

Drone aided anchor calibration for UWB indoor positioning

Wouter Dedeurwaerder

Supervisors: Prof. dr. ir. Heidi Steendam, Prof. dr. ir. Clara-Mihaela Ionescu
Counsellor: Stef Vandermeeren

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Electromechanical Engineering

Department of Telecommunications and Information Processing
Chair: Prof. dr. ir. Herwig Bruneel

Department of Electrical Energy, Metals, Mechanical Constructions & Systems
Chair: Prof. dr. ir. Luc Dupré

Faculty of Engineering and Architecture
Academic year 2017-2018



Drone aided anchor calibration for UWB indoor positioning

Wouter Dedeurwaerder

Supervisors: Prof. dr. ir. Heidi Steendam, Prof. dr. ir. Clara-Mihaela Ionescu
Counsellor: Stef Vandermeeren

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Electromechanical Engineering

Department of Telecommunications and Information Processing
Chair: Prof. dr. ir. Herwig Bruneel

Department of Electrical Energy, Metals, Mechanical Constructions & Systems
Chair: Prof. dr. ir. Luc Dupré

Faculty of Engineering and Architecture
Academic year 2017-2018



Permission for Usage

The authors give permission to make this master dissertation available for consultation and to copy parts of this master dissertation for personal use. In the case of any other use, the limitations of the copyright have to be respected, in particular with regard to the obligation to state expressly the source when quoting results from this master dissertation.

De auteurs geven de toelating deze scriptie voor consultatie beschikbaar te stellen en delen van de scriptie te kopiëren voor persoonlijk gebruik. Elk ander gebruik valt onder de beperkingen van het auteursrecht, in het bijzonder met betrekking tot de verplichting de bron uitdrukkelijk te vermelden bij het aanhalen van resultaten uit deze scriptie.

Wouter Dedeurwaerder & Levi Vuylsteke, June 2018

Foreword

This master's dissertation counts as the crowning of our five years of engineering studies. On the one hand, it has taught us many things about performing scientific measurements, collecting, interpreting and processing data, stating conclusions and writing a thorough report. On the other hand, not to be underestimated, since we are doing this thesis by two, it has learned us a lot of interpersonal skills. One aspect is communicating but also coping with different insights and views has to be dealt with. However, this thesis would not have been possible without the help of many people involved. Therefore, we want to acknowledge the work of the following people involved.

First of all, we would like to thank our promotores Prof. Dr. Ir. Heidi Steendam and Prof. Dr. Ir. Clara-Mihaela Ionescu for the opportunity to have this topic as thesis and for the guidance. Next, words of gratitude are in place for our counsellor Ir. Stef Vandermeeren for the advice, insights and fast feedback throughout the thesis. Furthermore, we want to thank the technical staff Davy Moreels and Philippe Serbruyns for helping us out with the drone and casings for the hardware. For providing us with extra information and scripts to gather more information from the Pozyx devices, we would like to acknowledge the people involved with this start-up.

Next to the aforementioned people associated with the contents of the thesis, other help was provided from other people in terms of support. Therefore, last but not least, we want to thank our friends and family who stood by us in difficult times.

Wouter Dedeurwaerder & Levi Vuylsteke, June 2018

Drone aided anchor calibration for UWB indoor positioning

Wouter Dedeurwaerder, Levi Vuylsteke

Master's dissertation submitted in order to obtain the academic degree of

MASTER OF SCIENCE IN ELECTROMECHANICAL ENGINEERING

Supervisors: Prof. dr. ir. Heidi Steendam, Prof. dr. ir. Clara-Mihaela Ionescu

Department of Telecommunications and Information Processing

Department of Electric Energy, Metals, Mechanical Constructions & Systems Academic year

2017-2018

Faculty of Engineering and Architecture, Ghent University

Abstract

In recent years, indoor positioning systems (IPS) have gained more attention, especially when linked to a global positioning system (GPS). This link facilitates the next challenge IPS faces, namely a smooth GPS/IPS integration. For applications requiring highly accurate results, Ultra Wideband technology is considered as one of the most promising choices. It uses a high bandwidth to achieve short pulses with an accurate timing. Coupling the anchors of this indoor positioning system to its global coordinates turns out to be a difficult task.

To solve this, an automated drone procedure for the calibration of the global coordinates of Ultra Wideband anchors is proposed. Here, the drone uses both its known global position and the estimated distance with the UWB anchors to then calibrate the position of the latter. In order to achieve this, corrections to the measurements are needed, since a bias occurs on the measured distance. As this bias depends on the measurements being in Line-of-Sight (LOS) or Non-Line-of-Sight (NLOS), this should be identified. Therefore, NLOS identification features are proposed, which are then used in a machine learning classification approach, where further the bias is finally corrected in a machine learning regression approach. Initial simulations of the proposed drone procedure resulted in final calibrated anchor position errors $< 10cm$.

Keywords

UWB, calibration, drone, positioning, IPS

Drone aided anchor calibration for UWB indoor positioning

Wouter Dedeurwaerder, Levi Vuylsteke

Supervisor(s): Prof. dr. ir. Heidi Steendam, Prof. dr. ir. Clara-Mihaela Ionescu, Ir. Stef Vandermeeren

Abstract—In recent years, indoor positioning systems (IPS) have gained more attention, especially when linked to a global positioning system (GPS). This link facilitates the next challenge IPS faces, namely a smooth GPS/IPS integration. For applications requiring highly accurate results, Ultra Wideband technology is considered as one of the most promising choices. It uses a high bandwidth to achieve short pulses with an accurate timing. Coupling the anchors of this indoor positioning system to its global coordinates turns out to be a difficult task. To solve this, an automated drone procedure for the calibration of the global coordinates of Ultra Wideband anchors is proposed. Here, the drone uses both its known global position and the estimated distance with the UWB anchors to then calibrate the position of the latter. In order to achieve this, corrections to the measurements are needed, since a bias occurs on the measured distance. As this bias depends on the measurements being in Line-of-Sight (LOS) or Non-Line-of-Sight (NLOS), this should be identified. Therefore, NLOS identification features are proposed, which are then used in a machine learning classification approach, where further the bias is finally corrected in a machine learning regression approach. Initial simulations of the proposed drone procedure resulted in final calibrated anchor position errors $< 10\text{cm}$.

Keywords—UWB, calibration, drone, positioning, IPS

I. INTRODUCTION

THE well known problem with GPS signals is that they do not perform well in indoor environments, due to the additional signal loss incurred by the walls of the buildings making the detection and decoding a difficult task [1]. Therefore, complementary indoor positioning systems have been developed. The current three dominant players in IPS systems are Bluetooth Low Energy (BLE), WiFi and geomagnetic technology, because of their lower installation and maintenance costs. A downside to these technologies is their accuracy, with errors being higher than 1m . For critical applications, requiring accurate position estimates, Ultra Wideband is considered as one of the most promising choices [2]. To link the UWB anchors in their relative frame to the GPS frame, each anchor needs to be calibrated with its exact GPS coordinates. Performing these calibrations manually and finding out at which GPS location they are located is time-consuming and expensive [3].

For automated calibration purposes, Cooperative Relative Positioning (CRP) and Simultaneous Localization and Mapping (SLAM) are widely used, but they are not able to calibrate the system in a global frame. Therefore, an automated procedure is proposed, where a drone with both UWB technology (to measure the distance to the anchors) and RTK technology (to gain an accurate GPS location) is flown around a building to locate the indoor UWB anchors.

II. ULTRA WIDEBAND LOCALIZATION

A. Ultra Wideband

Ultra Wideband (UWB) is a novel radio technology that uses low spectral power density to transmit information over a short

distance over a large bandwidth (larger than 500 MHz). Where conventional technologies transmit information by varying the signal itself, UWB transmits information by generating small pulses at specific time intervals and occupying a large bandwidth. Amongst others, advantages of UWB for localization include accuracy, multipath immunity and suitability for indoor situations. Disadvantages include the cost and incompatibility of the hardware and the time-consuming calculations needed.

B. Two way ranging

Ranging is the method by which the distance between two UWB devices (mostly a tag and an anchor) is estimated. In the Pozyx with the UWB DW1000 chip, Two way ranging is used. This method relies on the Time of Flight (TOF) or Time of Arrival (TOA) from a measurement signal. A timed pulse is sent by a device, and processed for a fixed time t_{reply} by the other device after which it is sent back. From the total round trip time (TOA), the time it takes from one device to another is calculated as the TOF. This time multiplied with the speed of light c (the speed at which the signals are sent) gives then the distance between the devices:

$$d = \text{TOF} \cdot c = \frac{\text{TOA} - t_{\text{reply}}}{2} \cdot c \quad (1)$$

To reduce the errors associated with the clock drift from this system, extensions are used in which the signal is sent back and forth, called symmetrical double-sided two way ranging (SDS-TWR).

C. Localization

The ranging measurements from the previous section can then be used in a localization algorithm called multilateration. Here, from each fixed anchor the distance to the mobile tag is measured. Centered around each anchor, an imaginary sphere is created with as radius the distance to the tag. As such the position of this tag can be calculated by the intersection of the spheres. For 2D localization, at least 3 anchors are needed. For 3D, at least 4.

III. FULLY AUTOMATED CALIBRATION PROCEDURE

In a fully automated drone procedure, many actors are present. In the diagram of Figure 1, the hardware and different connections are shown. A UWB calibration unit (CU) from Pozyx is responsible for the ranging with the UWB anchors. The RTK (real time kinematic) from Emlid provides accurate GPS measurements up to 2cm . This is achieved by using the phase of the GPS signals and sending corrections from a static base. The measurements from these UWB and RTK devices are

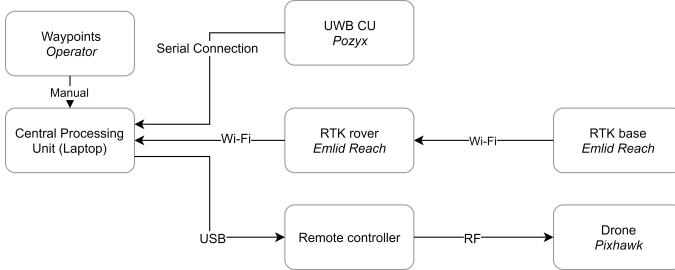


Fig. 1. Communication scheme for the procedure

sent to a central processing unit (CPU) which then creates the path for the drone to follow. Next to the hardware, also accurate measurements and corrections for the UWB are needed. Therefore, first a measurement campaign is carried out after which the measurements are corrected through a Machine Learning algorithm.

IV. (NON)-LINE-OF-SIGHT MEASUREMENTS

A. Line-of-Sight

Line-of-Sight (LOS) propagation is a characteristic of electromagnetic radiation and means the waves travel in a direct path (DP) from the source to the receiver. In Figure 2, RX_1 is in LOS condition with respect to TX .

B. Non-Line-of-Sight

A Non-Line-of-Sight (NLOS) measurement occurs when the DP is either completely blocked (in which case the first arriving path comes from the reflected signal) or from DP excess delay (in which the signal traverses through different materials in a straight line resulting in additional TOF delays) [4]. In Figure 2 with respect to TX , RX_2 is an NLOS example where the DP is delayed and RX_3 an example where the DP is completely obstructed and only the reflections arrive.

C. Bias

Even though LOS measurements can be viewed as ideal measurements, still a bias occurs. This can be explained by the clock drift of the onboard oscillators, resulting in a wrong TOF estimate and thus a wrong measured distance. An effort is made to mitigate this error by the UWB software [5], yet clearly not all errors are mitigated:

$$d_{\text{measured}, \text{LOS}} = d_{\text{actual}} + d_{\text{bias, LOS}} + n_{\text{thermal}} \quad (2)$$

In NLOS measurements, additional to this bias is another bias:

$$d_{\text{measured}, \text{NLOS}} = d_{\text{actual}} + d_{\text{bias, LOS}} + d_{\text{bias, NLOS}} + n_{\text{thermal}} \quad (3)$$

This $d_{\text{bias, NLOS}}$ is caused by the DP blockage and excess delay. n represents the thermal noise, which can be assumed Gaussian distributed with mean 0.

D. Measurement campaign

To investigate the quality of both LOS and NLOS measurements, many geometrical parameters were varied in measurements to extrapolate their importance. Based on Figure 3, the following were varied:

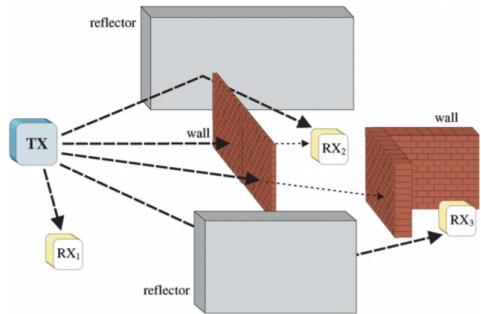


Fig. 2. Examples of different LOS/NLOS situations [4]



Fig. 3. Geometrical orientation of Transmitter (TX) and Receiver (RX)

- Horizontal distance (Δe_x)
- Vertical distance (Δe_z)
- Z-orientation (orientation around e_z -axis)
- Obstacles (different widths, materials, environments)

As it is suggested by Pozyx to always place the anchors and receivers with their e_z -axis facing upwards (since the signal is best transmitted through the $e_x e_y$ -plane), it was chosen to not vary X- or Y-orientation. The onboard parameters were chosen as to perform the most accurate ranging.

V. (NON)-LINE-OF-SIGHT FEATURES

Due to an extra bias occurring in NLOS measurements, it is important to be able to distinguish them from LOS measurements. In the literature many methods are proposed, which are handled in the following subsections. It turned out that some methods performed better in some cases and vice versa. Therefore, all the identification methods will be used as features or parameters in a machine learning approach.

A. Identification via Received Signal Strength

In [6], an identification method is based on the Received Signal Strength (RSS). The basic idea is that in general the RSS in LOS conditions can be over a hundred times stronger than the RSS in NLOS conditions. Mapping out a subset of measurements results in Figure 4, where the distinction is clearly visible. Now, per distance a threshold is calculated capturing the highest 90% of RSS values for LOS. Then it is possible to create a linear fit of these thresholds, denoting the lowest expected RSS value for a given measured distance.

B. Identification via Standard Deviation

In [7], an identification method is proposed depending on the standard deviations of the measured distances at a particular distance. Based on measurements, it was seen that the presence of

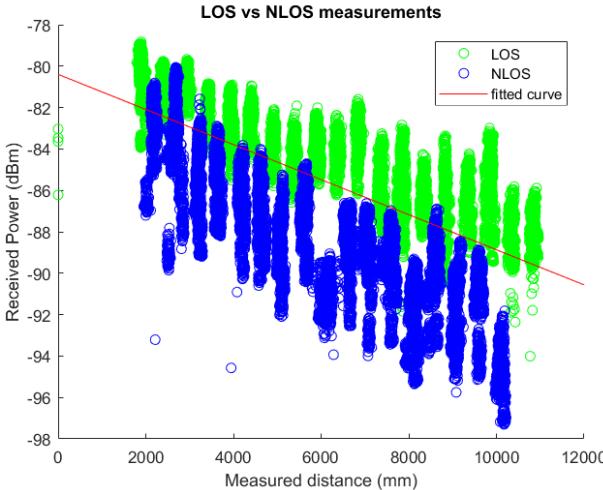


Fig. 4. LOS/NLOS measurements and linear fit for the thresholds

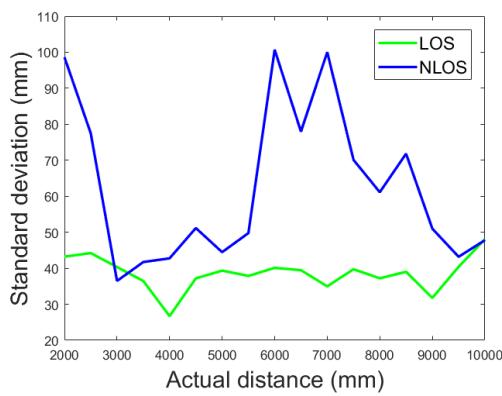


Fig. 5. Standard deviation of measured distance for LOS and NLOS measurements

NLOS errors increases the standard deviation of the measured distance in a significant manner. Applied to our test data -after the filtering of erroneous measurements-, it can be seen that for most cases the standard deviation from LOS measurements stays below 50 mm, while in NLOS goes to higher values, see Figure 5. Therefore, a high standard deviation could indicate NLOS measurements. It is defined as:

C. Identification via CIR time series

Other widely used techniques focus on the shape of the Channel Impulse Response (CIR) [8]. Typically, from the shape of the CIR, it is possible to identify an LOS or NLOS measurement. For example, while an LOS situation returns one sharp peak (the direct path), an NLOS situation typically has more peaks (due to the reflections) and less sharp peaks (more spread out). The following multipath channel statistics (MPCs) are based on the shape of the signal.

C.1 Kurtosis

The first characteristic is the kurtosis, defined as the “tailedness” or “peakyness” of the signal. In other words, for LOS where there is only one sharp peak, one expects a high kurtosis.

For NLOS, where there are multiple more broader spread peaks, the kurtosis will be lower. The kurtosis is defined as the fourth standardized moment:

$$\kappa = \left(\frac{X - \mu}{\sigma} \right)^4 = \frac{\mu_4}{\sigma^4} \quad (4)$$

Where μ represents the mean and σ the standard deviation of the CIR.

C.2 Mean excess delay

Another commonly used statistic is the mean excess delay (MED). It provides information about the time delay for which the energy contained in the multipath falls below a certain level and is defined by the first moment of the power delay profile of the channel:

$$\tau_m = \frac{\int_0^\infty t|h(t)|^2 dt}{\int_0^\infty |h(t)|^2 dt} \quad (5)$$

Where $h(t)$ represents the CIR data in function of time. A high delay signifies NLOS conditions.

C.3 Root mean square delay spread

The RMS delay spread characterizes the richness of multipath in the response and is defined as the square root of the second central moment of the power delay profile:

$$\tau_{rms} = \sqrt{\frac{\int_0^\infty (t - \tau_m)^2 |h(t)|^2 dt}{\int_0^\infty |h(t)|^2 dt}} \quad (6)$$

Thus, a higher delay spread denotes an NLOS measurement. In the paper, this is suggested as the best statistic of the three.

D. Identification via First Path Power

Because the former methods are based on probability models, in [8], an approach is presented which uses the difference between the power contained in the whole signal and the power contained in the first arriving path. The method can be used in real-time and is independent from environment conditions. In LOS measurements, the First Path power should not differ that much from the total power, whereas in NLOS measurements, the First Path may be significantly less than the total power. The paper suggests that a difference higher than 6dBm signifies NLOS.

VI. MACHINE LEARNING APPROACH

A. Introduction

The features that can indicate Non-Line-of-Sight situations are complex and correlated. Each separate feature is not robust and accurate enough when evaluating in different test environments. A solution to this is a combination of these features, with each a given weight. However this weight tuning is complex and time-consuming. This complexity is also found in the mitigation of the distance bias. Since the NLOS bias introduces an extra bias, two different mitigation methods are proposed based on the NLOS identification, see Figure 6.

B. Machine learning procedure

To deal with very complex correlations in a high dimensional feature space, supervised machine learning methods are brought forward. In machine learning, a model is learned by data without explicit programming. This data consists of an input X and output Y . The input X consists of M amount of samples, each defined as a vector with N amount of features. Mathematically, this is denoted as $X \in \mathbb{R}^{M \times N}$, with for the k -th sample $\mathbf{x}_k = \{x_1, x_2, \dots, x_N\}$. The output variables, which we want to predict as accurate as possible, will be described as $Y = \{y_1, y_2, \dots, y_M\}$. The goal is to learn a hypothesis $h : X \rightarrow Y$ so that $h(X)$ is a good predictor for the corresponding output Y . The identification of NLOS/LOS is a biclassification problem, this means the output of the predictor will be two discrete classes, e.g. $h(\mathbf{x}) \in \{0, 1\}$. The error mitigation methods are regression methods, with $h(\mathbf{x}) = d_{bias, estimate} \in \mathbb{R}$. The feature set of the input includes the geometrical parameters (the horizontal and vertical distance and the z-orientation) as well as the features discussed in previous section. A feature selection of this feature set is done in order to improve performance by deleting the irrelevant features for the particular problem. To prevent overfitting or underfitting of the model, the model is trained & tuned by a 5 fold cross validation. This method randomly partitions the training set into 5 equal size subsets. A single subset is retained as validation data, used to evaluate the models accuracy, while the other 4 subsets are used to train the model. This process is repeated 5 times with each of the 5 subsamples used exactly once as the validation data. Finally, in order to evaluate the models performance on unseen data, test samples are taken, completely independent from the training set in different situations (different environments, walls etc.).

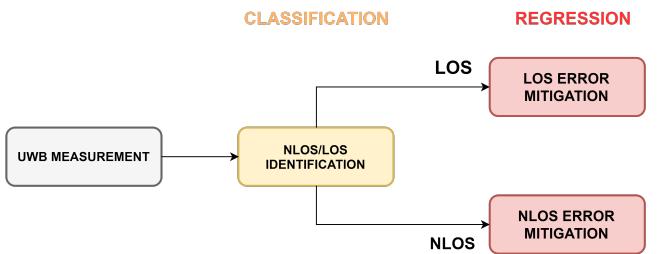


Fig. 6. Machine learning approach

C. Methods

The various methods discussed beneath can be used for classification and/or regression problems.

