

Figure 2.7: SPI Schematic

2.3.2 Anchor

The anchors are fixed antennas composed of a DWM1000 and an ESP8266 [8]. They are placed at known position in the room and are used to compute the ToF between the tag and the anchor using the SDS-TWR explained in section 2.2.1.

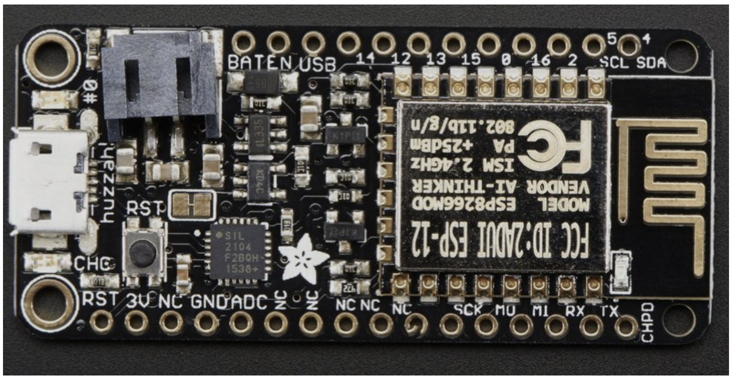

Figure 2.8: ESP8266 mounted on a Feather Huzzah board. Taken from [1]

The micro-controller has been combined with the development board Feather Huzzah from Adafruit [1], the Fig. 2.8 represents this module. This board can be flashed using an USB serie connection, allowing a easy deployment of the code, it also have the advantage to be light and small, an useful feature to deploy several anchors in a room without much cluttering.

2.3.3 Tag

The tag is the object we want to know the localization. It is composed of a DWM1000 antenna, a PSoC[6] 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.

[6]The exact model is the : CY8C5888LTI-LP097 [12]

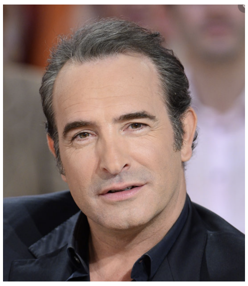

Figure 2.9: PSoC card Photo need to be added

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 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 on Fig. 2.10. Four different buttons can be seen.

Figure 2.10: Main window of the android application

Navigation

The Navigation button opens a map of an environment[7] 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 displayed, computed from the bottom left corner of the map. The used map is shown in Fig. 2.11.

[7]The room UA5.214 in this case, which is one of the electronic lab at the ULB.

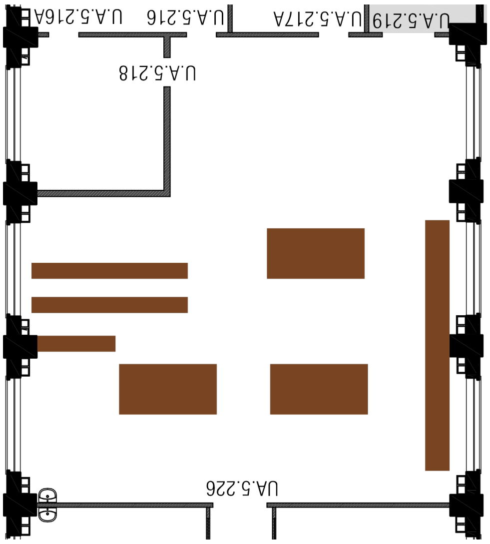

Figure 2.11: Map of the UA5.214

When this function runs, the application enters a loop that continuously request the PSoC to perform a SDS-TWR with each anchors. 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 a (x, y) estimated 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 coming from [13] can be found in make a link to an annex and put the state machine in annex representing the state machine of an anchor and a tag. While the state machine of the anchor only perform 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 being : $0x02034637$. This feature allows to quickly debug the USB communication.

The "Test orientation" has been designed to test the detection of the orientation the cellphone. The three angles necessary to characterize the orientation of the device, respectively the Azimuth, the Pitch and the Roll. The goal of this button was mainly to assess that the recuperation of the orientation of the cellphone was working.

Calibration

The "Calibration" has been designed to enhanced 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 transmit a packet. Those errors on the timestamps are called the transmit/receive antenna delay and must be configured to match the actual antenna delay.