C.1 Logistic Regression

Unlike the name of the method, Logistic Regression methods are used for classification problems. The name regression is due to the fact that the method outputs the probabilities of the different discrete class labels. This is done by letting the hypothesis $h(\mathbf{x})$ be the independent variable of the Sigmoid function $g(z)$. This function maps each real number of $h(\mathbf{x})$ to the $(0, 1)$ interval:

$$g(h(\mathbf{x})) = \frac{1}{1 + e^{-h(\mathbf{x})}} \quad (7)$$

In order to get to discrete class labels $y \in \{0, 1\}$, the output of the hypothesis function is translated as:

$$\begin{cases} g(h(\mathbf{x})) \geq 0.5 \rightarrow y = 1 \\ g(h(\mathbf{x})) < 0.5 \rightarrow y = 0 \end{cases} \quad (8)$$

C.2 Univariate Regression

Univariate means that only one feature $\mathbf{x} = x_i$ is taken into account in the hypothesis $h(\mathbf{x}) = h(x_i)$. The hypothesis takes following general polynomial form:

$$h_\Theta(x_i) = \Theta_0 + \Theta_1 x_i + \Theta_2 x_i^2 + \Theta_3 x_i^3 + \dots + \Theta_n x_i^n \quad (9)$$

$$= \Theta * \phi(x_i) \quad (10)$$

With $\Phi(x_i) = [1, x_i, x_i^2, x_i^3, \dots, x_i^n]$ and $\Theta = [\Theta_0, \Theta_1, \Theta_2, \dots, \Theta_n]$. The parameter vector Θ is tuned with the help of the cross validation of the training set.

C.3 Multivariate Regression

In multivariate regression, we take into account multiple features in the regression problem. An example of a hypothesis for two features x_1, x_2 with interaction with maximum polynomial degree 2 has the form:

$$h(\mathbf{x}) = \Theta_0 + \Theta_1 * x_1 + \Theta_2 * x_2 + \Theta_3 * x_1 * x_2 + \Theta_4 * x_1^2 + \Theta_5 * x_2^2$$

With a high number of features, a high number of possible regression forms is possible. In order to find the optimal regression form, a grid search is performed. Here, the amount and the selection of features, the interaction term and the maximum polynomial degree are varied.

C.4 Random forest

Random forest is a type of machine learning ensemble methods which makes predictions by averaging over the predictions of several n uncorrelated decision trees ($h_1(\mathbf{x}), h_2(\mathbf{x}), \dots, h_n(\mathbf{x})$). A decision tree $h(\mathbf{x})$ predicts a certain outcome, discrete or continuous, based on conditional control statements in its nodes. An advantage of the random forest method is that it can be used for both classification and regression. However a time-consuming hyperparameter tuning is necessary. Hyperparameters are parameters that need to be set before the model is trained. The most important hyperparameters are the maximum number of features that are allowed in the individual decision tree, the number of decision trees, the maximum allowed number of levels in each decision tree and the minimum leaf size. The optimal random forest model is found by performing a grid search over these parameters.

C.5 Support Vector Machine (SVM)

The advantage of SVM is that they can be used for classification and regression analysis. In short, the goal of support vector machine algorithms is to find the optimal separating hyperplane which maximizes the margin of the training data. For biclassification, this margin means intuitively the largest distance to the

nearest training-data point of the two classes. For regression this is the optimal distance that minimizes the deviation of the points to the hyperplane. The most important hyperparameters in this method are the choice of the kernel function and the regularization and gamma parameters.

D. Performance comparison

D.1 NLOS/LOS Identification Performance

In Table I, the results of the different classification methods on unseen test data can be seen. Although SVM has the highest accuracy rating, the False Positive Rate (FPR) is quite elevated. The False Positive rate gives the relative amount of times $h(\mathbf{x})$ predicts an LOS case while the actual measurement is taken in an NLOS environment. Since a higher importance is given to LOS cases in the eventual localization, this needs to be avoided at all cost. The simple Logistic Regression with the lowest FPR is thus proposed to be further used.

D.2 LOS error range mitigation performance

As can be seen in Figure 7, all the machine learning methods perform quite well. The random forest however gives the highest performance with an improvement of 64% on the mean absolute error and an improvement of 83.3% on the mean squared error relative to the unseen data. With this method, a mean absolute error of 3.8cm could be achieved.

D.3 NLOS error range mitigation performance

As expected, due to the higher hidden complexities, the NLOS mitigation methods perform worse, especially the simple univariate and multiple regression, see Figure 8. Here the random forest method outperforms the others with an improvement of 42.3% on the mean absolute error and an improvement of 57.9% on the mean squared error relative to the unseen data. With this method, a mean absolute error of 17cm could be achieved.

TABLE I

EVALUATION OF THE CLASSIFICATION ON AN UNSEEN TEST SET.

	Logistic	Random forest	SVM
Accuracy (%)	89.6	91.6	94.2
FPR (%)	1.3	4.4	9.5

VII. LOCALIZATION

A. Nonlinear Weighted Least Squares

Due to errors on the (corrected) distance measurements and inaccuracies of the RTK module on the position measurements of the drone, an exact solution of the anchor location cannot be obtained. The most appropriate solution is the minimization of the sum of the squares of the errors. For 2D this is:

$$(x_a, y_a)^* = \underset{i=1}{\operatorname{argmin}} \sum_{i=1}^n W_i e_i^2 \\ = \underset{i=1}{\operatorname{argmin}} \sum_{i=1}^n W_i (d_i - \sqrt{(x_i - x_a)^2 + (y_i - y_a)^2})^2 \quad (11)$$

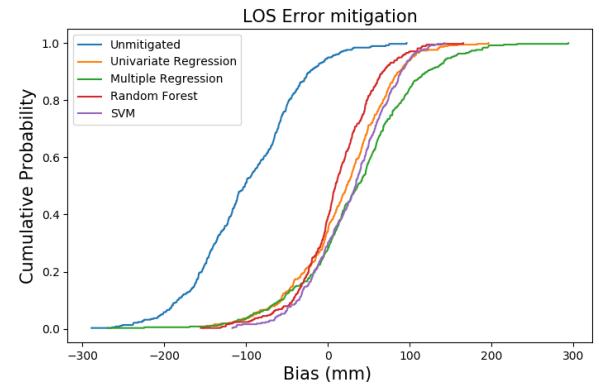


Fig. 7. Cumulative distribution function of the bias on unseen LOS data

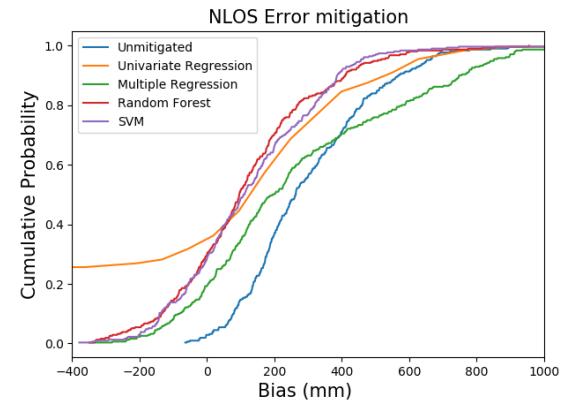


Fig. 8. Cumulative distribution function of the bias on unseen NLOS data

where n is the amount of different distance measurements, (x_i, y_i) the coordinates of point i , (x_a, y_a) the coordinates of the to be localized anchor and d_i the distance between point i and the anchor. Due to the fact that the corrected NLOS distances have a higher error variance than the corrected LOS distances, a lower weight W should be given to NLOS measurements. The accuracy of the nonlinear weighted method strongly depends on the careful choice of the weighting factor per measurement point. This weighting selection is done by:

$$W_i = \frac{1}{\sqrt{\sigma_{pos}^2 + \sigma_{d,i}^2}} \quad (12)$$

with σ_{pos}^2 the position error variance, caused by the inaccuracy of the RTK module and $\sigma_{d,i}^2$ the distance error variance. A lower variance is given to LOS measurements.

B. Geometric Dilution of Precision

Next to the fact that a minimum amount of errors leads to a more accurate final position estimate, the Geometric Dilution of Precision (GDOP) factor should also be taken into consideration. The GDOP describes the additional multiplicative effect of the geometry of the different measurement positions on the precision of the final anchor position estimate. The value can be calculated by all visited positions of the drone combined with

the estimated position of the anchor. The lower this value, the higher the accuracy.

VIII. DRONE PROCEDURE

The proposed drone procedure relies on the calculation of the best position to fly to to achieve a low GDOP. The drone flies around (either by predefined waypoints or at random within a specified range) until the UWB device on the drone has successfully made a connection with an anchor. Then the subprocedure for this particulate anchor will be started. At first, the height of the anchor will be estimated by moving up and down in search for the highest RSS value, because the signal is best transmitted in the $e_x e_y$ -plane. By making a full rotation and matching the obtained signal strength profile with the typical orientation profile, one can estimate the direction of the anchor. Now, a first estimate of the anchor position can be calculated. As soon as there are two measurements, the anchor is estimated using those in the former localization algorithm.

Next, a fictional grid is placed around the current drone position, and for each point of the grid, the improvement of the GDOP if that point were visited is calculated. Thereafter, the drone moves to the point with the lowest GDOP. This can be seen in Figure 9, where the drone moves to the left bottom corner because there the lowest GDOP is found. Positions where the drone has already flown or positions that are not reachable, are deliberately given a high GDOP value to force the drone to find a new point.

Due to not being able to control the drone flight and the RTK not being received yet, this procedure had to be simulated manually in parts. To this extent, the UWB devices were placed on tripods and moved manually based on the input given by the procedure. As no GPS coordinates were available, it was chosen to perform the tests in 2D on a self generated grid. The results can be seen in Figure 10. Starting at (3, 0), the drone moves towards the wall, each time calculating the point with the best GDOP. When the wall is reached, the drone cannot move further in the same line and has to follow the wall. The circles represent how the localization is performed. This procedure was both performed for LOS and NLOS. For LOS, the improvement of localization per iteration is shown in Figure 11. Although in the initial iterations, the corrections may perform worse, in the end their error is significantly lower. For LOS, the error improved from 6cm to 1.9cm, and for NLOS from 26cm to 9cm. It however should be noted that this improvement will be less when also the error of the RTK and vibrations of the drone are present. In these 2 tests we have performed, the algorithm and procedure work well. However, more tests should be executed to test the procedure on a larger scale of environments.

IX. CONCLUSION

A first step towards a fully automatic drone procedure is proposed and tested. Based on the identification and error mitigation using a machine learning approach, the distance errors were reduced both in LOS and NLOS case. Hereby, the localization was significantly improved with the applied corrections, which indicates this could be a sound basis for further work. For future prospects, the drone and RTK should be incorporated and additional decision structures need to be implemented.

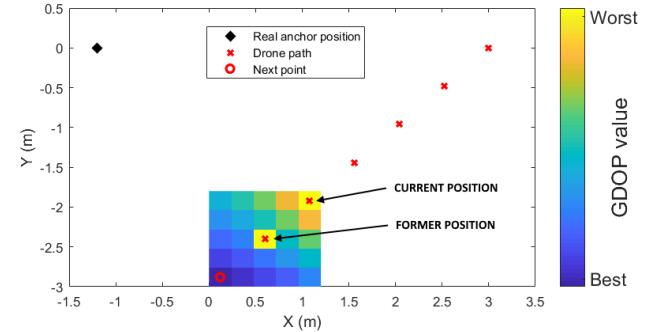


Fig. 9. Creation of GDOP grid to find next point

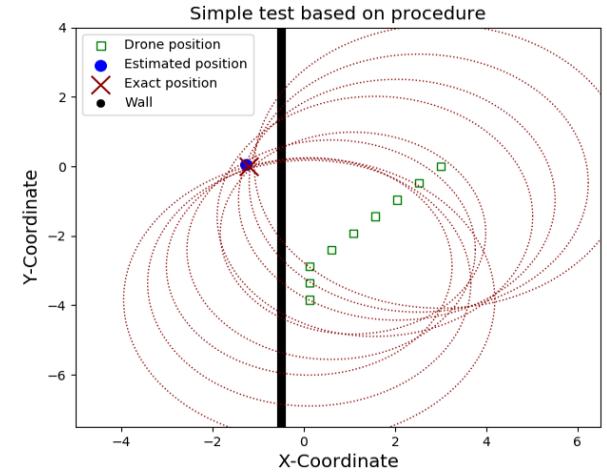


Fig. 10. Simple test on the proposed procedure

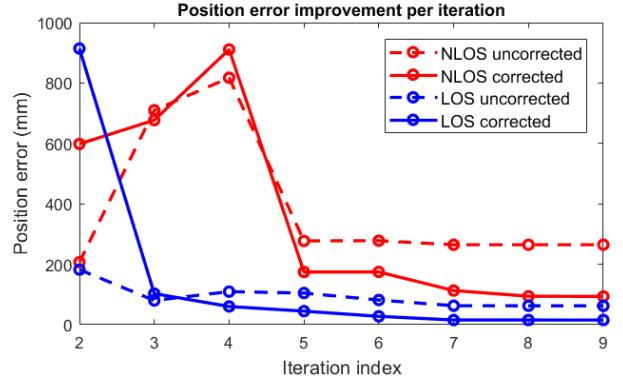


Fig. 11. Position error improvement

REFERENCES

- [1] Ozsoy, K., Bozkurt, A., & Tekin, I. (2013). Indoor positioning based on global positioning system signals. *Microwave and Optical Technology Letters*, 55(5), 1091-1097.
- [2] Alarifi, A., Al-Salman, A., Alsaleh, M., Alnafeesah, A., Al-Hadhrami, S., Al-Ammar, M. A., and Al-Khalifa, H. S. (2016). Ultra Wideband Indoor Positioning Technologies: Analysis and Recent Advances. *Sensors*, 16(5), 707.
- [3] Van de Velde, S., Van Torre, P., & Steendam, H. (2013, December). Fast and robust anchor calibration in range-based wireless localization. In *Signal Processing and Communication Systems (ICSPCS), 2013 7th International Conference on* (pp. 1-6). IEEE.
- [4] Dardari, D., Conti, A., Ferner, U., Giorgetti, A., & Win, M. Z. (2009). Ranging with ultrawide bandwidth signals in multipath environments. *Proceedings of the IEEE*, 97(2), 404-426.

- [5] Decawave. (2015). Sources of error in DW1000 based two-way ranging (TWR) schemes. Retrieved from <https://www.decawave.com/content/sources-error-two-way-ranging>
- [6] Xiao, Z., Wen, H., Markham, A., Trigoni, N., Blunsom, P., & Frolik, J. (2015). Non-line-of-sight identification and mitigation using received signal strength. *IEEE Transactions on Wireless Communications*, 14(3), 1689-1702.
- [7] Wylie, M. P., & Holtzman, J. (1996, September). The non-line of sight problem in mobile location estimation. In *Universal Personal Communications, 1996. Record., 1996 5th IEEE International Conference on* (Vol. 2, pp. 827-831). IEEE.
- [8] Gururaj, K., Rajendra, A. K., Song, Y., Law, C. L., & Cai, G. (2017, September). Real-time identification of NLOS range measurements for enhanced UWB localization. In *Indoor Positioning and Indoor Navigation (IPIN), 2017 International Conference on* (pp. 1-7). IEEE.

Contents

Permission for Usage	i
Foreword	iii
Abstract	v
Extended Abstract	vii
List of Figures	xix
List of Tables	xxiii
List of Abbreviations	xxv
1 Introduction	1
1.1 Background	1
1.2 Problem Statement	3
1.3 Objective of the thesis	5
1.4 Structure of the thesis	6
2 Ultra Wideband localization	7
2.1 Ultra Wideband	7
2.1.1 Working principle	7
2.1.2 Advantages	8
2.1.3 Disadvantages	9
2.2 Ranging algorithm	10
2.2.1 Two way ranging	10
2.2.2 Received Signal Strength (RSS)	11
2.3 Localization	13
2.4 Pozyx	14
2.5 Available data	14

3 Fully Automated Calibration System	17
3.1 Procedure	17
3.2 Hardware	18
3.2.1 UWB	18
3.2.2 Drone	18
3.2.3 RTK	18
3.3 Communication	19
3.3.1 Information retrieval	19
3.3.2 Link between parts	20
3.4 Accurate measurements - UWB ranging	21
3.4.1 Measurement campaign	21
3.4.2 NLOS/LOS identification	21
3.4.3 Error mitigation	21
3.5 Localization	22
3.6 Automated procedure	22
4 (Non)-Line-of-Sight measurements	23
4.1 Line-of-Sight vs Non-Line-of-Sight	23
4.1.1 Line-of-Sight	23
4.1.2 Non-Line-of-Sight	24
4.1.3 Bias	24
4.2 Measurement campaign	25
4.2.1 Measurement set-up	25
4.2.2 Influences	27
4.2.3 Accuracy of the set-up	28
5 (Non)-Line-of-Sight features	31
5.1 Identification via RSS	31
5.2 Identification via standard deviation	35
5.3 Identification via CIR time series	36
5.3.1 Kurtosis	36
5.3.2 Mean excess delay	37
5.3.3 Root mean square delay spread	38
5.3.4 First path vs. Total power	40
5.4 Validation	41
6 Machine learning approach	43
6.1 Introduction	43
6.2 Data handling & evaluation	45
6.2.1 Preprocessing	45

6.2.2	Data splitting	46
6.2.3	Evaluation metrics	49
6.3	Machine learning methods	50
6.3.1	Random forest	51
6.3.2	Support vector machine	54
6.3.3	Univariate regression	58
6.3.4	Multivariate regression	59
6.3.5	Logistic regression	60
6.4	Feature Selection	62
6.5	Performance Comparison	65
6.5.1	Classification	65
6.5.2	Regression	65
7	Localization Algorithm	69
7.1	Ideal Linear Least Squares	69
7.2	Nonlinear Weighted Least Squares	70
7.2.1	Errors	70
7.2.2	Nonlinear weighted squares problem statement	71
7.2.3	Weight selection	72
7.2.4	Solution methods	72
8	Drone Procedure	75
8.1	Geometric dilution of precision	75
8.1.1	Principle	76
8.1.2	Mathematical formulation GDOP	76
8.1.3	Applied on our procedure	77
8.2	Procedure set-up	77
8.3	Procedure	77
8.4	Best point to fly to	82
8.5	Results	85
9	Conclusion and future prospects	89
A	Appendix	97

List of Figures

1.1	GNSS positioning systems. Left: 3D, right: 2D [1]	1
1.2	Dedicated positioning infrastructure	2
1.3	Non-Line-of-Sight problems faced in CRP calibration techniques	4
1.4	Simplified drone calibration procedure	6
2.1	Bandwidths of different technologies [25]	8
2.2	Integration increases signal-to-noise ratio allowing increase in operating range [27]	9
2.3	Two ranging methods [28]	11
2.4	RSS variance for different distances [26]	12
2.5	Trilateration with the theoretical case in black and a practical case in blue [6]	13
2.6	Process from received signal to Channel Impulse Response [32]	15
2.7	Channel Impulse Response gathered from DW1000 registers	16
2.8	CIR diagnostics gathered from DW1000 registers [31]	16
3.1	Diagram of the different hardware technologies	17
3.2	Real time kinematic working principle [33]	19
3.3	Initial explored communication diagram of the different technologies. The connection in red was not established.	20
3.4	Final diagram for the connection with the different technologies	21
4.1	Different LOS or NLOS conditions from a transmitter (TX) to several receivers (RX_i) [32]	24
4.2	Thermal noise on UWB measurements	25
4.3	Fresnel zones	26
4.4	Measurement set-up for a Transmitter (TX) and Receiver (RX)	27
4.5	Measurement error with use of Laser Meter	28
5.1	LOS and NLOS measurements and their least square approximation propagation model: (a) Linear (b) Log-distance [36]	32
5.2	Scatter plot of LOS/NLOS measurements and the linear fit for the thresholds	33

5.3	Distribution of measured distance for LOS/NLOS measurements at an actual distance of 4.5m	34
5.4	Standard deviation of measured distance for LOS and NLOS measurements . .	35
5.5	Zoomed in response	36
5.6	Multipath Channel statistics [41]	37
5.7	Normalized histogram of kurtosis values for LOS and NLOS situations . . .	38
5.8	Normalized histogram of squared RMS Delay Spread	39
5.9	Normalized histogram of squared RMS Delay Spread where the distance is higher than 6.5m	40
5.10	Normalized histogram of $P_{RX} - F_{FP}$: full overlap	41
6.1	Machine learning approach	44
6.2	Supervised learning procedure	46
6.3	Simple example of overfitting and underfitting	47
6.4	Bias-Variance trade off [46]	47
6.5	Data splitting	48
6.6	An example of a decision tree with depth length 3 applied on the LOS data .	52
6.7	Influence of different depth lengths of decision trees applied on the LOS data	52
6.8	Influences of the hyperparameters on the improvement of the MAE for the LOS mitigation method.	54
6.9	SVM classification example	56
6.10	Kernel function [52]	57
6.11	SVM Regression principle [54]	58
6.12	Grid search over the regularization and the gamma parameters. The brighter the higher the validation accuracy.	59
6.13	Univariate regression on the 'Measured Distance' feature	60
6.14	The improvement on MAE for each feature dependent on the maximum polynomial degree for the LOS mitigation method	61
6.15	The improvement on MAE for each feature dependent on the maximum polynomial degree for the NLOS error mitigation method	61
6.16	The logistic function	62
6.17	Ranking of feature importances for LOS mitigation method	64
6.18	Ranking of feature importances for NLOS mitigation method	64
6.19	Ranking of feature importances for LOS/NLOS identification	65
6.20	The cumulative distribution function of the bias of the corrected distance from an unseen LOS test set.	66
6.21	The cumulative distribution function for each machine learning method applied on unseen NLOS test set.	67
7.1	Ideal linear least squares.	70

7.2	Example nonlinear weighted least squares for LOS and NLOS	74
8.1	Two examples of measurement error in satellite navigation	76
8.2	Schematic overview of waypoints	78
8.3	Flow chart of the proposed drone procedure	79
8.4	Search for RSS profile in function of orientation	80
8.5	Search of maximum value for RSS in function of height displacement	81
8.6	Detailed flow chart of the “find next point”-step. The full lines present subsequent steps, the dotted lines datatransfers.	83
8.7	Application of the grid to search for the best point to fly to	84
8.8	Simple test on the proposed procedure	87
8.9	GDOP Improvement	87
8.10	Results of the procedure in a Line-of-Sight situation	88
8.11	Results of the procedure in a Non-Line-of-Sight situation	88
A.1	Relation between bias and signal strength, indicated by DW1000 user manual [28]	97
A.2	Indoor positioning technologies [5]	98

List of Tables

5.1	Results of RSS identification on different datasets	34
5.2	Results of identification on unseen LOS measurements	42
5.3	Results of identification on unseen NLOS measurements	42
6.1	Number of data samples used in each problem	49
6.2	Confusion matrix	49
6.3	Summary machine learning methods	51
6.4	Evaluation of the different classification methods on the unseen test set	65
6.5	Evaluation of the different regression methods on the unseen test set	66
6.6	Evaluation of the different regression methods on the unseen test set	67

List of Abbreviations

AM	Amplitude Modulation
BLE	Bluetooth Low Energy
CIR	Channel impulse response
CRP	Cooperative relative positioning
CU	Calibration unit
DP	Direct path
DW	DecaWave
EFK SLAM	Extended Kalman filter for Simultaneous Localization and Mapping
FAST SLAM	Factored Solution to the Simultaneous Localization and Mapping problem
FCC	Federal Communications Commission
FP	First path (power)
FP	False Positive
FPR	False Positive Rate
FN	False Negative
GDOP	Geometric dilution of precision
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HDOP	Horizontal dilution of precision
ICU	Interface control unit
IPS	Indoor positioning system
IR	Infrared
LOS	Line-of-Sight
MAE	Mean absolute error
MED	Mean excess delay
MPC	Multipath Channel statistic
MSE	Mean squared error

NLOS	Non-Line-of-Sight
ONLOS	Optical Non-Line-of-Sight
OOB	Out-of-the-bag
PCB	Printed Circuit Board
PRF	Pulse repetition frequency
RBF	Radial Based Function
RFID	Radio Frequency Identification
RMS	Root mean square
RMSDS	Root mean square delay spread
RSS	Received signal strength
RTK	Real Time Kinematic
RTLS	Real Time Location System
RX	Receiver
RX	Received Power
SDS-TWR	Symmetric Double Sided-Two Way Ranging
SLAM	Simultaneous Localization and Mapping
SVM	Support vector machines
TDOA	Time difference of arrival
TP	True Positive
TN	True Negative
TOA	Time of Arrival
TOF	Time of Flight
TWR	Two way ranging
TX	Transmitter
UWB	Ultra Wideband
UAV	Unmanned aerial vehicle
VNLOS	Visual Non-Line-of-Sight
WLAN	Wireless local area network

Chapter 1

Introduction

1.1 Background

In order to determine the location of an object, a positioning system is needed. This position can be found in a local or global frame. The most well known examples are the global navigation satellite systems (GNSS). By determining the exact position of each satellite (>3) and measuring their distance to the object, imaginary spheres can be drawn. The 3-D global position of the object is then found by the approximate intersection of these spheres.

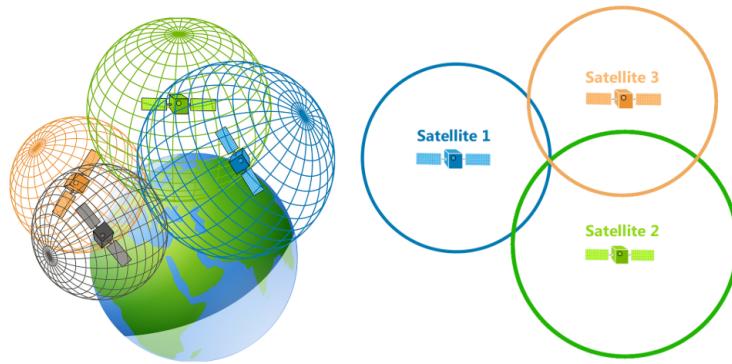


Figure 1.1: GNSS positioning systems. Left: 3D, right: 2D [1]

However, in indoor environments, due to additional signal loss incurred by propagating through walls of buildings, the detection and decoding of GPS signals is difficult [2]. This results in impossible or highly inaccurate distance estimates. Hence, complementary indoor positioning technology to GPS has been developed in the last decades. This research has gained significance by the fact that nowadays more than 80-90% of the western people's time is spent indoors. By 2019, the indoor positioning market is estimated to grow to 4.4 billion with strong demand in healthcare, retail, travel and other sectors [3].

While GNSS has become the dominating system for open-sky, several systems share the indoor market. The diversity of requirements (typically precision, robustness, reliability, cost and scalability [4], [5], [6]) imposed by different applications has resulted in a high number of different available indoor technology systems [7], see Figure A.2. While airports and musea may prefer low cost and scalable systems for location-based content, high specific industry [8] and healthcare applications such as inventory management and patient monitoring require high accurate and reliable systems [9].

The different indoor technologies can be split up in technologies for which a dedicated infrastructure is necessary or unnecessary. This dedicated infrastructure generally consists of a number of fixed nodes, called anchors. These indoor anchors perform similar functions as the satellites in the GNSS systems, namely performing measurements between itself and the to be localized object equipped with a compatible sensor called a tag. A simple drawing of this infrastructure can be seen in Figure 1.2.

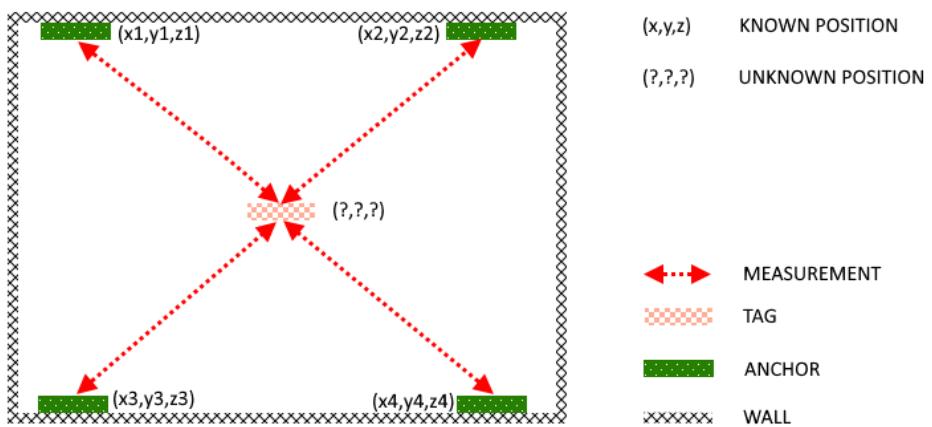


Figure 1.2: Dedicated positioning infrastructure

The three dominant technologies for low accuracy ($> 1m$ error) applications, such as proximity marketing/advertising, are Bluetooth Low Energy (BLE), WiFi and geomagnetic technology [3]. The main advantage of these systems is the overall lower installation and maintenance cost [10]. This is due to the fact that geomagnetic is structure free technology and the ability of BLE and WiFi technologies to use existing building infrastructure. Many buildings are nowadays already fitted with Bluetooth and WLAN hardware used for communication purposes. Another advantage of BLE and WiFi technologies is a commercial market advantage caused by the presence of compatible WiFi or Bluetooth sensors in public devices such as smartphones [11]. This removes the need of providing external tags for localization.

For critical positioning applications however, requiring highly accurate results, Ultra Wideband (UWB) technology is considered as one of the most promising choices [6]. UWB is a radio technology that uses a broad frequency spectrum and hence can employ very short waveforms. These have the capabilities to achieve accurate ranging up to $10cm$. In addition, due to the low frequency part of UWB pulses, the signal can effectively pass through obstacles such as walls and objects [5]. This makes the Ultra Wideband positioning system suitable in small indoor areas with a large amount of obstacles [7], [6].

UWB is one of the most recent positioning technologies and since the Federal Communications Commission (FCC) approved the use of UWB radio in 2002 for the spectrum range of $3.1\text{--}10.6GHz$, a variety of UWB commercial applications have been exploited and several UWB standards have been developed [12]. However, due to the requirement for a dedicated transmitter-receiver infrastructure, UWB has not entered mass market yet [13].

1.2 Problem Statement

Just like the satellites in GNSS systems, the exact positions of the UWB anchors in the dedicated infrastructure need to be determined before any localization can be performed. The more accurate these positions are determined, the more accurate the localization of the indoor positioning will be. This exact position measuring of the anchors is called the calibration process of the localization system. This calibration process however requires a lot of manual labour and is therefore costly, time consuming and may require expensive professional tools [14], [15].

To overcome these drawbacks, a variety of automatic calibration techniques have been developed in the last decades. All these state of the art methods calibrate the anchors in a local frame, which implies localization in relative coordinates. However, a global calibration and thus a global localization can facilitate one of the next challenges positioning systems face in the future, namely the need of a seamless integration of indoor positioning systems with outdoor GNSS systems [16], [17]. A perfect example is the future use of unmanned aerial vehicles (UAVs) in urban areas. These UAVs are equipped with GPS, VISION and UWB sensors in order to accurately locate themselves [18]. When flying close to buildings or going from outdoors to indoors, in order to fuse GPS data with UWB data, the global coordinates of the anchors are needed.

One of the most simple automatic calibration techniques is the use of cooperative relative positioning algorithms (CRP). By measuring the distance of each anchor to its surrounding anchors, the entire network can be localized in a relative frame [19]. In spite of the wall pen-

tration abilities of UWB, obstructions cause erroneous measurements making CRP sometimes highly inaccurate in areas with high level of obstacles [15], see Figure 1.3. Another disadvantage is that in bad geometrically placed situations (e.g. the anchors are place near eachother) the anchor localization can have high errors. The most simple solution to these problems is the use of extra virtual anchors placed in those difficult situations, only used to facilitate the calibration process [20].

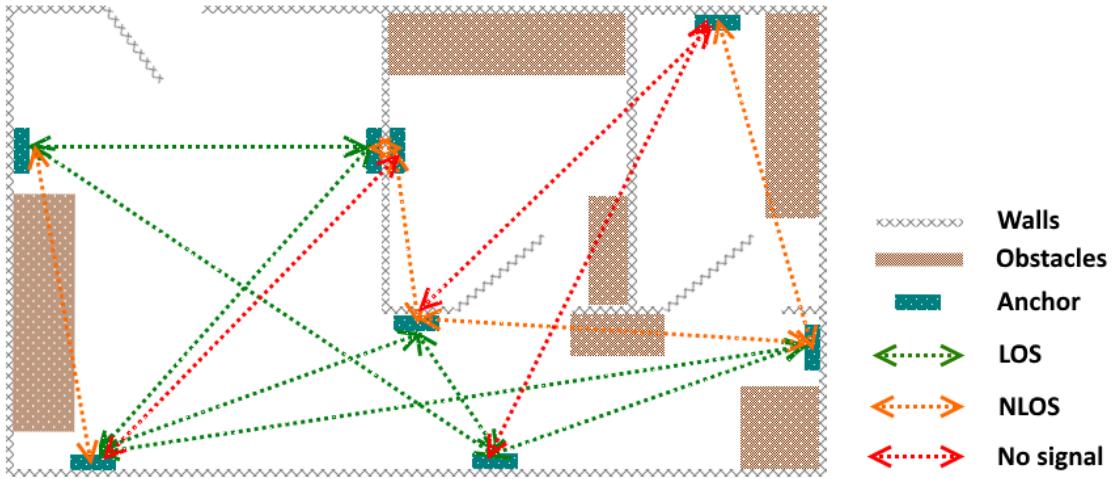


Figure 1.3: Non-Line-of-Sight problems faced in CRP calibration techniques

Other popular calibration techniques make use of extra equipment including an extra dynamic UWB device. This device will perform measurements with the fixed anchors and will from now on be denoted as the calibration unit (CU). The particular advantage of the use of the CU is its mobility, meaning that now from a high number of positions, measurements can be performed. This higher number of positions and measurements reduces the error of the calibration. Next to this obstacles can easily be avoided. This logically leads to a more accurate final position estimate than the CRP algorithms, which only can perform distance measurements from fixed anchor points.

An example is the live dynamic calibration technique called Simultaneous Localization and Mapping (SLAM). In contrast to predetermining the position of the anchors, the position of the CU and the anchor are simultaneously estimated. In SLAM, the map and the CU trajectory are estimated without prior knowledge on the anchor locations [21]. However, the use of SLAM is limited to automated devices. Most well known algorithmic implementations, such as FAST SLAM and EFK SLAM rely on motion models and require the knowledge of the so called control vector that is controlling the CU [15]. An example of a SLAM calibration process is the localization of an indoor flying UAV equipped with a CU. Simultaneously the

location of the UAV and the coordinates of the fixed anchors are calculated [22].

It is easily noticed that these state of the art calibration methods only have the ability to calibrate the anchors in a local frame. A first old approach to globally calibrate the indoor UWB anchors was the collection of a set of outdoor measurements from the CU attached with a GPS module. However, due to the large errors in GPS coordinates and the extra errors induced by the walls, it is obvious that these methods could only provide very rough anchor information with anchor localization errors up to ten meters. Since the development of Real Time Kinematic (RTK) and the decreasing cost of these devices [23], the old outdoor calibration approach could significantly be improved. RTK is a satellite navigation technique that enhances the precision of position data from satellite based systems. While GNSS have accuracy up to a meter, RTK provides up to centimeter-level accuracy. An updated dynamic calibration technique is thus proposed in which an RTK, instead of a GPS module, is attached to the CU. Due to the high mobility of the CU and the accuracy of the RTK, estimation errors of the positions of the fixed anchors could now be accurate up to 5cm. This technique maintains the advantage of estimating the positions in a global frame. The main disadvantage in this technique are the extra wall obstructions caused by measuring outdoors.

1.3 Objective of the thesis

The objective of the thesis is to develop and test a novel UWB anchor calibration process for a UWB localization system. This calibration process differs from other state-of-the art calibration processes by the fact that it can locate the position of the anchors in a global frame. This novel calibration will be done with the help of a drone equipped with a CU and an RTK module. The main goal of this thesis is to develop an automated procedure in order to achieve a minimal estimation error of the global coordinates of the localized anchors.

In Figure 1.4, a simplified procedure is shown. A stable flying drone follows an appropriate path, preferably near open windows or thin walls, to make range measurements (blue line) with the fixed anchors. Meanwhile, the precise global position of the drone is measured with the help of the RTK module (red line). Combining the CU and the RTK measurements, a first estimation of the fixed anchor(s) can be made. To improve the estimation, the drone flies to a new appropriate optimal position that is calculated based on algorithms and methods proposed in this thesis. After reaching the new position, novel measurements are taken.

The main issues in this thesis are to find appropriate and robust methods to identify whether distance measurements are disturbed by obstacles such as walls, correct the distance errors that are mainly caused by the disturbance of obstacles, calculate the estimated anchor position based on a localization algorithm and combining all this in an automated procedure. A

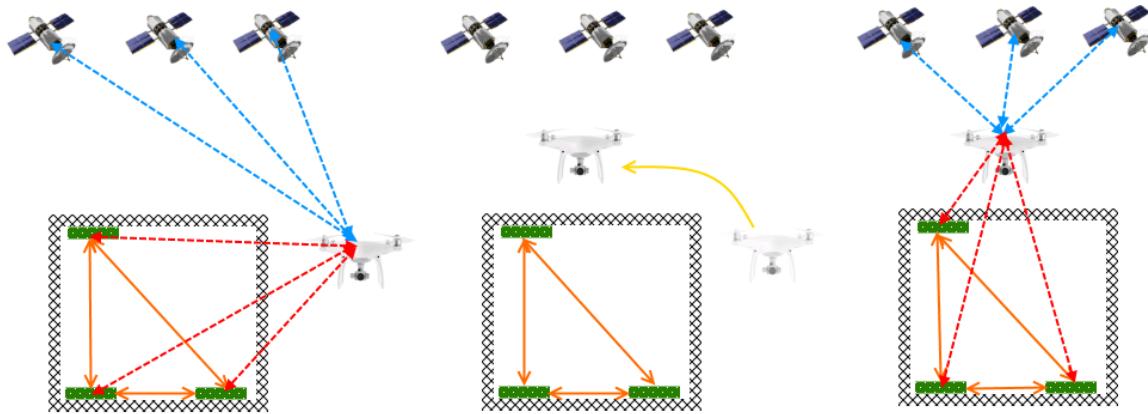


Figure 1.4: Simplified drone calibration procedure

future prospect in this procedure can be the integration of the state-of-the-art CRP algorithm (orange line) into the procedure in order to further reduce position errors.

1.4 Structure of the thesis

In Chapter 2, the theory of Ultra Wideband technology used for localization is explained as well as the specific advantages and disadvantages of UWB as an indoor positioning system. Based on the collection of ranging measurements in a UWB network, the principle of multilateration is explained that is used for estimating the position of a tag. Chapter 4 provides the reader with a thorough explanation about the differences between Line-of-Sight and Non-Line-of-Sight and the impact on the error occurring in distance measurements. Next to this the extensive measurement campaign that was performed is explained. This campaign was done in order to investigate the influence of different parameters on the identification of Non-Line-of-Sight and on the error on the distance measurements. In Chapter 5, different features are extracted in order to find good indicators to identify Non-Line-of-Sight identification. These indicators are evaluated on some independent measurement samples. Because the identification of Non-Line-of-Sight and the mitigation of the error depends on a high amount of variables that are internally correlated, machine learning methods are presented in Chapter 6 in order deal to with these complexities. After this, the various methods to solve the identification and the error mitigation are evaluated on independent test samples. In Chapter 7 a mathematical framework is explained in order to estimate the position of the anchors based on the corrected distance measurements and the identification of Line-of-Sight/Non-Line-of-Sight. The results of the previous chapters are finally combined in an automated drone procedure. A simple waypoint procedure is presented and the different levels of this procedure are separately tested and evaluated. Finally an overall conclusion of the thesis is presented in the last Chapter 9.

Chapter 2

Ultra Wideband localization

In this thesis, all distance measurements will be executed with Ultra Wideband (UWB) technology. First, in Section 2.1, the technology of Ultra Wideband localisation and how it compares to other technologies will be discussed. Section 2.2 provides an explanation of the ranging method used for measuring the distance between two devices. The localization algorithm based on this ranging is explained in Section 2.3. Furthermore, Section 2.4 gives information about the specific hardware used for these UWB measurements and finally, Section 2.5 shows the different data that can be extracted from these devices.

2.1 Ultra Wideband

For indoor positioning, UWB technology has been proven to show better performance to other technologies such as Radio Frequency Identification (RFID), Infrared (IR), Ultrasonic, Wireless local area network (WLAN), Bluetooth and others [6]. UWB is a radio technology that uses low spectral power density to transmit information over a short distance over a large bandwidth (larger than 500 MHz). Where conventional systems transmit information by varying the amplitude, frequency and/or phase of a sinusoidal wave, UWB transmissions transmit information by generating very short pulses and occupying a large bandwidth, which leads to a lower power density. This can be seen in Figure 2.1, where the large bandwidth and low power density of UWB compared to conventional technologies is clearly visible.

2.1.1 Working principle

UWB can rely on a broad bandwidth spectrum, which results in short pulses, as the bandwidth and length of the pulse are inversely proportional. Due to this large bandwidth, The FCC sets a power emission limit of $-41.3dBm/MHz$ for unlicensed transmissions [24]. Consequently, the pulses fall below the noise floor of $-40dBm/MHz$ (the red striped line in Figure 2.1).

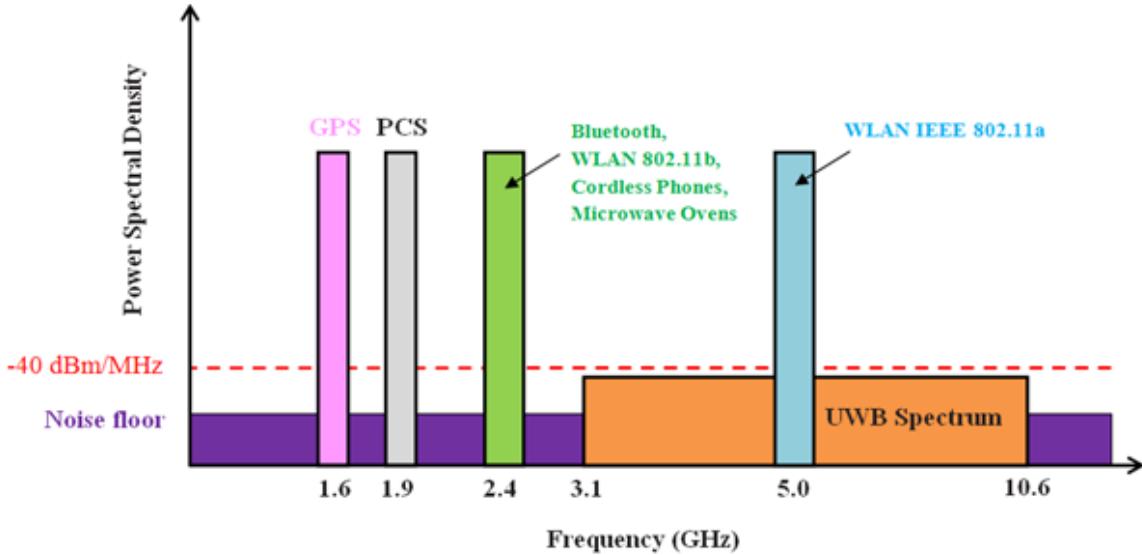


Figure 2.1: Bandwidths of different technologies [25]

To cope with this issue, a train of pulses is sent by the transmitter (typically a power of 2) to represent a single bit of information. These received pulses will be accumulated and the resulting signal will be above this noise level, making the signal receivable [26]. This accumulation of pulses is called Coherent Integration and an example of this technique for 16 and 512 pulses can be seen in Figure 2.2, where the significant better signal of the integration of 512 pulses is visible.