2.3.5 Precision obtained

The precision obtained with this locating system depends on several factors. It depends on 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

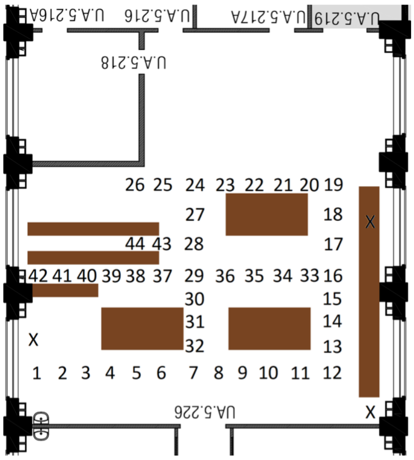

Figure 2.12: Locations corresponding to the measures represented in Fig. 2.13. Taken from [13]

For each location, the measurement was repeated hundred times. The error remains below 45 cm for 80% of the measurements but reaches up to 85 cm in the worse case. Such deviations can be explained by the geometry of the room, which is not trivial, the wall are not parallel, there is window in it, etc...[13].

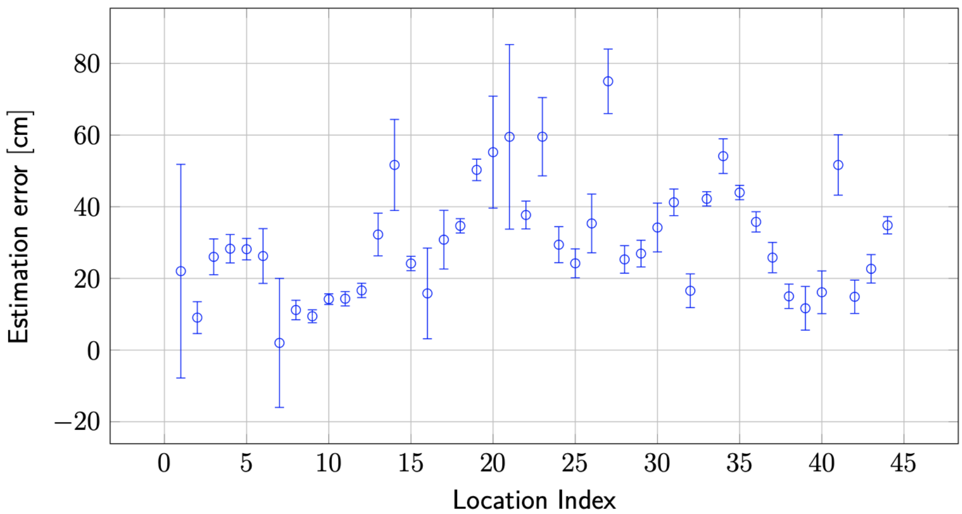

Figure 2.13: Error on the estimated location in function of the position of the tag. Taken from [13]

To achieve a better locating, 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 it has to communicate with more anchor. Such algorithm dealing with four anchor is presented in [12].

2.4 Multi-path aided locating system

2.4.1 Channel Impulse Response

The Channel Impulse Response (CIR) has already been briefly introduced 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[8] [11]. 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 the way that have been reflected once 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.

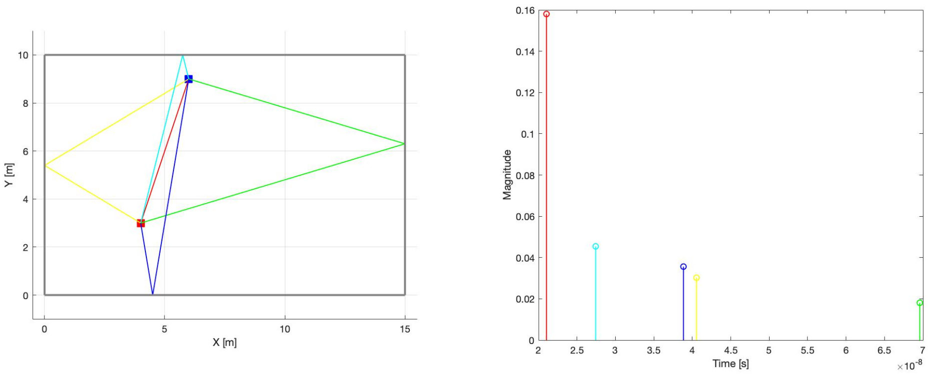

Figure 2.14

As one can see, those different arrival time for each ray correspond to the different peaks on the right figure of the Fig. 2.14. The height of the peak depend on the attenuation suffered by the emitted peak 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 [16]. While the anchors are some touchable devices, the VA are not physically implemented. In order to create those, one need to know the location of an anchor in a room as well as the exact geometry of the room. On Fig. 2.15, the Virtual Anchors (VAs) have been created using the method of images. To find the location of the VA 1, the left wall act as the plane 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.

[8]In automatic, the transfer function which correspond to the step response is commonly used.