2.1.2 Advantages

In this section all major advantages of UWB for localization purposes are discussed, some of which briefly already discussed in former sections.

Accuracy Due to the broad spectrum of UWB, shorter pulses can be generated relative to other technologies. This implies that more accurate measurements can be obtained.

Multipath immunity Multipath is the propagation phenomenon that results in radio signals reaching the receiving antenna by two or more paths, this can be caused by reflections of the signal on walls or other obstacles. It causes unwanted constructive and destructive interference and phase shifting of the signal. Due to the short pulses of the UWB signal (0.16ns), it is easier to identify the direct path (Line-of-Sight) from reflectors. Compared to for example WiFi, where any object within 1.2m will cause an overlapping pulse due to the longer pulse time of 4ns .

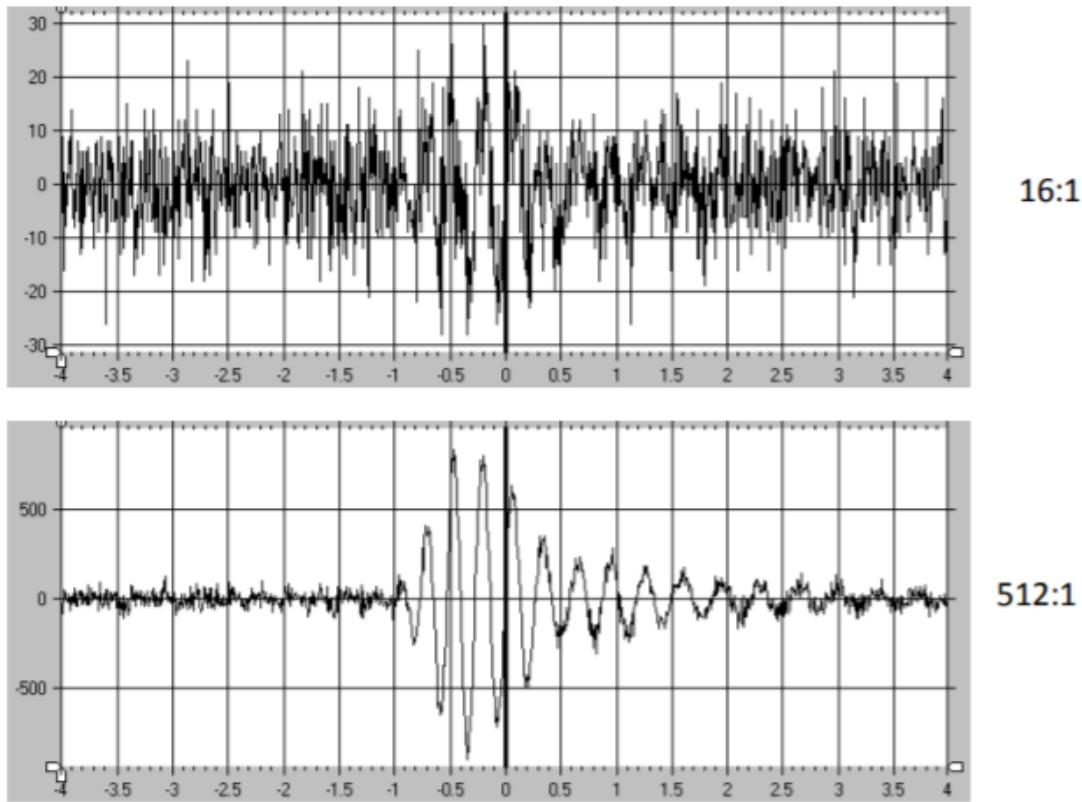


Figure 2.2: Integration increases signal-to-noise ratio allowing increase in operating range [27]

Coexistence with other technologies Although UWB works on the frequency of other narrowband technologies, it introduces no interference, as the power level is below the noise level of these other technologies (See Figure 2.1). Thus, it can be used for localization purposes without affecting the performance of existing technologies.

Suitable for indoor situations Mainly because of the multipath immunity and the co-existence with other existing technologies, the UWB is well suited for indoor environments. In addition, the low frequency of the pulses enables the signal to effectively pass through obstacles, which improves accuracy [6].

2.1.3 Disadvantages

Although the UWB does seem to have many advantages concerning indoor environments and accuracy, it has also some minor disadvantages, which need to be addressed:

Cost Due to the need of a dedicated transmitter-receiver UWB infrastructure, it is more costly than conventional WiFi or BLE based positioning systems [6], which can mostly rely on using existing building infrastructure.

Compatibility Since UWB has not entered mass market yet, existing public devices do not have the technology onboard. Where for example smartphones have WiFi and BLE possibilities, for UWB an extra module needs to be added in order to localize the smartphone.

Time-consuming A UWB receiver needs both signal acquisition, synchronization and tracking to be done with very high precision in time relative to the pulse rate. All these steps are time-consuming resulting in a lower update rate [6].

2.2 Ranging algorithm

Ranging is the method where two nodes (in most cases tag and anchor) communicate between themselves and exchange messages by which the mutual distance can be calculated. In this section, two common methods for ranging with UWB are covered.

2.2.1 Two way ranging

The first method relies on the Time of Flight (TOF) or Time of Arrival (TOA) from a measurement signal. As can be seen in Figure 2.3a, the ranging is performed by the following procedure: one device sends a signal pulse to another device ($t = TOF$), which then processes this signal for a predefined known time ($t = t_{reply,B}$) and sends it back to the first device ($t = TOF$).

This first device can then measure the time elapsed between sending the pulse and receiving the pulse, also known as the Time of Arrival. The TOF can be calculated by subtracting the processing time from the TOF and then dividing by two. Since the waves travel at the speed of light, the distance between the devices is then calculated by multiplying the TOF with the speed of light c :

$$d = TOF \cdot c = \frac{t_{round,A} - t_{reply,B}}{2} \cdot c \quad (2.1)$$

Error in Two Way Ranging Because this way of ranging depends highly on the processing time, an error is induced according to:

$$Error = \frac{1}{2}t_{reply,B}(e_A - e_B) \quad (2.2)$$

Where e_A and e_B represent the frequency error with respect to the nominal oscillator frequency of the onboard free running crystal oscillator for making the electrical signals. For practical values of such a frequency offset, the error in the accuracy of the range is large [28].

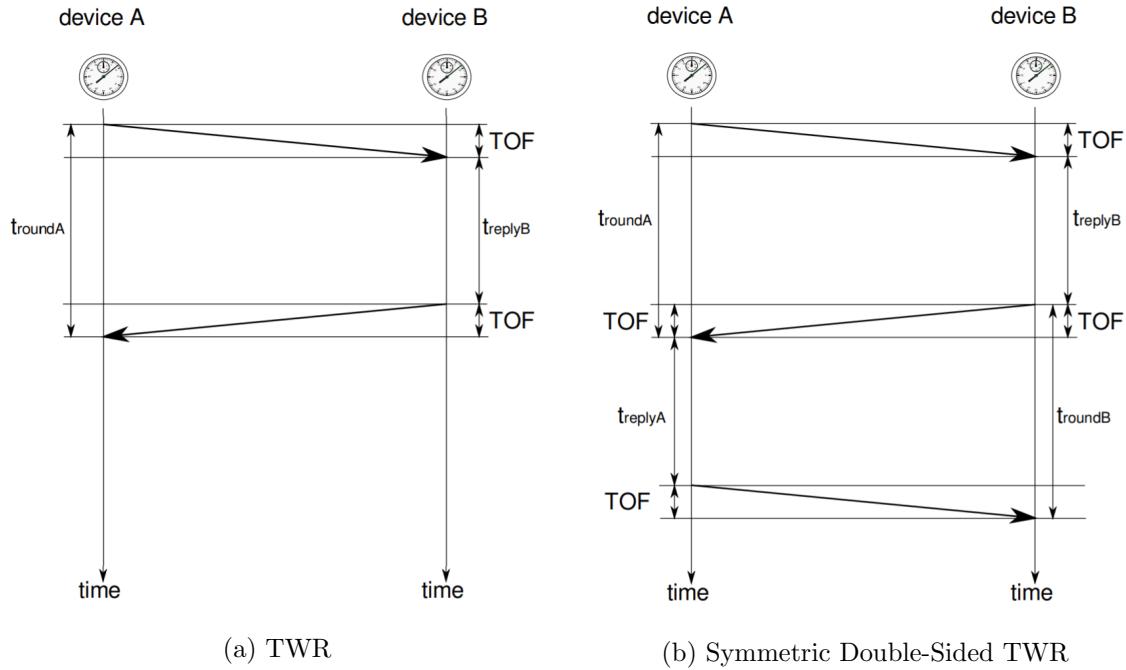


Figure 2.3: Two ranging methods [28]

Symmetric Double-Sided Two Way Ranging To minimize this error, an alternative scheme is used introducing another message, as seen in Figure 2.3b. The former mentioned procedure is now performed twice, and the distance now can be calculated as:

$$d = \text{TOF} \cdot c = \frac{t_{\text{round},A} - t_{\text{reply},A} + t_{\text{round},B} - t_{\text{reply},B}}{4} \cdot c \quad (2.3)$$

The dominant error with the method is given by:

$$\text{Error} = \frac{1}{4} \Delta_t (e_A - e_B) \quad (2.4)$$

In this formula, the dependence on t_{reply} has been replaced by a dependence on the term $\Delta = t_{\text{reply},A} - t_{\text{reply},B}$, which is much smaller and introduces a smaller error.

2.2.2 Received Signal Strength (RSS)

Another popular method for estimating the distance is by looking at the Received Signal Strength (RSS). The key idea in this technique is that the further two devices are from each other, the smaller this signal strength will be. This is based on the transmission formula of Friis, which relates the power between nodes and their mutual distance in ideal conditions [28]:

$$P_r = P_t + D_t + D_r + 20 \log_{10} \frac{\lambda}{4\pi d} \quad (2.5)$$

Here, P_r represents the power available at the receiving antenna and P_t the power provided into the transmitting antenna. D_t and D_r represent the isotropic directivity of the transmitting and receiving antenna and d the distance between the two nodes. From this formula, when the distance d increases, the received power P_r would have to decrease if the directivity factors and transmitted power stay constant.

However, especially in indoor situations where multipath is present, this does not hold because ideal conditions are almost never achieved. The received strength will be a combination of all received pulses. In some conditions, obstructions will weaken the signal resulting in a lower received strength. In other conditions, the different reflections of the signal will strengthen the total received power. In Figure 2.4, the mean RSS value and error bars are shown for different distances. It is clear that there is both a large dispersion on the RSS-value, and that there is a major overlap of RSS-values for different distances. For example, a strength of $-84dBm$ could indicate a distance of 2 to 4 meters, but even a distance of 5 meters.

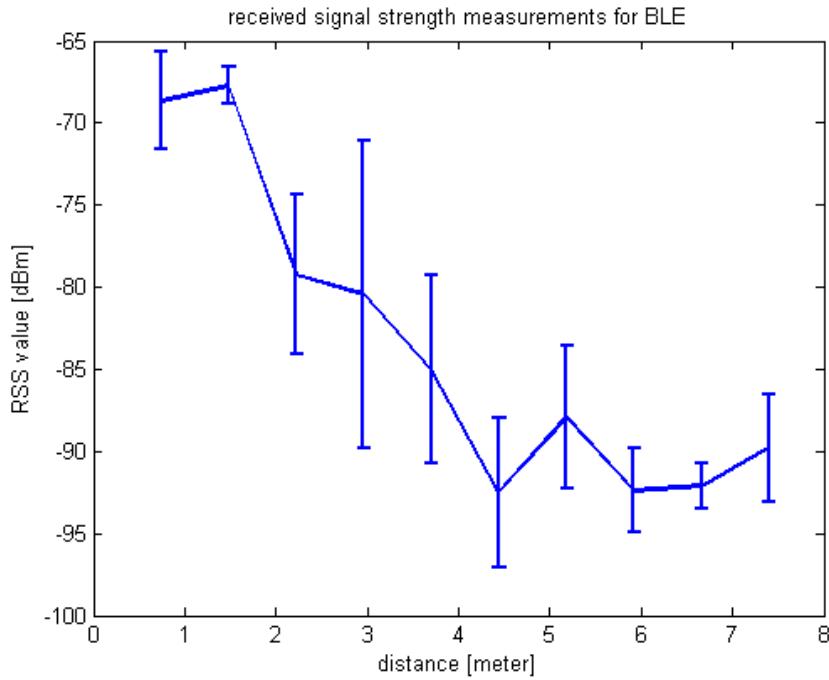


Figure 2.4: RSS variance for different distances [26]

2.3 Localization

In this section, the localization algorithm is explained. Typically, one wants to find the location of a mobile unit, called a tag, based on some reference points, called the anchors. The position for these anchors is known and through their distance from the tag, the position of the latter can be calculated. This method is called multilateration. To explain how this method works, let us first assume an ideal world with ideal measurements and explain the method for 2-D, accompanied by Figure 2.5. If one tag and one anchor are present, and both the position of the anchor p_a and the distance a between this anchor and the tag is known, one can be sure that the tag is located somewhere on the circle around the anchor with a radius $r = a$. Adding another anchor p_b to this configuration with a distance b between this anchor and the tag, it is sure that the tag also is located on the circle around p_b with radius $r = b$. The combination of these two circles gives two intersections, meaning that both conditions can only be satisfied on these two points. Adding another anchor c to the configuration and doing the same thing, a unique location can be found for the tag. This process of uniquely defining a tag in 2-D is called trilateration and is shown with the black lines in Figure 2.5. To move to 3 dimensions, another dimension is added and as such another anchor is needed to uniquely define the tag, which is called multilateration.

In the real world, distance measurements are not perfect and the absolute anchor position is not exactly known. Therefore, the distance which is estimated will not correspond to the actual distance. Hence, the position based on these distances will result in an estimated position differing from the real position, as can be seen in Figure 2.5 in blue.

Thus, for accurate positioning, precise anchor positions need to be known, calling the need for an accurate calibration.

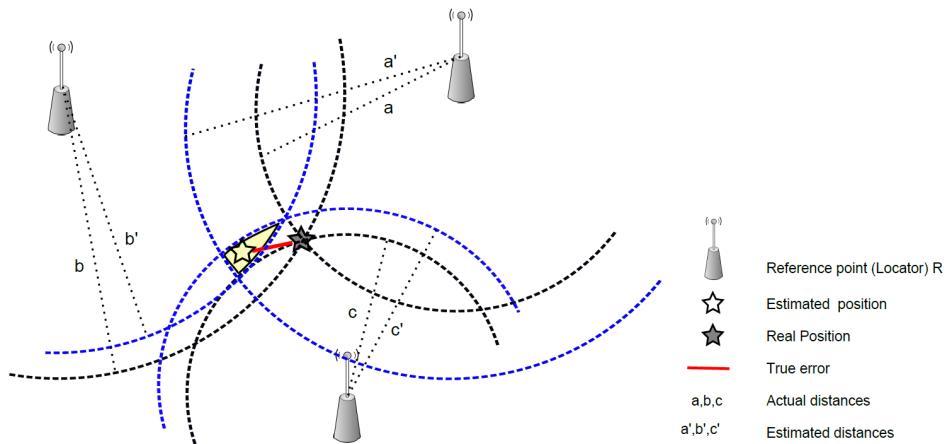


Figure 2.5: Trilateration with the theoretical case in black and a practical case in blue [6]

2.4 Pozyx

In this thesis, the hardware from Pozyx is used for the UWB measurements. Pozyx is a start-up and already provides UWB solutions in 45 countries. On the Pozyx chip, the well known DecaWave DW1000 UWB chip is fitted, which can be used in Two-way ranging or TDOA location systems and promises accuracies up to 10cm [29].

The DW1000 UWB-module itself only provides accurate timestamps and Pozyx uses these timestamps in state-of-the art algorithms to obtain the most accurate range and positioning information. Next to this information, Pozyx also provides the possibility for transmitting messages over the air, which enables not only over-the-air data transfer but also remote command possibilities. In this way, it is possible to measure the distance between two hard-to-reach devices with a third device.

Anchor placement Pozyx uses two way ranging and multilateration for the positioning. As this positioning comes with errors, Pozyx provides some rules of thumb for the placement of the fixed anchors to reduce these errors (this will be more explored in the next chapters) [30]:

- Placed high and in Line-of-Sight, to reduce distance measurement errors
- Spread around the tag, to reduce the accumulated error in the localization
- Placed vertically with the antenna on top, since by this orientation the signals are best transmitted
- Placed at different heights if 3-D positioning is needed, to reduce the accumulated error in the localization

2.5 Available data

In later chapters, the quality of the distance measurements will be investigated. To enable this, it can be useful to list the data Pozyx provides for a range measurement:

- timestamp [ms]: a unique timestamp for each measurement
- distance [mm]: the measured distance with the Two Way Ranging method
- RSS [dBm]: the Received Signal Strength (rounded to the integer)

For regular cases this is sufficient, however for our goal it is needed to have an idea of the full received signal for e.g. the NLOS/LOS identification. For the LOS bias correction, the RSS needs to be available on a more accurate scale. Via some registers, it is possible to acquire additional diagnostics provided by the DW1000 chip [31]. One of these is the Channel Impulse Response (CIR) data. This data comes from the received signal, which is first processed

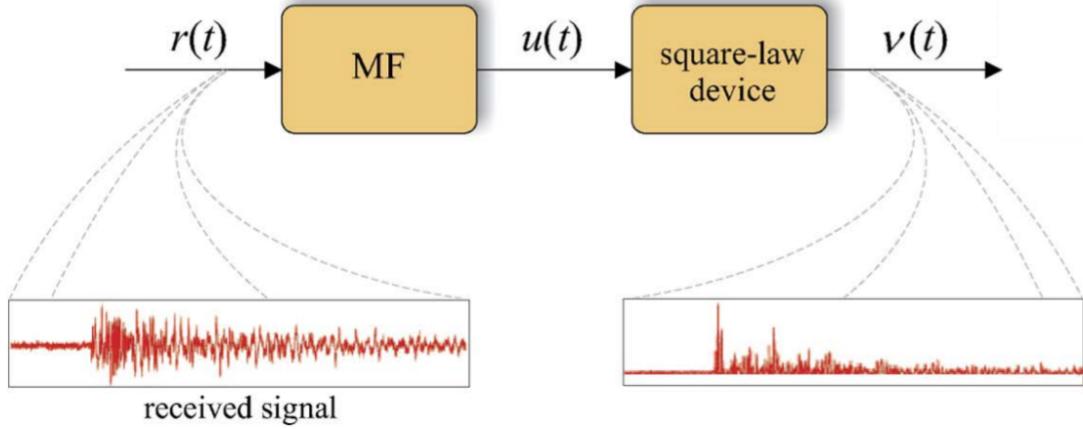


Figure 2.6: Process from received signal to Channel Impulse Response [32]

through a Matched Filter (MF) matching the signal with the UWB pulse. Subsequently, the obtained signal is sent through the square-law device, which removes the signs of the signal peaks. The process is shown in Figure 2.6 [32] and an example of a CIR signal sampled with 1000 samples can be seen in Figure 2.7.

Next to the CIR, other diagnostics useful for the NLOS identification are available, based on this CIR. An example of some of these characteristics is available in Figure 2.8 [31]. Here, the First Path (FP) represents the first peak that is detected above the noise level. This is seen as the direct path from the transmitter to receiver. However, due to obstructions, the real first path may be below the noise level and not detected. Every peak after this FP counts as a peak generated due to reflections. The Peak Path signifies the maximum peak of the signal. This peak path could indicate NLOS situations, which will be discussed in Chapter 4.

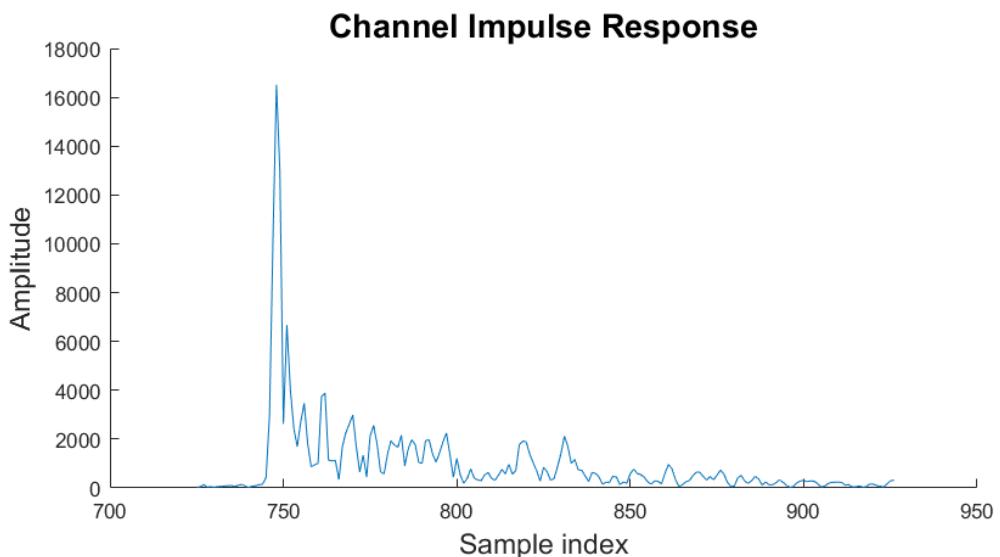


Figure 2.7: Channel Impulse Response gathered from DW1000 registers

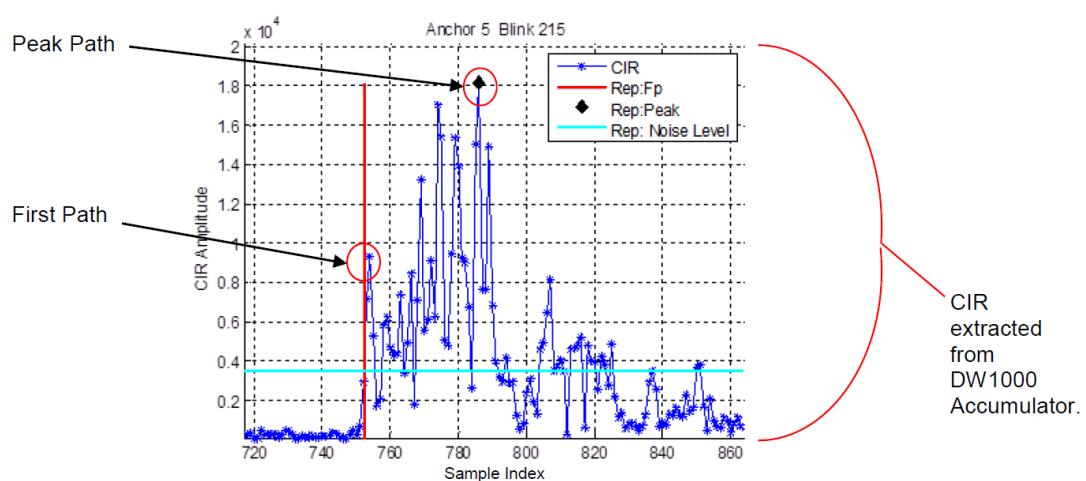


Figure 2.8: CIR diagnostics gathered from DW1000 registers [31]

Chapter 3

Fully Automated Calibration System

The goal of the thesis is to achieve a fully automated calibration procedure following the objective of the thesis. Such a broad procedure comprises many actors on different levels and needs to be addressed likewise. In this chapter, the different levels (e.g. hardware) and their actors (e.g. UWB, RTK ...) are discussed.

3.1 Procedure

Following the problem statement, the calibration procedure can be presented in essence by the diagram of Figure 3.1. At the same time, UWB measurements between the drone and anchors and RTK measurements between the drone and satellites are gathered. These are then sent to the Central Processing Unit (CPU), in this case a laptop. Based on this data, an optimal flight pattern for the drone is generated. The CPU sends this pattern to the drone, which then moves accordingly.

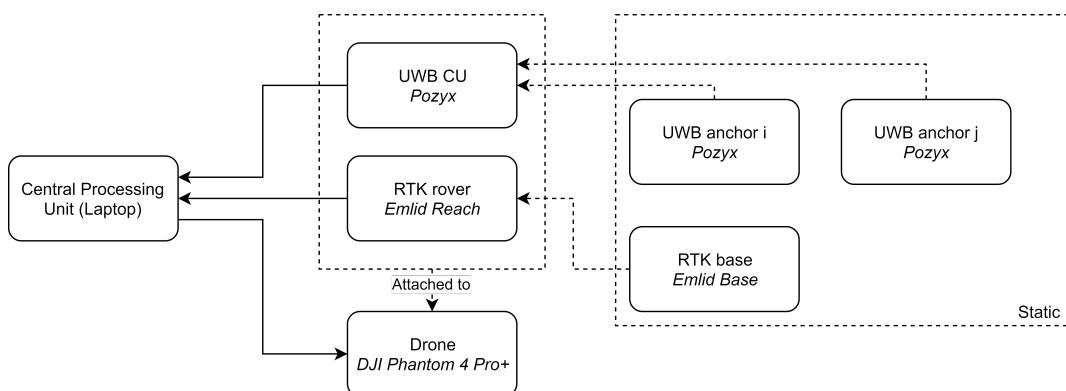


Figure 3.1: Diagram of the different hardware technologies

3.2 Hardware

First of all, to establish such a scheme, all components must be present and suited for the given task. In this section, a short overview is given for the different technologies cooperating in the procedure.

3.2.1 UWB

For indoor positioning systems, UWB is a highly suited radio technology. Here, the hardware from start-up *Pozyx* is used, translating the accurate timings from the *DW1000 UWB* chip into a handy Arduino/Python libraries, promising accuracies up to 10cm .

In regular cases, many fixed UWB anchors are used to locate the position of the mobile tag. In our case, to locate the position of the anchors, the working is opposite. Here, a moving UWB tag or calibration unit (CU) is placed on the drone making distance measurements with the different fixed anchors in an attempt to locate them.

3.2.2 Drone

For accurate UWB/RTK measurements and localization, a stable drone is needed. Therefore, the *DJI Phantom 4 Pro+* was used, the most recent high-end version of the *Phantom* series by the popular consumer drone corporation *DJI*. With the use of many kinds of sensors, a very stable flight is guaranteed. On top of that, obstacle detection algorithms make sure the drone never crashes into objects.

3.2.3 RTK

To achieve an accurate calibration procedure, accurate measurements for the global coordinates are needed. Where normal GPS solutions can not provide this, real time kinematic (RTK) systems can, promising accuracies up to 2cm . In this technology, not only the satellite signal itself but also the phase of it is used. As the resolution of this phase is in the order of 20cm and the accuracy by which this phase can be measured is 1 over 100, this results in a theoretical accuracy of about 2mm (which in practice will be lower due to noise), a large improvement on the traditional GPS. However, to achieve this, there is a constant need of an interaction between 2 modules. One is the base, a fixed module transmitting correction data to the other module, the rover. The rover is the moving module, of which the position is to be determined, see Figure 3.2.

In our set-up, the RTK hardware of *Emlid* was used. *Emlid* was chosen for the accuracy, cost and easy connection capabilities. The rover module is (together with the UWB CU) attached to the drone and provide accurate GPS measurements for the exact position of the drone.

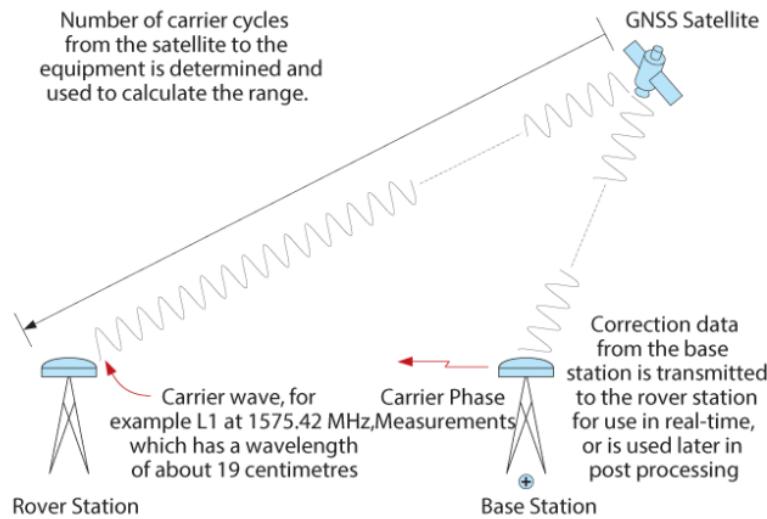


Figure 3.2: Real time kinematic working principle [33]

The base module of *Emlid* is fixed on the ground and sends corrections to the base.

3.3 Communication

An important consideration is the communication with the different technologies, which will be discussed in the first subsection. Not only the direct communication with the technologies, but also the communication between them has to be considered, which will be handled in the second subsection. An overview of the chased connection scheme is available in Figure 3.3.

3.3.1 Information retrieval

UWB

For the communication with the UWB hardware, two main options are available. The first one is using the Pozyx as a shield on an Arduino with C code and sending the data from the Arduino to the CPU. The second one is working with Python directly on the Pozyx and sending the data remotely by UWB to another Pozyx device attached to the CPU. A first attempt was made by using an *Arduino Tian*, which can transmit data to the CPU with its onboard WiFi capabilities. Sending data from the *Tian* to our computer over WiFi was no problem. However, getting the data from the UWB on the *Tian* did not work as this *Arduino* version is not supported by the *Pozyx* libraries.

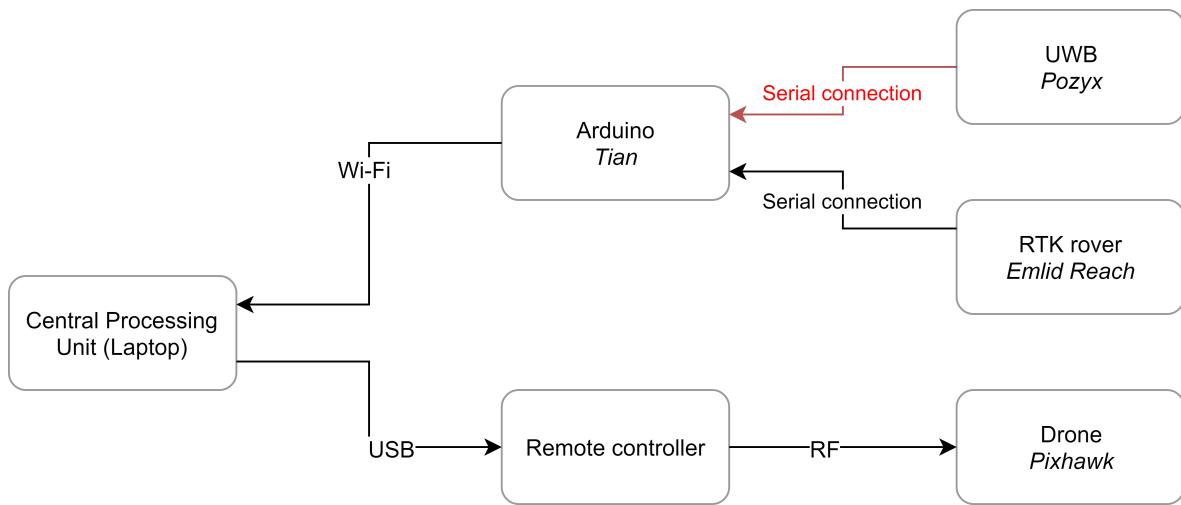


Figure 3.3: Initial explored communication diagram of the different technologies. The connection in red was not established.

Therefore, it was suggested to work with Python. With use of a *Pozyx* device attached to the computer, the data from the *Pozyx* CU could be read on the computer. However, this has as consequence that the data rate is halved.

RTK

At the moment of writing, the RTK had just arrived and initial tests showed that the data from the rover and base could be captured on a computer. Further connection tests were out of the time scope.

Drone

To maintain the stability and obstacle detection of the drone, *DJI* does not allow users to take over the internal processes of the *Pro+* version. Many third party software providers deliver software, which acts on the remote controller to predefined certain waypoints for other versions of the *Phantom*, but even this is not available for the *Pro+* version. For future prospects, open source drone software like *Pixhawk* could be advised, where altering the controller code is easy and documented.

3.3.2 Link between parts

The former sections already mention some of the problems encountered while trying to manage all parts for the calibration. Following the decision to work remotely with the *Pozyx*, the RTK needs to be directly connected with the CPU, with its onboard WiFi. A diagram for this final set-up is available in Figure 3.4. Consequently, the timing of the received measurements

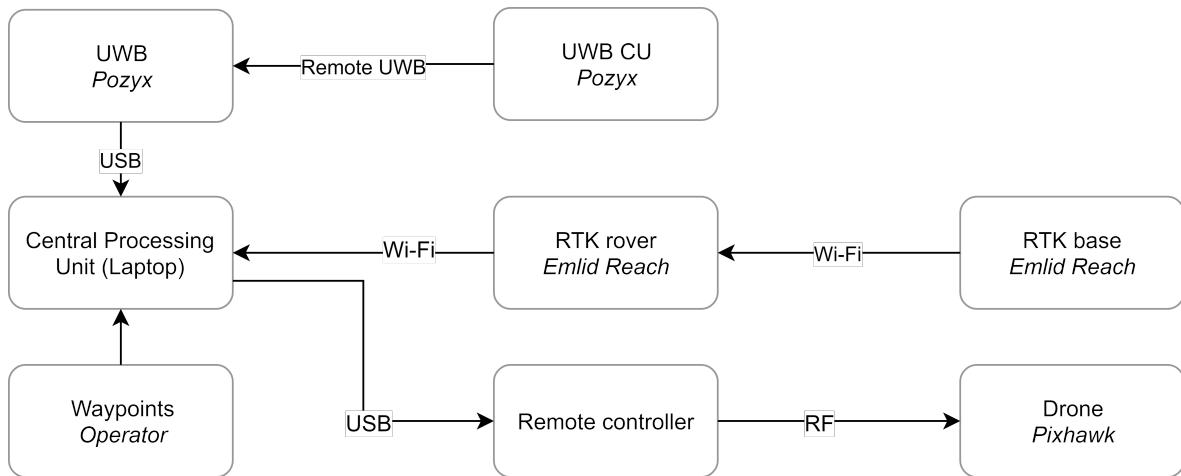


Figure 3.4: Final diagram for the connection with the different technologies

should be synchronized, which can be a problem if different transmitting technologies are used and the drone flies at a significant distance.

3.4 Accurate measurements - UWB ranging

To achieve an accurate calibration, as accurate distance measurements as possible should be achieved.

3.4.1 Measurement campaign

For a good procedure, it is important to perform reliable measurements and therefore to identify the different influences of parameters (such as orientation or position) on the measurement error. Therefore, an extensive measurements campaign is needed, covering all these different parameters. This will be discussed in Chapter 4.

3.4.2 NLOS/LOS identification

Extra errors are induced under Non-Line-of-Sight (NLOS) conditions (where there are obstructions present) than under Line-Of-Sight (LOS) (where there is clear sight). Therefore, it is important that there exist decision strategies to decide on the condition present. Different metrics will be approached in Chapter 5.

3.4.3 Error mitigation

In both conditions mentioned above, an error on the measured distance is present. Based on the measurements and the (Non)-Line-of-Sight condition, a machine learning solution is presented in Chapter 6 reducing the errors by a major factor. Since a simple metric or

combination of metrics to decide on the (Non)-Line-of-Sight condition in Chapter 5 can be imprecise, most of the presented machine learning solutions can easily be applied to perform a more complex classification based on an extensive amount of features.

3.5 Localization

After the collection of distance measurements and applying the corrections on these by the machine learning methods, a final position estimate of the anchor needs to be calculated based on the principle of trilateration. This can be achieved by a nonlinear least-squares localization, as can be found in Chapter 7. A weighting factor should be in this algorithm due to the different error variance on the distance estimates in LOS/NLOS conditions

3.6 Automated procedure

Next up is the actual procedure. Here, the goal was to create a fully automated calibration procedure, meaning that the drone should start and do everything without human interaction. Due to the impossibility to automatically control the drone, another course had to be taken. A first consequence is that a fully automated procedure could not be investigated and therefore, a drone procedure will be proposed where minimal human interaction is needed. The operator will have to set a few waypoints where the drone should fly after which the drone is able to perform the procedure by itself. The reason for this decision is that in this way, certain parts of the procedure could be tested without the need of a working drone. All of this will be discussed in Chapter 8.

Chapter 4

(Non)-Line-of-Sight measurements

When performing ranging measurements with the Pozyx, two important factors have to be considered. The first one is a bias occurring even in ideal measurements, where there are no obstructions (LOS). The second one is that the bias occurring in the other case (NLOS) is even greater and less predictable and as such NLOS measurements need to be identified. In this chapter, first the difference between LOS and NLOS will be explained as well as the bias occurring in both situations in Section 4.1. To be able to investigate on which parameters this bias depends, an extended measurement campaign was conducted. To ensure the bias does not come from mistakes in the campaign, also the accuracy of the measurement campaign should be taken into consideration. This is discussed in Section 4.2.

4.1 Line-of-Sight vs Non-Line-of-Sight

In this section, the difference between Line-of-Sight and Non-Line-of-Sight is explained. This is an important distinction with different characteristics and will be used throughout the following chapters.

4.1.1 Line-of-Sight

Line-of-Sight (LOS) propagation is a characteristic of electromagnetic radiation and means the waves travel in a direct path (DP) from the source to the receiver. For low frequency applications ($< 3MHz$), waves can follow the ground (which is for example the case with AM radio stations) or can be reflected by the ionosphere. However, for UWB, which is above this $3MHz$, these effects do not apply and any obstructions between transmitting and receiving antenna block or attenuate the signal. Analogical to how our vision is blocked with obstructions as the light can not cross these obstructions. In Figure 4.1, RX_1 is in LOS condition with respect to TX .

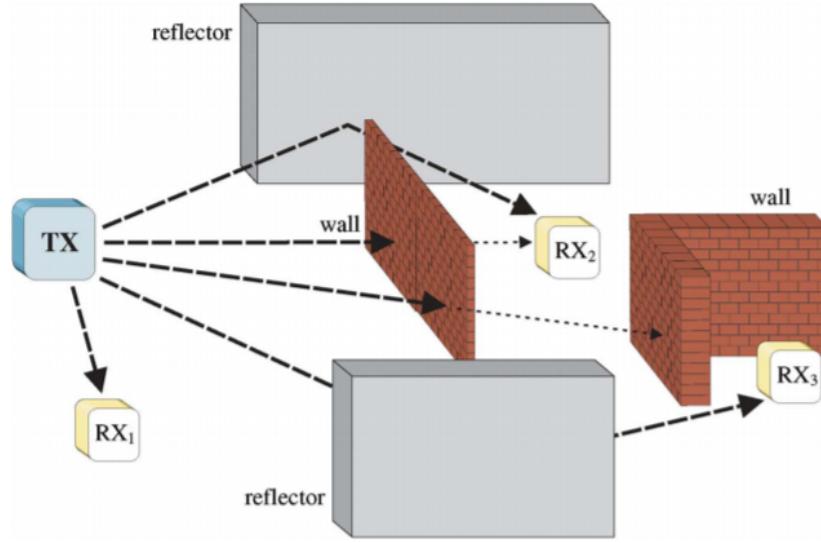


Figure 4.1: Different LOS or NLOS conditions from a transmitter (TX) to several receivers (RX_i) [32]

4.1.2 Non-Line-of-Sight

A Non-Line-of-Sight (NLOS) measurement occurs when the DP is either completely blocked (in which case the first arriving path comes from the reflected signal) or from DP excess delay (in which the signal traverses through different materials in a straight line resulting in additional TOF delays) [32]. In Figure 4.1 with respect to TX , RX_2 is an NLOS example where the DP is delayed and RX_3 an example where the DP is completely obstructed and only the reflections arrive.

4.1.3 Bias

LOS Even when measurements are performed under pure LOS conditions, the measured distance can be distributed around a value which does not correspond to the real distance, due to a bias acting:

$$d_{measured,LOS} = d_{actual} + d_{bias,LOS} + n_{thermal} \quad (4.1)$$

The reason for this bias is the formed discussed error by clock drift in TWR, see Section 2.2.

NLOS Additional to a bias acting in LOS conditions, in NLOS an extra bias occurs:

$$d_{measured,NLOS} = d_{actual} + d_{bias,LOS} + d_{bias,NLOS} + n_{thermal} \quad (4.2)$$

Here, $d_{bias,LOS}$ is the bias that would occur in LOS conditions and $d_{bias,NLOS}$ is an additional bias that occurs in NLOS conditions due both DP blockage (when the DP is completely

obstructed the receiver can only observe NLOS components, resulting in estimated distances larger than the true distance) and DP excess delay (when the DP has to traverse obstructions, the propagation time will be longer resulting in larger estimated distances). While $d_{bias,LOS}$ can be tried to corrected, $d_{bias,NLOS}$ depends heavily on environment conditions, type and widths of obstructions. All these are not known in advance and cause difficulties to correct the bias. The noise $n_{thermal}$ on the measurements can be assumed to be mainly thermal noise [34]. This noise has a Gaussian distribution, see Figure 4.2. To prove this, a one-sample Kolmogorov-Smirnov normality test was applied to all the measurements for the measured distance. Only in 4% of the cases, the null hypothesis that the distances come from a standard normal distribution were rejected against the alternative that they do not come from such a distribution with a significance level of 5%.

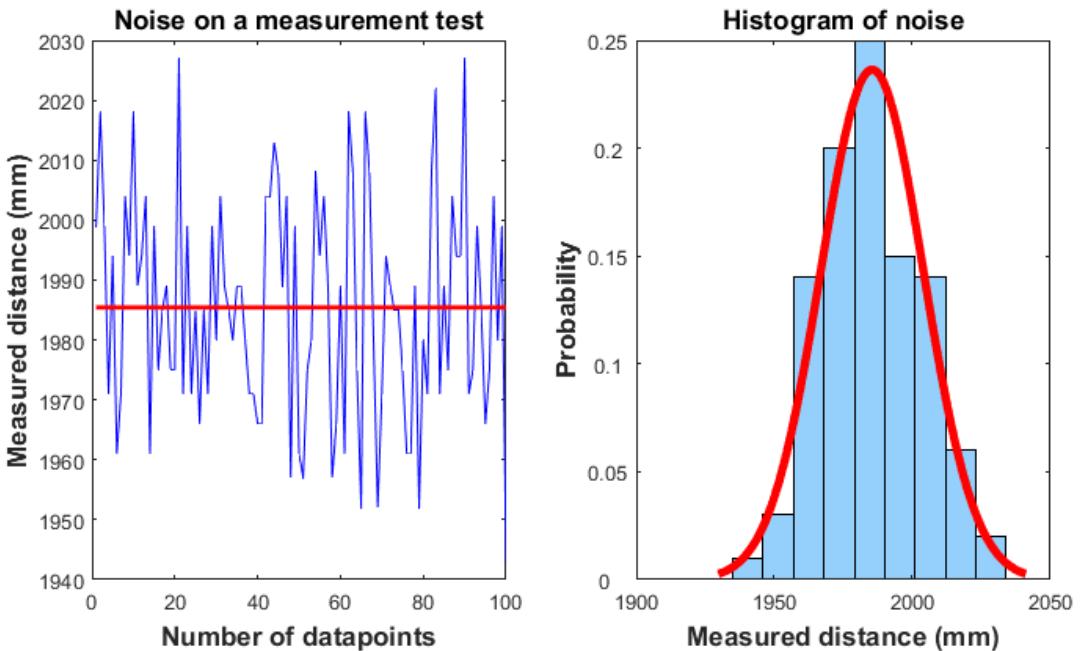


Figure 4.2: Thermal noise on UWB measurements

4.2 Measurement campaign

4.2.1 Measurement set-up

A measurement campaign was performed to investigate qualitatively and quantitatively the influence of different parameters on the quality of a UWB ranging measurement. Two Pozyx devices are placed on separate tripods. To have no influence from the computer attached to one of these devices, a third Pozyx device plugged into the computer was used to measure the data remotely. A python script was used to store this data in .txt-files, so this could be

processed later.