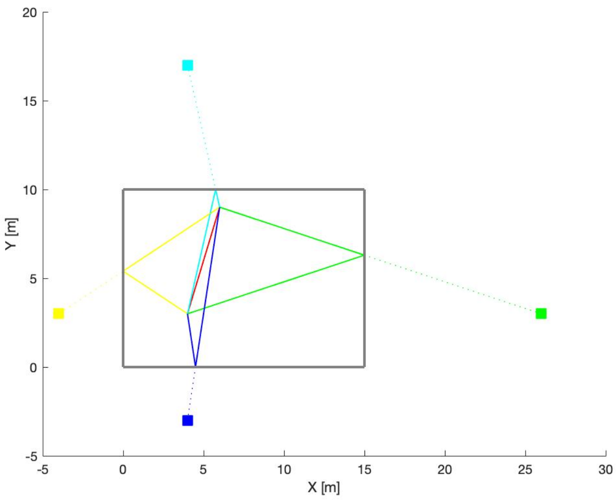

Figure 2.15: Antenne VA example

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 [17], 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. The history of the locations of the tag are cross-correlated with the new detected position, the objective being to have some coherent behaviour.

In [10], the Cooperative Multipath-Assisted Indoor Navigation and Tracking (Co-MINT) proposed uses other tags in the room as anchor to perform the location in a room. 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 perform the localization.

In [14], a localization method providing the tag location without using a history of the previous locations is proposed. Using the CIR, several pulses are detected, each 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[9].

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.

[9] 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

3.1 General algorithms

3.1.1 Peaks extraction

In order to use the CIR recovered either in the simulation or from the experimental set-up, those need to be processed. The major objective is to retrieve the different peaks that originates from the physical anchor and the VAs. Unfortunately, the CIR obtained is not as simple as shown in Fig. 2.3. In order to obtain the same results, one would need some antennas with an infinite bandwidth[1], a "clean" room and to receive to the receiver side only the rays coming from the LoS or reflected once.

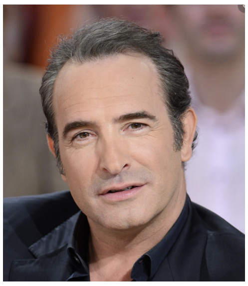

Figure 3.1: CIR example.

The CIR shown for Fig. **??** has been generated in a rectangular cluttered room of 15 over 10m, filled with furnitures. The selected bandwidth is the same as for the experimental set-up : 499.2MHz. The positions of the tag have been kept the same as in Fig. 2.14. As one can observe on Fig. **??**, getting the different peaks is not as trivial.

From the local maxima of this CIR, only the most significant peaks should be extracted, since they likely are associated with the LoS or first reflection. The parameter chosen to reflects the significance of a local maxima is its prominence.

The algorithm performs a search through the local maxima of the CIR. First the global maxima is considered as being the first peak corresponding to the LoS. Hence only the local maxima arriving after are considered. Then, the rest of the local maxima are selected from the biggest to the shortest up to the point where n peaks are chosen. All of those peaks are finally sorted based on the time associated with those peaks.

The algorithm is formalized right beneath :

[1]In order to obtain Dirac peaks, one need an infinite bandwidth since the $\delta(t) \xrightarrow{\mathscr{F}} 1$

Algorithm 1: Peaks Extraction

Data: CIR such that $CIR(i)$ is a tuple $[time, val]$, $n \in \mathbb{N}$ the number of peaks requested

Result: $Peaks$, ordered list of n tuples.

Initialization;
$Peaks(1) \longleftarrow max(CIR)$;
$CIR \longleftarrow CIR(max : end)$;
$i \longleftarrow 2$;
$r \longleftarrow 5$;
while $i < n$ *and* $r < r_{max}$ **do**
  for el *in* CIR **do**
    if $el > max(CIR)/r$ **then**
      $Peaks(i) \longleftarrow el$;
      $i \longleftarrow i + 1$;
  $r \longleftarrow r + 1$
$Order(Peaks)$

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 associate those with the anchor and two virtual anchors to find an intersection point as in the Fig. 2.5. Those three peaks are extracted with the algorithm 1 from the received CIR at the tag. As brieftly explained is section 2.4.3, there is two main difference with the theoretical case.

1. The peaks are not associated with a specific anchor
2. Peaks being to close in the infinite bandwidth response could be combined

The first problem has a kind of direct solution, which is trying all the different of VAs possible. This is quite straightforward in a simple square or rectangular, since only four VA exist, leading to six different combinations : $C(4, 2) = \frac{4!}{(4-2)!2!} = 6$. Of course, with more complex geometry, the number of permutation to test just keep growing.