For the LOS groundtruth measurements, the Pozyx devices were put on a height of 1.5m and orientated face-to-face. All groundtruth distances were measured with a laser meter. The tests were performed in a room ($4 \times 20 \times 10 m^3$) exceeding the dimensions of the tests. This set-up makes sure that for every measurement the Fresnel zones stay unobstructed, guaranteeing the best quality for the signals.

Fresnel zones Fresnel zones are multiple 3D cylindrical ellipse shaped regions of space between transmitter and receiver, as seen in Figure 4.3, and give an insight of the signal strength. There are many Fresnel zones, but only the first 3 play an importance in signal strength, the first being the strongest.

Each Fresnel zone is defined by the phase-shift, which occurs when a transmitted signal (which departs from an angle from the Line-of-Sight line) reflects off an object within this zone and results in constructive or destructive interference at the receiver.

The formula for the maximum radius of the Fresnel zone is:

$$r = 17.32 \cdot \sqrt{\frac{d(km)}{4f(GHz)}} \quad (4.3)$$

For a distance of 20m and lowest frequency of 3.1GHz, the radius of the Fresnel would be 0.70m. For the distances in this thesis, which are smaller than 20m, the height of the set-up of 1.5m and open space make sure the Fresnel is unblocked. Given the limited time for the

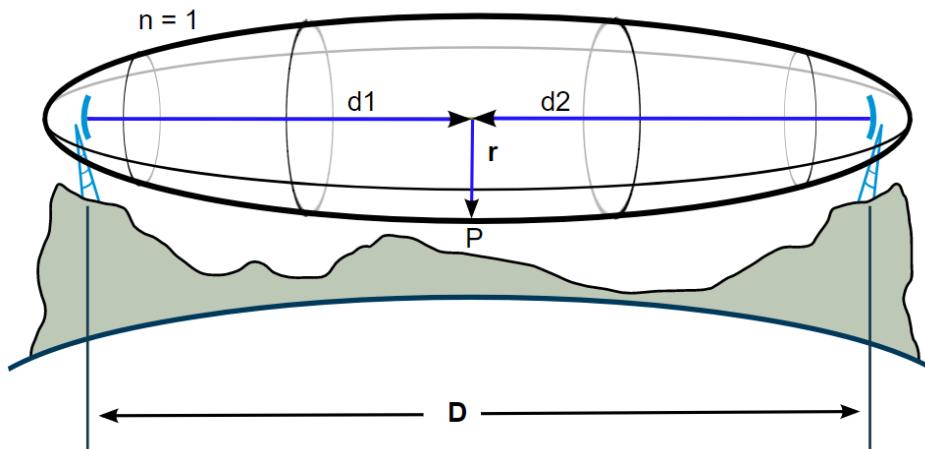


Figure 4.3: Fresnel zones

testing, not all effects could be investigated upon. The Pozyx devices offer to set parameters mainly having an influence on the ranging distance. For accurate measurements, the following

settings are advised: a bitrate of 110 kbit/s, a Pulse Repetition Frequency (PRF) of 64 MHz and a preamble rate of 4096. In all our measurements, these settings were applied.

4.2.2 Influences

Many factors have an influence in the quality of measurements and as a consequence in the bias occurring in different situations. In order to achieve an extensive measurement campaign, all the following influences were considered.



Figure 4.4: Measurement set-up for a Transmitter (TX) and Receiver (RX)

Horizontal distance

The horizontal distance is the distance between the parallel e_z -axes (ref. Figure 4.4) of the two devices. Since this parameter has a major influence, measurements ranging from 2m to 10m with intervals of 0.5m were taken, resulting in 19 states for this parameter.

Vertical distance

Next to the horizontal distance, the vertical distance (or height) was varied, which is the difference in e_z -coordinate from the e_xe_y -planes. Height differences from 0m to 2m were investigated in intervals of 0.5m, resulting in 5 possibilities for this parameter.

Z-orientation

Since the rotation with respect to the e_z -axis also has a proven influence [35], the rotation of the receiver was varied over a complete rotation with intervals of 22.5°, resulting in 16 states for this parameter.

Another test where next to the orientation of the receiver, also the orientation of the transmitter was varied, showed that the same patterns were seen. To reduce the time needed to

perform all measurements, it was chosen to fix the orientation of the transmitter and only variate the receiver orientation.

X or Y-orientation

The UWB antenna transmits best in the direction of the $e_x e_y$ plane (recommended by Pozyx as seen previous in Chapter 2), so it is advised to place anchors upwards (with the e_z -direction facing the sky) [30]. Since in this set-up neither rotation around e_x or e_y -axis is permitted, these rotations were not investigated in the measurement procedure.

Obstacles

For the NLOS identification, the same previous parameters (distance, orientation) were varied in environments where obstructions were present. Many different walls with varying thickness and materials were investigated.

4.2.3 Accuracy of the set-up

Line-of-Sight The real distance between the two devices was measured with use of a laser distance meter, the Fluke 414D. This meter promises an accuracy up to $3mm$. As the measurements were performed by hand, vibrations by the hand and such have to be incorporated. Following Figure 4.5, a slight vibration of the hand could result in measured distance m instead of the real distance d . For an example, let us look at the case where the devices are $d = 3m$ apart. If one performs a distance measurement of distance and has a slight vibration, one might end up maximum $e = 3cm$ next to the UWB chip on the Pozyx device (due to geometry of the Pozyx).

$$m = \sqrt{d^2 + e^2} = \sqrt{3000^2 + 30^2} = 3000,15mm \quad (4.4)$$

As this deviation is small and only has an influence in low distances, where the vibrations of the hand are lower, this does not concern us and can be assumed to be maximum $1mm$. Combined with the distance accuracy of the laser meter, this totals to an accuracy of $4mm$.

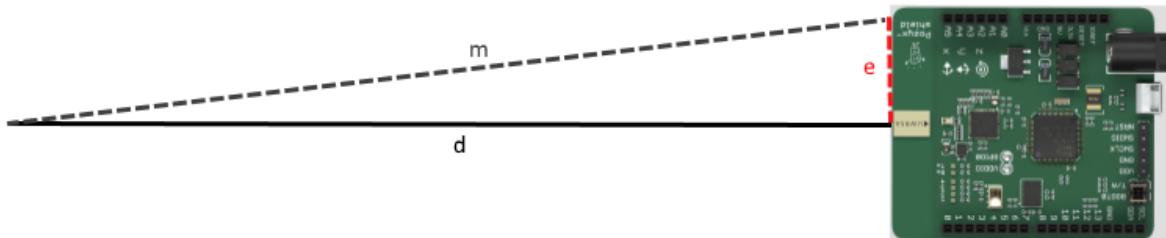


Figure 4.5: Measurement error with use of Laser Meter

Non-Line-of-Sight When performing measurements in NLOS conditions (where the sight between the Pozyx devices was obstructed and the laser meter could not be pointed directly from one to another), the width of the wall was measured and points adjacent on the wall were pinned. Then, the same procedure was performed, now measuring the distance from Pozyx devices to their pinpoint on the wall, therefore inducing twice the laser meter error. As the estimate of the wall can be assumed to be accurate to 5mm and combined with the vibration error of 1mm , the total inaccuracy for NLOS can be estimated $1\text{mm} + 2 \cdot 3\text{mm} + 5\text{mm} = 12\text{mm}$.

Chapter 5

(Non)-Line-of-Sight features

When performing ranging measurements, an important distinction has to be made whether the measurements are performed in Line-of-Sight (LOS) or Non-Line-of-Sight (NLOS). In this chapter, many identification techniques proposed in literature are explained and applied. It will be shown that the quality of the methods varies on environment and test case conditions and consequently, in some cases one method performs better than in others. Therefore, it is proposed to use all the identification techniques as features (or parameters) in a machine learning approach, which will be discussed in the next chapter.

5.1 Identification via RSS

In [36], [37], [39], an identification technique is proposed based on the Received Signal Strength (RSS). The basic idea is that in general, for the same distance, the RSS in LOS conditions can be over a hundred times stronger than the RSS in NLOS conditions. As this is the reason why the RSS is not a good parameter to base the ranging on, it can be a parameter to decide if a measurement is performed under LOS or NLOS conditions. In Figure 5.1, an example from [36] for the RSS for LOS and NLOS measurements is shown, comprising different situations, where the difference between LOS and NLOS is clearly visible. These findings were investigated by performing both LOS as NLOS ranging under different variables (such as distance, orientation of the receiver) and a scatter plot with all data was generated, as seen in Figure 5.2. From this figure, it is clear that the RSS can be a good parameter for the identification, however there should be taken care at low distances. As seen on both figures, at low distances, the RSS values tend to overlap and the distinction between LOS and NLOS becomes less clear. For distances larger than $3m$, the distinction becomes more and more clear. This is because the signal strength decreases quadratically with the distance in LOS conditions but to a power of 4 in NLOS conditions. Only measurements up to $10m$ were taken, since this is a realistic final distance for the drone. For each measurement, the actual distance measured with a laser meter and the reported distance and RSS from the Pozyx were gathered.

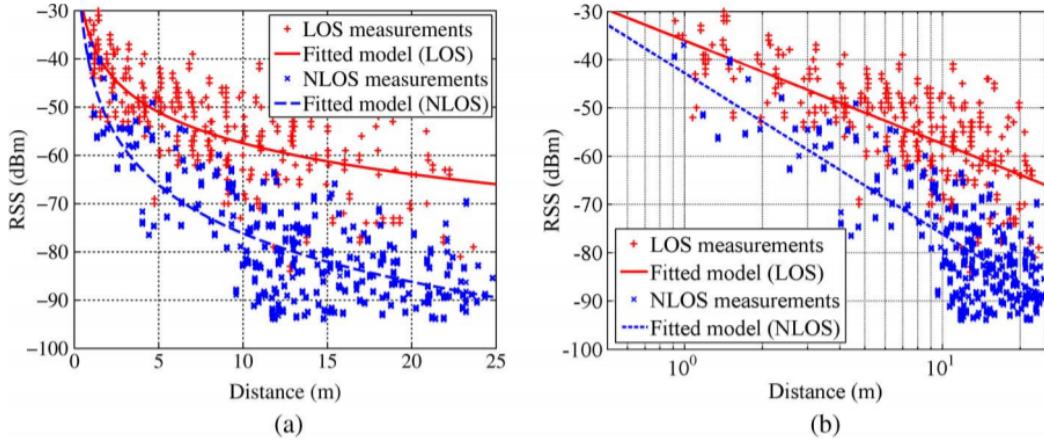


Figure 5.1: LOS and NLOS measurements and their least square approximation propagation model: (a) Linear (b) Log-distance [36]

Trying to find a quick and easy way to identify an NLOS situation, all measurements taken on the same actual distance were taken together in a set, independent from orientations or their reported distance. For each set, it is then possible to calculate the expected RSS value for 90% of the LOS measurements. Based on the expected RSS value of these measurements, a linear fit was created to obtain an expected RSS value for any distance. An example for the created fit based on these threshold is visible in Figure 5.2.

This function for the expected minimum RSS value for an LOS measurement given the measured distance can then be used as a threshold to decide on LOS vs NLOS. If the expected RSS from the fit for the measured distance is higher than the effective measured RSS, it can be classified as NLOS. If the expected RSS for the measured distance is lower than the effective RSS, it can be classified as LOS.

For a measurement set-up where only the distances and orientations of the receiver where varied (the height was only varied for part of the measurements and will thus not be used in the method), this method was applied, resulting in the threshold-function as displayed in Figure 5.2 following the equation:

$$RSS_{threshold}(d) = -80.3977 dBm - 0.84708 \frac{dBm}{m} \cdot d_{actual} \quad (5.1)$$

Next, this threshold equation was tested on the data using the following decision strategy:

$$\begin{aligned} RSS_{measured} > RSS_{threshold}(d_{measured}) &\Rightarrow LOS \\ RSS_{measured} \leq RSS_{threshold}(d_{measured}) &\Rightarrow NLOS \end{aligned} \quad (5.2)$$

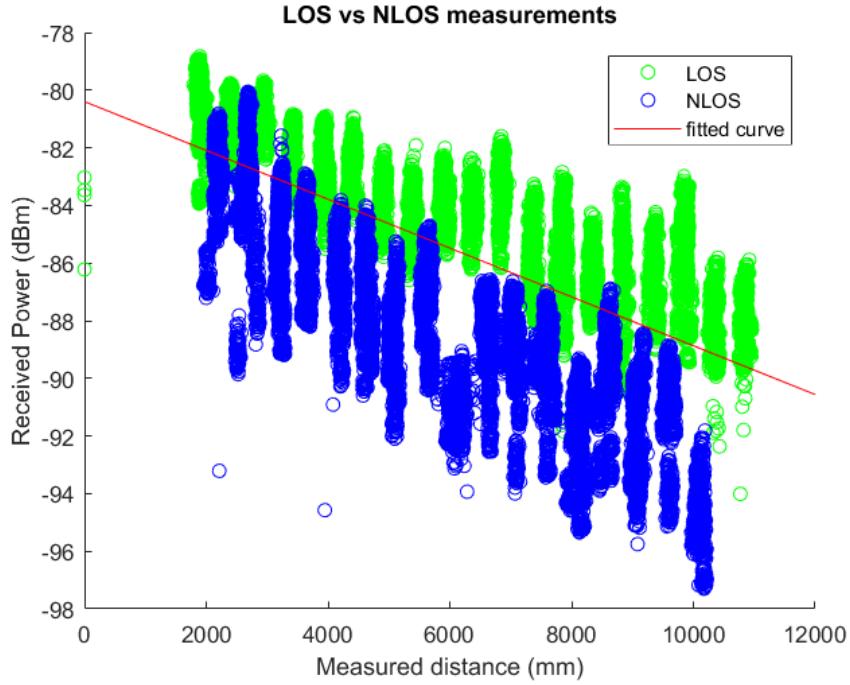


Figure 5.2: Scatter plot of LOS/NLOS measurements and the linear fit for the thresholds

A priori, for each dataset it was known if the measurements were performed under LOS or NLOS. The results of these tests are summarized in Table 5.1, where in the last column the percentage of right decisions is shown. The first two have the condition "same height", which means that for these measurements, the height was not varied. The condition "all heights" means this data also includes measurements where the Pozyx devices were not on the same height, but still the same thresholds as before are used.

From these results, it can be seen that in 90.42% of the LOS measurements where the devices were on the same height (#1), they were correctly identified as LOS. This is of course to be expected, as the threshold was created based on this data. For the NLOS measurements under the same conditions (#2), 94.08% were correctly identified.

Looking at the conditions where all height differences are considered, the identification for LOS measurements becomes worse, lowering to 77.68%. This is due the lesser ideal conditions were the height difference introduces lower signal strengths. However, also the NLOS measurements will have lower signal strengths, resulting in an ever better NLOS detection.

To conclude, this method gives a high performance for identifying NLOS measurements. LOS measurements have a higher probability to be wrongly identified as NLOS measurements than the other way around, which is good for our final localization. Due to their

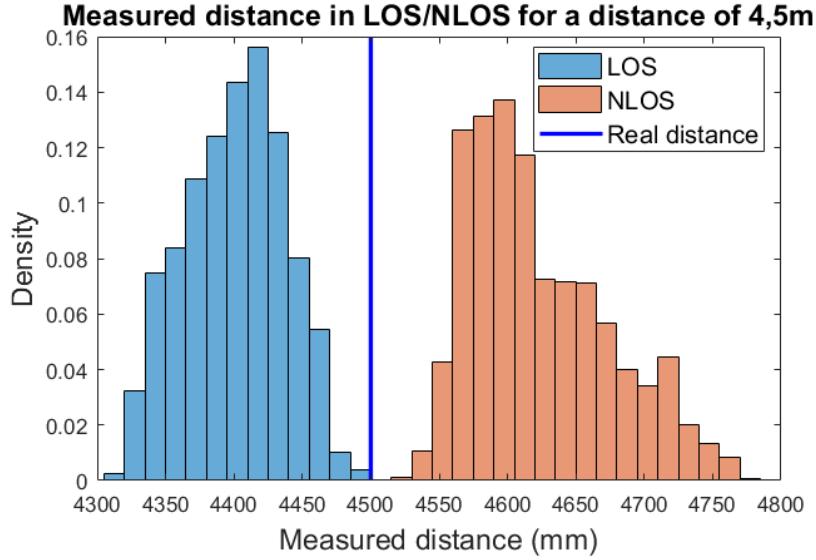


Figure 5.3: Distribution of measured distance for LOS/NLOS measurements at an actual distance of 4.5m

unreliability, NLOS measurements will be given a smaller importance in the final algorithm. As such, in a case of wrong identification, it is worse to assign a higher importance to a less reliable NLOS measurement than it is to assign a smaller importance to an LOS measurement.

Table 5.1: Results of RSS identification on different datasets

#	(N)LOS	Conditions	Total points	Right decision	Wrong decision	Perc. (%)
1	LOS	Same height	19 417	17 557	1 860	90.42
2	NLOS	Same height	24 518	23 067	1 451	94.08
3	LOS	All heights	42 940	33 356	9 584	77.68
4	NLOS	All heights	31 568	30 109	1 459	95.38

It should be noted that this method ignores the fact that, at a given actual distance, the measured distance in LOS will differ from the measured distance in NLOS. An example of this is visible in Figure 5.3, where the LOS measurements are centered around a distance less than the actual distance and the NLOS measurements around a distance higher than the actual distance. As this method is based on defining the thresholds on basis of the actual distance instead of the measured distance, this could introduce additional errors. However, given that the results prove this method works in its current form, this is not a problem.

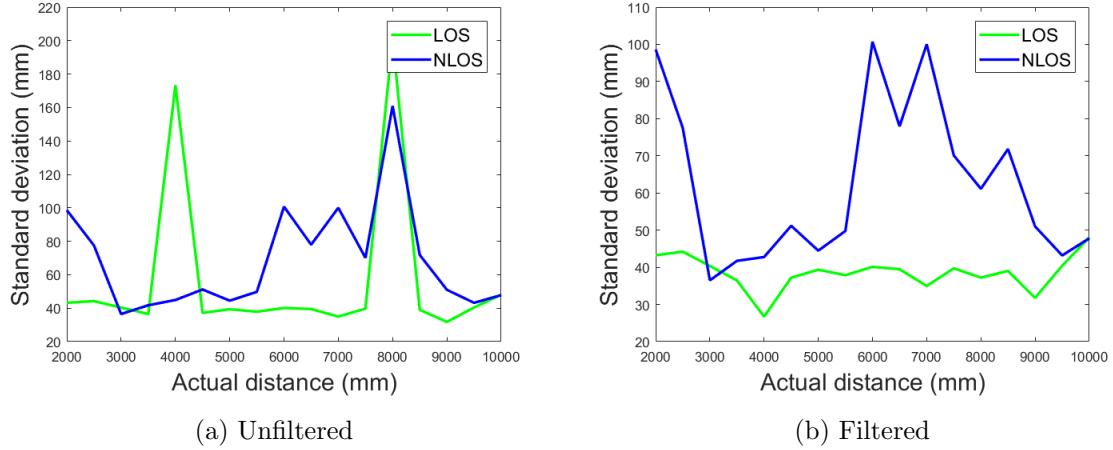


Figure 5.4: Standard deviation of measured distance for LOS and NLOS measurements

5.2 Identification via standard deviation

In [40], an identification technique is proposed relying on the comparison of the standard deviation between a sample statistic and the known standard deviation of that statistic in LOS conditions. Based on measurements, it was seen that the presence of NLOS conditions increase the standard deviation of the measured distance in a significant manner.

For the same dataset as the previous section, the standard deviation on the measured distances was calculated and presented visually in Figure 5.4a. At first, one could say there is no clear distinction. However, the LOS peaks at resp. 4m and 8m could result from a measurement error on these distances, as the other LOS standard deviations appear to stay constant.

Looking at the results from the tests, it was seen that due to an unknown error the measured distance provided by the Pozyx was equal to 0mm, resulting in high standard deviations for these measurements. Filtering out these values, the values in Figure 5.4b were obtained. Apart from one case, the values for the standard deviation for NLOS cases are always above LOS cases. The reason why the NLOS curve drops below the LOS curve for this case could be that for this specific distance, the measurements in the NLOS case were better than the LOS case.

Based on this data, one could propose a threshold of for example 50mm to decide if a measurement is performed under NLOS conditions. However, for this method to work, many measurements must be disposable at the same location to calculate a reliable standard deviation.

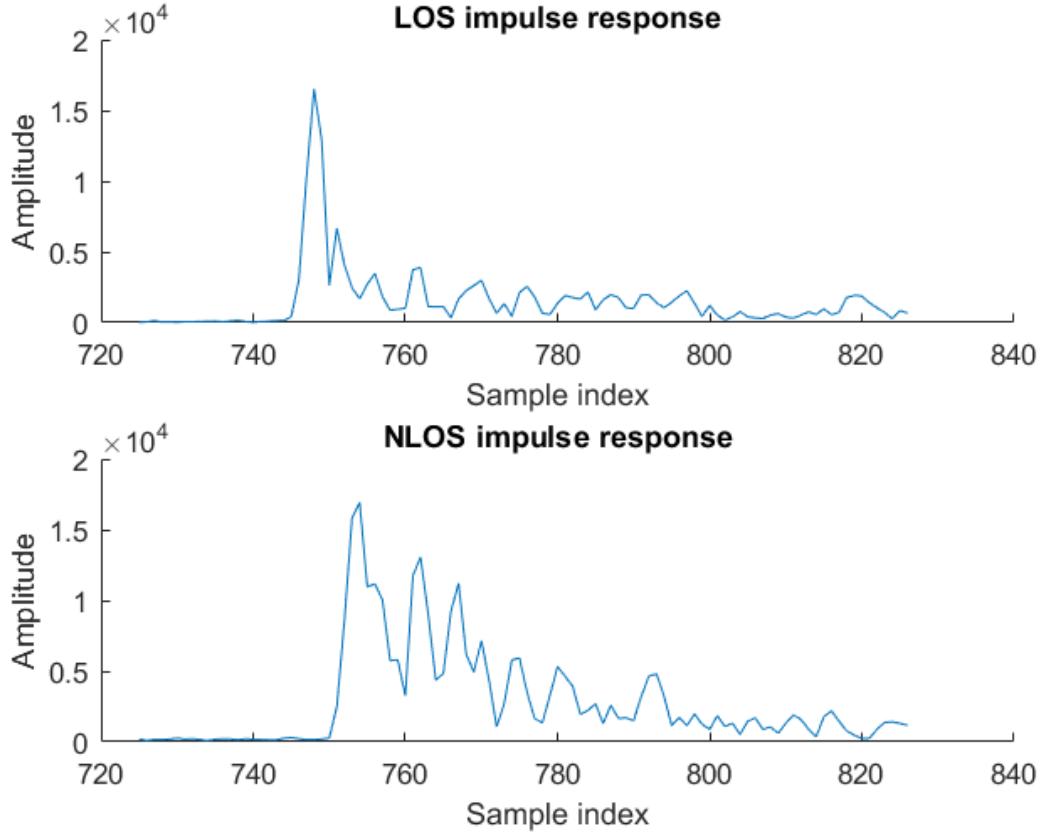


Figure 5.5: Zoomed in response

5.3 Identification via CIR time series

Other widely used techniques focus on the shape of the Channel Impulse Response [38], [41]. Typically, from the shape of the received signal (or channel impulse response), it is possible to identify an LOS or NLOS measurement. As an example, a typical response for both of these is shown in Figure 5.5. While an LOS situation returns one sharp peak (the direct path), an NLOS situation typically has more peaks (due to the reflections) and less sharp peaks (more spread out). These characteristics are visible in the time series of the CIR and will be the basis for the following multipath channel statistics (MPCs). A histogram for each of the following statistics from [41] is shown in Figure 5.6.

5.3.1 Kurtosis

The first characteristic is the kurtosis, defined as the 'tailedness' or 'peakyness' of the signal. In other words, for LOS where there is only one sharp peak, one expects a high kurtosis. For NLOS, where there are multiple more broader spread peaks, the kurtosis will be lower.

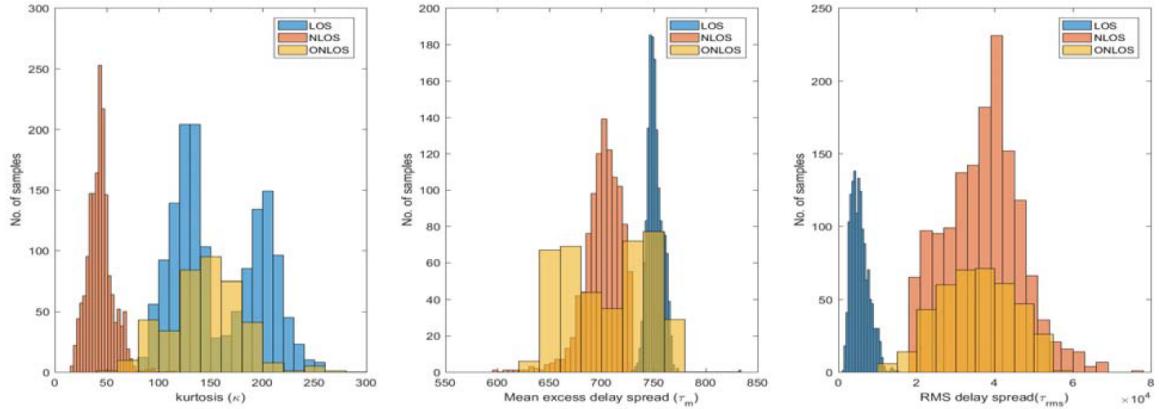


Figure 5.6: Multipath Channel statistics [41]

The kurtosis is defined as the fourth standardized moment:

$$\kappa = \left(\frac{X - \mu}{\sigma} \right)^4 = \frac{\mu_4}{\sigma^4} \quad (5.3)$$

Where μ represents the mean and σ the standard deviation of the CIR.

Testing this method on the datasets, it can be seen that at some distances this method had worked perfectly, while at others failed completely. As an example, for all data captured at a mutual distance of $7.5m$, the kurtosis values follow the expected distribution as seen in Figure 5.7b cfr. LOS-NLOS in Figure 5.6. Values for NLOS are distributed around a small kurtosis, while the values for LOS are distributed around a higher kurtosis level. From this figure, it would be possible to construct a threshold upon which to decide whether a given kurtosis value corresponds to LOS or not with an predefined uncertainty.

However, looking at Figure 5.7a, there is an overlap between the LOS and NLOS values, which is similar to the overlap between LOS-ONLOS in Figure 5.6. This ONLOS (or VNLOS) is short for Optical/Visual-NLOS and is a term created in [41] as an extra category next to LOS and NLOS. Visual-NLOS means that the visual signal is blocked (by an obstruction) but the direct path is not. This means that the path which the signal would follow if there was no obstruction, is still the path the signal follows with the obstruction, but with a small delay. Therefore, the signal has a similar shape as an LOS measurement, while still introducing an extra positive bias. As the paper states, the kurtosis is a good measure to filter out NLOS from LOS, but not VNLOS from LOS.

5.3.2 Mean excess delay

Another commonly used statistic is the mean excess delay (MED). It provides information about the time delay for which the energy contained in the multipath falls below a certain

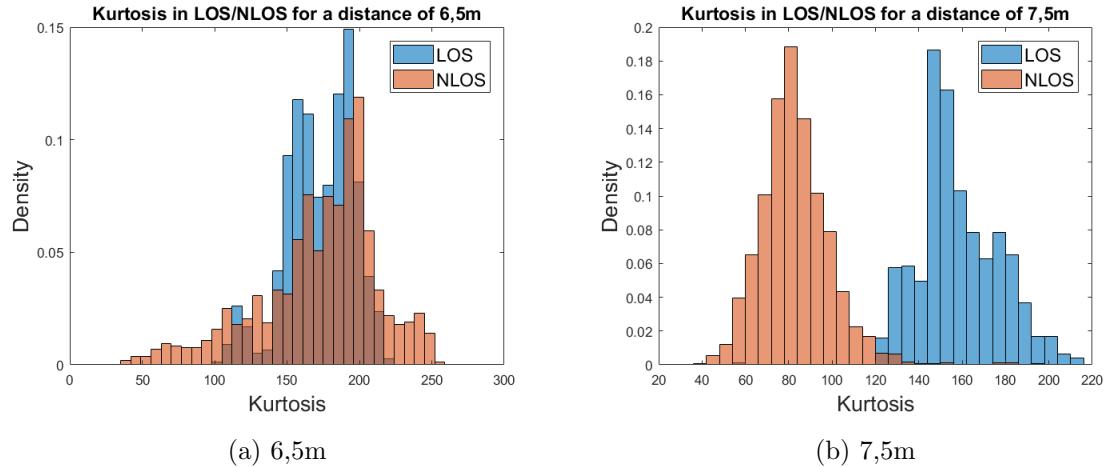


Figure 5.7: Normalized histogram of kurtosis values for LOS and NLOS situations

level and is defined by the first moment of the power delay profile of the channel:

$$\tau_m = \frac{\int_0^\infty t|h(t)|^2 dt}{\int_0^\infty |h(t)|^2 dt} \quad (5.4)$$

where the denominator of this equation represents the total energy comprised in the CIR $h(t)$. As this statistic is also highly dependent on multipath components, it is difficult to differentiate between VNLOS and LOS measurements, for the same reason as above, as can be seen in the middle figure in Figure 5.6.

5.3.3 Root mean square delay spread

The RMS Delay Spread characterizes the richness of multipath in the response and is defined as the square root of the second central moment of the power delay profile:

$$\tau_{rms} = \sqrt{\frac{\int_0^{\infty} (t - \tau_m)^2 |h(t)|^2 dt}{\int_0^{\infty} |h(t)|^2 dt}} \quad (5.5)$$

Based on the last figure of 5.6, it seems the best statistic to differentiate between LOS and both NLOS and VNLOS.

Put into practice, the statistic performs worse than expected, as is clear in Figure 5.8. Here, the RMS Delay Spread is squared to provide an equal axis as in Figure 5.6. While the LOS delay spread follows the expectations and the center is placed more to the right, the NLOS delay spread completely overlaps with it. After investigation, it was seen that the overlap primarily happens at low distances. This is because at lower distances, less opportunities for multipath are present. If only data starting from distances larger than $6,5m$ is taken, virtually no overlap is present in the distributions, as seen in Figure 5.9. This is because the

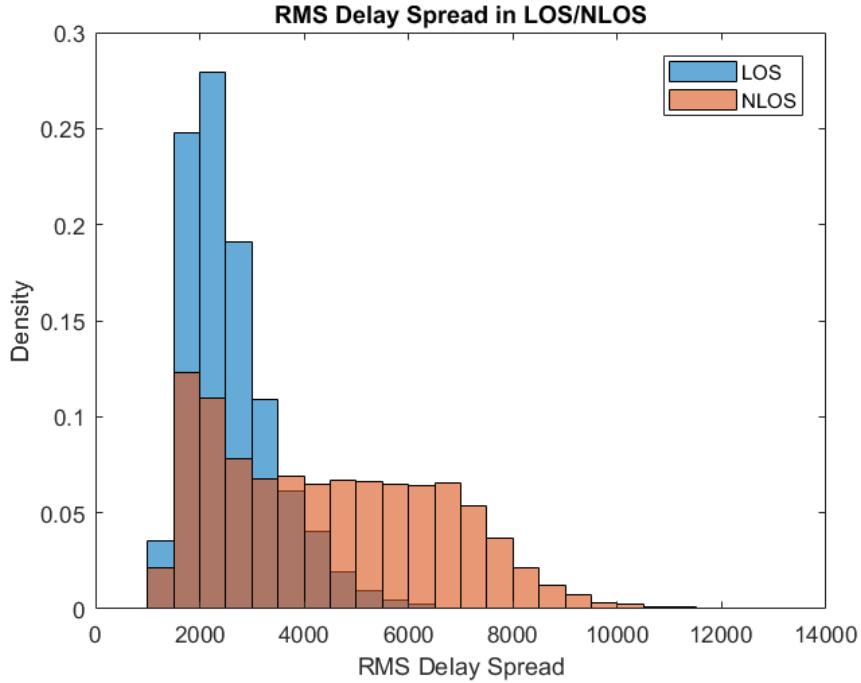


Figure 5.8: Normalized histogram of squared RMS Delay Spread

center of the RMS Delay Spread distribution stays the same for larger distances in LOS, but moves further in NLOS.

Following the same procedure as for the RSS, a threshold was determined from which an NLOS measurement would be identified. This was done by putting a threshold on the lowest 10% of the NLOS RMS Delay Spread values, to be able to identify 90% of the NLOS cases. With this obtained squared threshold of 4721.9 or a RMS Delay Spread threshold of 68.7, none of the LOS test cases was wrongly identified as an NLOS case. Therefore, it can be suggested to move the threshold to identify 95% of the NLOS cases. Now, the squared value for the threshold of 4430 identifies only 0.11% of the LOS measurements as NLOS. This is an acceptable increase, as now 5% more NLOS measurements should be rightly identified.

The disadvantage with these MPC statistics is that they usually require a longer computation time and are based on the probability models. In our case, they only work for large enough distances, which can be a problem if the drone only is able to fly at low distances. Therefore, a technique based on the power metric is proposed, as discussed in the next section.

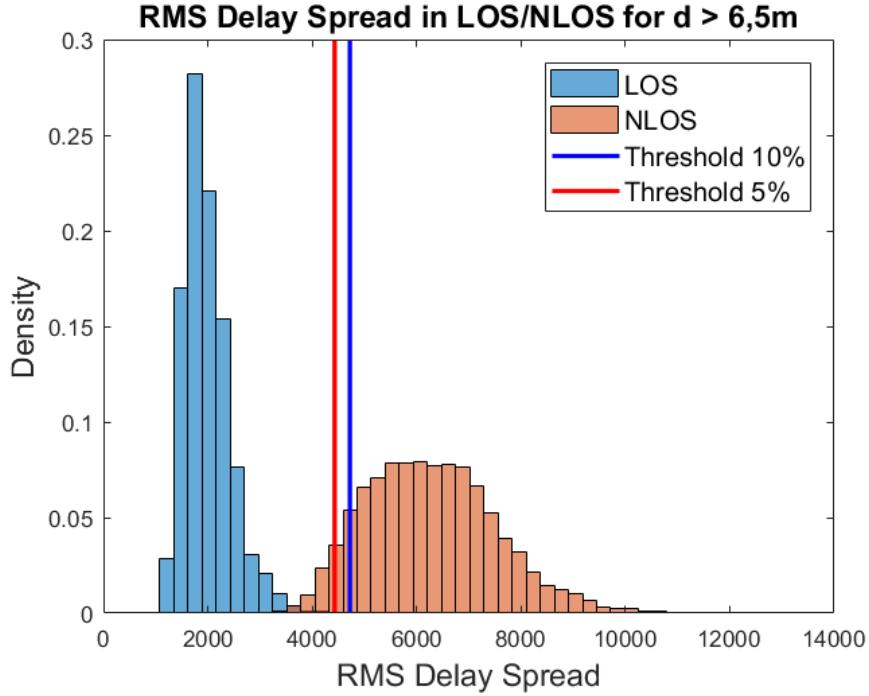


Figure 5.9: Normalized histogram of squared RMS Delay Spread where the distance is higher than 6.5m

5.3.4 First path vs. Total power

In this section, an approach is presented which uses the difference between the power contained in the whole signal and the power contained in the first arriving path [42]. The method can be used in real-time and does not need environment information.

First Path Power (FP) The First Path power is the estimated power contained in the first arriving path of the signal, it can be calculated with the following formula using the Channel Diagnostics from the DW1000 registers [29]:

$$P_{FP} = 10 \log_{10}\left(\frac{F_1^2 + F_2^2 + F_3^2}{N^2}\right) - A \quad (5.6)$$

Where F_i signifies the 1st, 2nd or 3rd amplitude around the first path, N is the preamble accumulation count value and A is the value for the Pulse Repetition Frequency, in this case 121.74.

Received Power (RX)

$$P_{RX} = 10 \log_{10}\left(\frac{C \cdot 2^{17}}{N^2}\right) - A \quad (5.7)$$

Where C is the Channel Impulse Response and N and A as formerly stated.

According to [41], as soon as the power difference $P_{RX} - P_{FP}$ is greater than $6dBm$, the measurement can be viewed as an NLOS measurement. Applied to our dataset, this method showed no usable results, since (ref. Figure 5.10) the distribution for LOS and NLOS overlap. Namely, 19% of LOS cases have a $P_{RX} - F_{FP} \leq 6$ and 85% of NLOS cases have a $P_{RX} - F_{FP} \geq 6$.

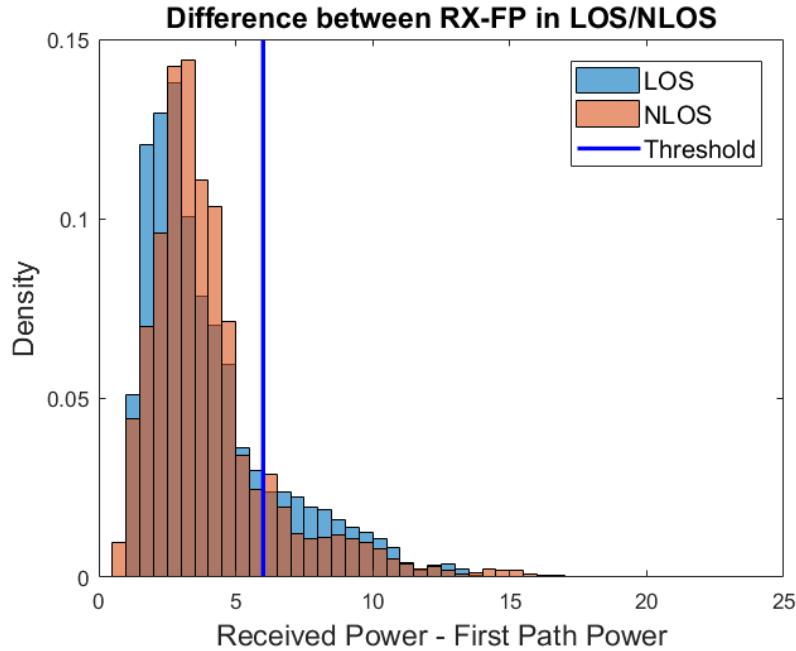


Figure 5.10: Normalized histogram of $P_{RX} - F_{FP}$: full overlap

Attempts to limit the dataset to high or low distances, which was successful for the RMS delay spread, still resulted in these observations.

5.4 Validation

Based on the dataset used in the former sections, two techniques have a high reliability based on a single measurement, namely the RSS and RMS delay spread method. If many samples at the same location are available, also the standard deviation on the measured distance could give an indication.

To validate this, the former proposed identification methods were used on new datasets in different conditions and on different days, to see how robust they are. In the results of Table 5.2 and 5.3, the number of tests and accompanying measurements are shown, followed by the features applied to the measurements. In the column RMSDS*, the RMSDS method is applied only to distances exceeding $6,5m$. If no such distances were in the dataset, an X is shown.

LOS For LOS, the different identification methods and their percentage of right identification is shown in 5.2, combining 28869 usable measurements after filtering. The RSS performs as expected, except for test 3, where only half of the measurements are identified correctly. Contrary to former conclusions, the RMSDS does not necessarily behave better if only distances greater than $6.5m$ are used (RMSDS*). The method on standard deviation is always correct for 90% of the measurements. In some cases, the First Path method performs very good, in others not.

Table 5.2: Results of identification on unseen LOS measurements

Set	meas.	RSS (%)	Kurtosis (%)	RMSDS (%)	RMSDS* (%)	Std. (%)	FP (%)
1	4617	82.6	22.9	89.2	X	100	98.8
2	9682	89.7	60.5	96.3	98.1	99	62.3
3	4597	51.3	66.4	62.8	X	98	92.2
4	9973	97.3	69.8	97.6	93.3	91	41

NLOS The same was done for NLOS, combining 17544 measurements in Table 5.3. Here, also the RSS is a good indicator, but contrary to former results, the standard deviation does not perform that well. In one case, the RMS delay spread gives a very good identification, in the other a very bad.

Table 5.3: Results of identification on unseen NLOS measurements

Set	meas.	RSS (%)	Kurtosis (%)	RMSDS (%)	RMSDS* (%)	Std. (%)	FP (%)
5	5177	89.3	84.3	24.2	38.3	53.4	57.2
6	12367	99.9	38.2	96.6	100	12	49.3

Conclusion From the above, it is clear that not one single feature is accurate enough to identify NLOS measurements right, since sometimes one parameter and sometimes another performs best. Therefore, a weighted decision could be proposed as follows:

$$p_{NLOS} = \frac{W_{RSS} \cdot p_{RSS} + W_{RMSDS} \cdot p_{RMSDS} + \dots}{W_{RSS} + W_{RMSDS} + \dots} \quad (5.8)$$

Here, p_x signifies the identification of feature x , and W_x the weight given to this feature.

However, the tuning of these weights is a difficult task and highly dependent on environment conditions. Next to this, possible complex correlations between the features are difficult to identify. Therefore, it is proposed to use all features discussed in this chapter in a machine learning approach, as handled in the following chapter.

Chapter 6

Machine learning approach

As seen in the previous chapter, various features can be good indicators for the presence of obstacles and multipath environments in the UWB ranging. The selection of only one feature of this set to robustly indicate NLOS/LOS is inadequate taking into account all possible situations, see Section 5.4. A solution to this is an identification based on a combination of these features. However, these features are correlated and have hidden complex patterns. Next to this, the importance (weight) given to each separate feature needs to be wisely chosen. This weight tuning is time-consuming and complex.

This high complexity is also found in the error mitigation methods. These methods try to minimize the ranging error/bias in order to have accurate final distance estimates. To recall, the main reason for the identification of LOS/NLOS is to know whether the measurements contain an extra bias induced due to NLOS situations. Two separate mitigation methods will thus be considered. A method correcting biases in LOS situations and a method for the correction of biases in NLOS situations. Since the LOS bias is less complex than the bias in the NLOS case, the LOS mitigation will be more accurate. In order to deal with these very complex relations, various machine learning methods are presented and applied on the identification of Line-of-Sight or Non-Line-of-Sight and based on this identification the correction of the ranging error. At the end of the chapter, these methods are evaluated and compared for each separate problem.