The second problem can be seen on Fig. **??**. Indeed, the two first peaks are in this case too close to each other in time, hence on the right graph, only one peak can be distinguished.

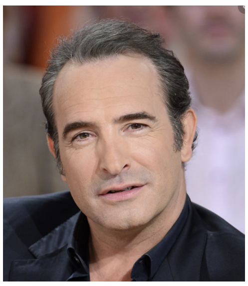

Figure 3.2: Comparison between an infinite BW CIR and a finite one

For each possible combination of anchors, the algorithm will try to solve the system of equations two times. The VAs being permuted for each combination since the origin of a specific reflected peak is not known. Based on each of those systems, three different subsystems formed by two out of the three equations are solved for (x_0, y_0). Each of those subsystem is supposed to give 0, 1 or 2 solutions depending on the intersection type between the two circles. The solutions from the three subsystems are regrouped in an array and compared. If three out of the six positions indicates the same area, one can assume that the position as been well retrieved.

In this case where the two VAs are selected in the right order,

- Mettre le schéma de l'antenne qui est dans un coin, et des 4 zones qui se dessinent - Ne pas oublier le cas où l'on a 6 zones en fait (dans les pièces rectangulaires) - Expliquer qu'on va tester toutes les possibilités 2 à 2, mais qu'on peut éliminer certaines avec l'algo speed 2. - Expliquer que si ça ne marche pas 2à2, on test les cas où les rayons qui arrivent sont confondus.

Les problèmes de cet algo proviennent en partie des symmétries, mais on verra ça dans la simulation, ne pas encore en parler mtn

3.3 Soft localization algorithm

3.3.1 CIR MSE

Pour la forme de la CIR, bien expliquer pourquoi ça ne marche pas vraiment. Même en atténuant les autres pics de la même manière que le pic principal. Parce que la réponse qu'on reçoit peut avoir un LoS légèrement plus obstrué que le reste, ou vice versa. Ca marche bien dans la simulation, mais ça sera probablement moins efficace IRL.

3.3.2 Time MSE

Méthode préférée à CIR

3.4 Speed-Up Algorithms

3.4.1 Speed-Up 1

The aim of this algorithm is to speed up the locating process by reducing the number of needed computations. To achieve this, the CIR, that needs to be computed at each location, is only computed in a reduced set of possible position for the tag.

Using the SDS-TWR, the ToF of the signal between the tag and the anchor can be computed[2]. 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. 3.3.

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

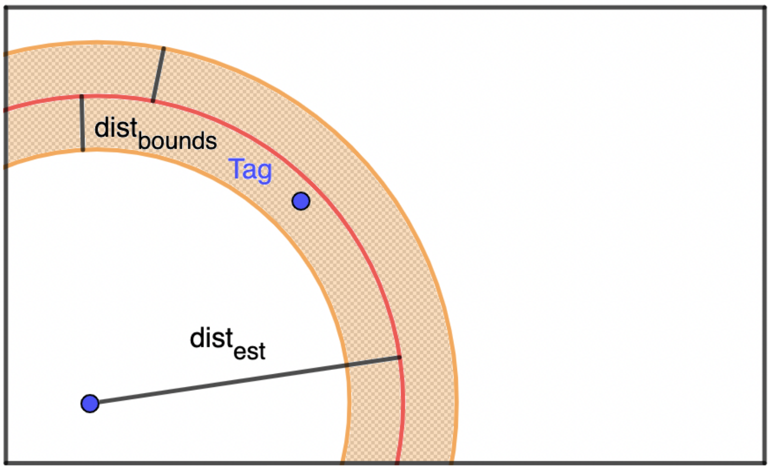

Figure 3.3: Example of algo 1

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.

3.4.2 Speed-Up 2

Choix des ancres en utilsant l'algorithme numéro 1

Chapter 4

Simulations

4.1 Creation de la simulation

Bien expliquer la maniere dont les simulations sont générées

4.2 Résultats de la simulation

Qu'est ce qu'on en obtient, qu'est ce qu'on peut en déduire
1 ancre
2 ancres ?
Explain the equation behind the simulation

Chapter 5

Experiences

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

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

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

[10] 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.

[11] Emmanuel Garonne. “Course of : Control System Design”. Belgium, 2017.

[12] Léon Guyard. “Navigation et localisation en intérieur grâce à l’émission d’ondes à large bande fréquentielle”. Belgium, 2019.

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

[14] 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.

[15] 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.

[16] 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.

[17] 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.

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

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

[20] 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.