6.1 Introduction

To deal with all these very complex correlations in a high dimensional feature space, supervised machine learning methods are brought forward. In supervised machine learning, a model is built/learned by data without explicit programming. This data consists of samples containing an input and output. The input X consists of M amount of samples, each defined as a vector with N amount of features. Mathematically, this is denoted as $X \in \mathbb{R}^{MXN}$, with

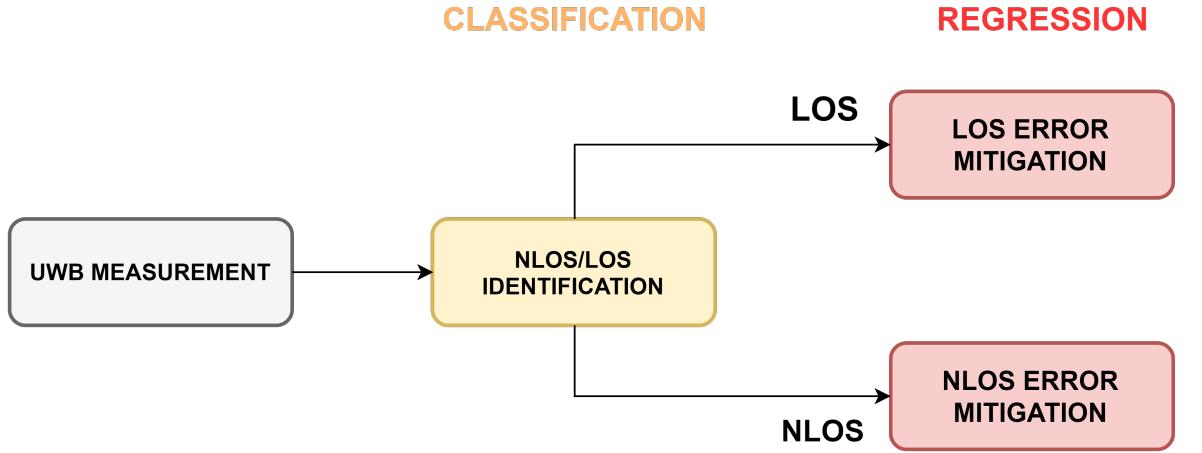


Figure 6.1: Machine learning approach

for the k -th sample $\mathbf{x}_k = \{x_1, x_2, \dots, x_N\}$. The output variables, which we want to predict as accurate as possible, will be described as $Y = \{y_1, y_2, \dots, y_M\}$. The goal is to learn a function/hypothesis $h : X \rightarrow Y$ so that $h(X)$ is a good predictor for the corresponding output value Y . This hypothesis can then be used to predict NLOS/LOS or the distance bias.

The features that will be considered in these complex problems, based on previous Chapter 5 are the measured uncorrected distance, the orientation of the CU, the height difference, the received power, the first path power, the mean excess delay, the root mean square delay spread, the kurtosis and the standard deviation of the noise. For ease, these will be denoted respectively as Distance, Orientation, Height, RX, FP, MED, RMSDS, Kurtosis and Std.

The various machine learning methods that will be considered can be used for classification and/or regression problems. In a classification problem, the hypothesis $h(X)$ will predict a discrete class label. The identification of LOS or NLOS is a typical binary classification problem, which for example can be noted as:

$$h(\mathbf{x}_k) = \begin{cases} 1 & LOS \\ -1 & NLOS \end{cases} \quad (6.1)$$

The other class of machine learning methods are used in regression problems. In these problems, a continuous variable needs to be predicted. The error mitigation methods are regression methods and the predictor will be noted as:

$$h(\mathbf{x}_k) = d_{bias,estimated} \in \mathbb{R} \quad (6.2)$$

This predictor will try to estimate the bias, $d_{bias,estimated}$ as close to the real occurring bias

$d_{bias,exact}$, denoted as:

$$d_{bias,exact} = d_{measured} - d_{actual} \quad (6.3)$$

Where $d_{measured}$ is the distance reported by the uncorrected TWR operation and d_{actual} the actual distance.

In the next sections of this chapter, data preprocessing and splitting is explained, the principles of the various machine learning methods are presented and finally, the various methods are compared for the NLOS/LOS identification problem and the error range mitigation.

6.2 Data handling & evaluation

Machine learning methods are heavily dependent on data. The handling of the data is thus a crucial part in all methods.

6.2.1 Preprocessing

After all measurement data has been collected from the extensive measurement campaign, see Chapter 4, the data was preprocessed. Firstly, this means formatting the data and the removal and fixing of missing data. Secondly, to avoid overfitting, the data is transformed. Overfitting is the production of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably [43]. Since noise is the main problem of overfitting, the dataset is narrowed down to the mean of all the datapoints of each specific measurement for every feature. Doing this can be justified by the fact that the noise on each measurement is mainly thermal noise. This noise has a Gaussian distribution with mean 0, see Chapter 4.

Lastly, the data will be rescaled. Since the range of values for each feature can vary highly in magnitude, some machine learning algorithms will not perform properly. For example in algorithms using euclidean distances between two points, features with greater magnitude will have a bigger contribution even though they might not be that important. Various rescaling methods exist, for which standardization is the most popular. By standardization, each feature is centered around 0 with a standard deviation of 1. This is done by calculating the distribution mean and standard deviation for each feature. This is calculated by:

$$x' = \frac{x - \bar{x}}{\sigma} \quad (6.4)$$

Where x is the original feature vector, \bar{x} is the mean of that feature vector and σ is its standard deviation.

6.2.2 Data splitting

After preprocessing, a next fundamental step is the splitting of the data, this is done in order to learn and tune the hypothesis $h(x)$ and evaluate its performance. The data is split up in a training set containing training and validation data and a test set containing test samples. The training samples are used to train the model, the validation samples are used to tune the parameters of the model and the test samples will be used only at the very last to evaluate the performance of the tuned model [44], see Figure 6.2.

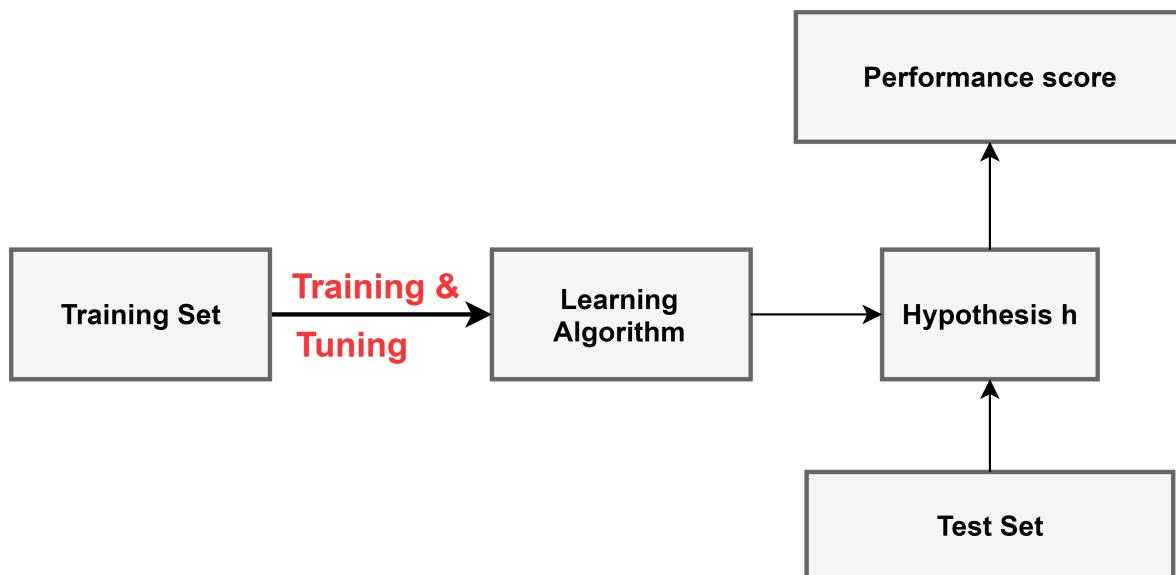


Figure 6.2: Supervised learning procedure

A very important note on this is that the test samples should be taken completely independent of the training set. This means that the test samples are taken in other settings (different environments for LOS, other walls for NLOS) than the samples taken for training and validation. This is to prevent wrong conclusions on the performance measurements. For example, a model that was heavily overfitted on measurements with a wall with a certain thickness will show good performance when tested with measurements taken at that same wall, but will perform badly in all other cases.

The main reason tuning of the model is done is due the trade off problem between the bias and variance error.

- The bias is an error from erroneous assumptions in the learning algorithm. A high bias can cause an algorithm to miss the relevant relations between features and target outputs (underfitting) [45]. To prevent confusion: this bias is different than the bias on the distance measurement d_{meas} .

- The variance is an error from sensitivity to small fluctuations in the training set. A high variance can cause an algorithm to model the random noise in the training data, rather than the intended outputs (overfitting) [45].

A clear example of this over- and underfitting can be seen in Figure 6.3.

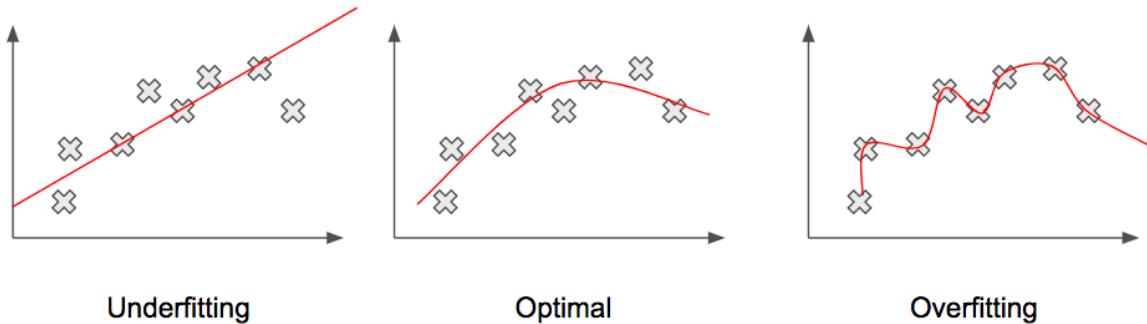


Figure 6.3: Simple example of overfitting and underfitting

Logically, one would like to minimize both errors, however this can not be done independently. A clear trade-off can be seen in Figure 6.4. The optimum model is where the total error, also called the generalization error, is minimum. In order to find this optimum point, the model needs to be fine tuned.

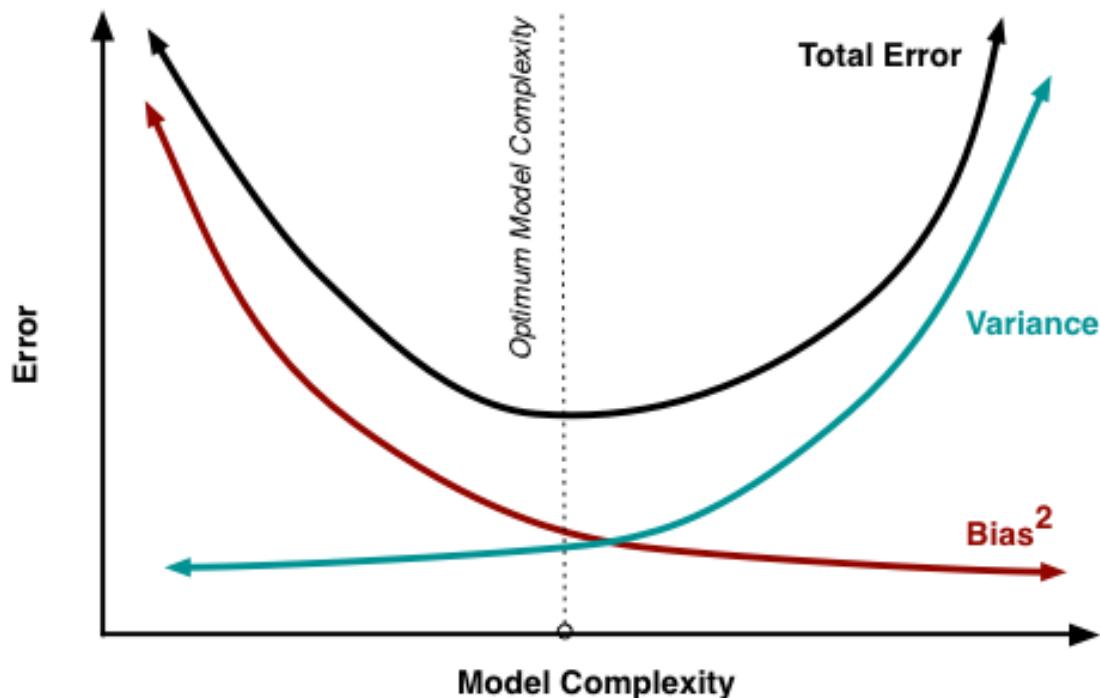


Figure 6.4: Bias-Variance trade off [46]

A very popular and effective method for tuning the parameters of the model is done by the use of the k-fold cross validation method, see Figure 6.5. This method randomly partitions the training set into k equal size subsets. A single subset is retained as validation data, used to evaluate the model's accuracy, while the other k-1 subsets are used to train the model. This process is repeated k times with each of the k subsamples used exactly once as the validation data. The goal of cross validation is thus to reduce the bias and variance error, before the model is evaluated on the test data.

However, an appropriate value of k needs to be chosen. A high value of k will elevate the variance, while a low value of k will result in a high bias. Most literature appoints a value of k=10, however since our data is quite limited and thus inherent to having a high chance of overfitting, we will use a value of k=5.

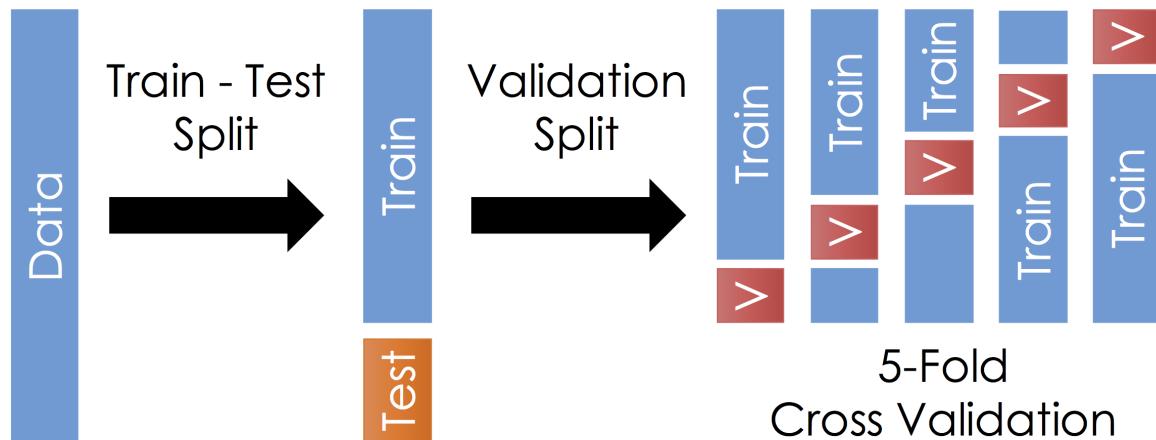


Figure 6.5: Data splitting

In Table 6.1, the amount of data samples used in the training and test set for each separate problem is shown. Since the measurement campaign is a very time consuming process, the machine learning methods can only be trained on a limited data set. This badly influences the full potential of some of the proposed methods.

In regular machine learning approaches it is suggested the training and test set should be respectively 70% and 30% of the total data set. Here, this ratio is around 60/40%. This is to make sure the variance on the performance statistics is low. A part of the LOS training data is used in the NLOS training set. This helps the machine learning to identify the important features that are only correlated to the extra $d_{bias,NLOS}$.

Table 6.1: Number of data samples used in each problem

	Training set	Test set
LOS Mitigation	640	380
NLOS Mitigation	628	312
Identification	943	692

Table 6.2: Confusion matrix

		Actual	
		LOS	NLOS
Predicted	LOS	True Positive (TP)	False Positive (FP)
	NLOS	False Negative (FN)	True Negative (TN)

6.2.3 Evaluation metrics

To evaluate the performance of each tuned model, some evaluations metrics need to be defined. The evaluation metrics used in classification differ completely from the ones used in regression problems, due to the different nature of the outcome of the predictor.

Classification

To explain the different classification metrics, first all the different possible outcomes are shown in Table 6.2. This is also called a confusion matrix.

The first classification metric, the accuracy, is the most intuitive. It is defined as the ratio of correct predicted observations on the total number of predicted observations. It is denoted as:

$$\text{Accuracy} = \frac{\text{Total number of correct predictions}}{\text{Total number of predictions}} = \frac{TP + TN}{TP + FP + FN + TN} \quad (6.5)$$

However, in our LOS/NLOS identification case, it is important to avoid False Positive predictions. This is when $h(\mathbf{x})$ predicts an LOS case while the actual measurement is taken in an NLOS environment. That means a high reliability is given to a measurement with a high error variance. These False Predictions need to avoided at all cost. False Positive predictions can be evaluated by the False Positive Rate (FPR), also called the Fall-out metric. This is defined as:

$$FPR = \frac{FP}{FP + TN} \quad (6.6)$$

Regression

The accuracy of the different regression models will be evaluated by using two statistic error definitions. The mean absolute error (MAE) is a measure of the average absolute error between predicted values and observed values, while the mean squared error (MSE) represents the variance of the differences between predicted values and observed values. Mathematically they are formulated as:

$$MAE = \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{n} = \sum_{i=1}^n \frac{|y_i - h(x_i)|}{n} \quad (6.7)$$

$$MSE = \sum_{i=1}^n \frac{(y_i - \hat{y}_i)^2}{n} \quad (6.8)$$

With n the size of the samples, y_i the output of the i -th sample and $\hat{y} = h(x_i)$ the prediction of the i -th sample with x_i the input.

The main difference between these two definitions is the weight that is given to large errors. Due to the squaring of the errors in the MSE definition, large errors are given a greater weight. This means the MSE should be more useful when large errors are particularly undesirable.

To denote the improvement of the error on a certain subset we will denote:

$$MAE \text{ Improvement}(\%) = \frac{MAE_{uncorrected} - MAE_{model}}{MAE_{uncorrected}} \quad (6.9)$$

$$MSE(\%) \text{ Improvement} = \frac{MSE_{uncorrected} - MSE_{model}}{MSE_{uncorrected}} \quad (6.10)$$

6.3 Machine learning methods

In this section, the various machine learning methods will be discussed. As previously mentioned a distinction has been made between classification and regression problems. Some of the presented methods have the ability to be used in both cases. However, another distinction can be made based on the form of the predictor $h(x)$. Parametric methods have a predetermined $h(x)$ form while the predictor of non-parametric methods is purely constructed according to information derived from the data.

Both have particular advantages and disadvantages. Non-parametric methods are only efficient when enough data is presented. Even though they are inherently more likely to overfit (high variance) and computations take longer than parametric methods, the performance of non-parametric methods can be much higher for very complex problems. Typical to non-parametric machine learning models is the need for tuning hyperparameters. These differ

Table 6.3: Summary machine learning methods

	Classification	Regression
Non-Parametric	Random forest Support vector machine	Random forest Support vector machine
Parametric	Logistic	Univariate Multiple

from the regular to be trained parameters because they need to be set before the learning process begins. Since tuning can be very time consuming, only the most important hyperparameters will be investigated upon.

The use of parametric methods is generally more easy to understand and interpret than non-parametric approaches. It requires less data to have a good fit and computation times are significantly lower. However parametric methods are often avoided in too complex problems because of the specified predetermined form $h(x)$. A short summary of the methods can be found in Table 6.3.

6.3.1 Random forest

The first method that will be discussed is random forest. Random forest is a type of machine learning ensemble method, which makes predictions by averaging over the predictions of several independent decision trees. Since its introduction it has been extremely successful as a general purpose method [47]. An advantage of the random forest method is that it can be used for both classification and regression.

Principle

To give a clear understanding of the principle of random forest, an explanation of decision trees needs to be given first. A decision tree is a tree $h(X)$ where each node represents a conditional feature statement, each link represents a decision and each leaf represents an outcome. Shortly a decision tree predicts a certain outcome, discrete or continuous, based on conditional control statements in the nodes. An example of a decision tree with depth length 3 applied on our LOS data can be seen in Figure 6.6.

The main advantage of the use of decision trees for machine learning is the invariance under scaling and various other transformations of feature values. Normalization rescaling in the data preprocessing part is thus not necessary for this method. Another advantage is the robustness to inclusion of irrelevant features. However, they are seldom accurate due to the fact that deeply grown trees tend to easily overfit the training set (high variance) [48].

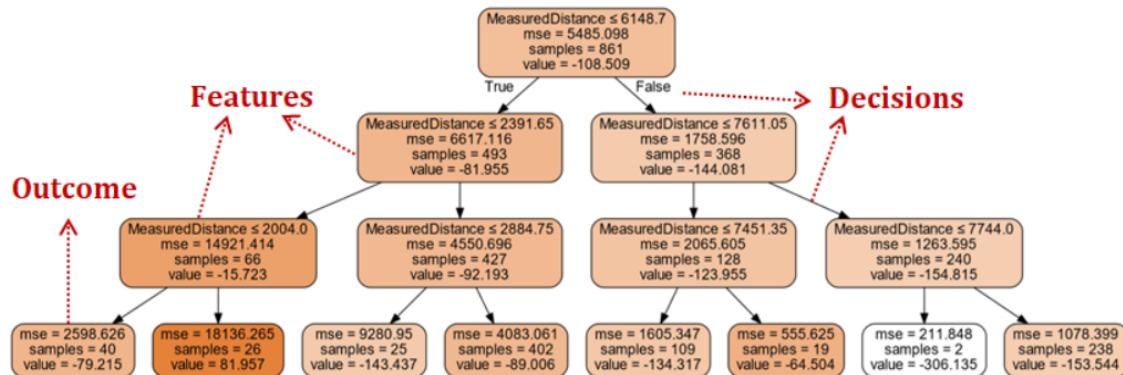


Figure 6.6: An example of a decision tree with depth length 3 applied on the LOS data

This overfitting can clearly be seen when applying decision trees with different depth lengths on our LOS training data, see Figure 6.7. The deeper the tree, the more different outcome levels.

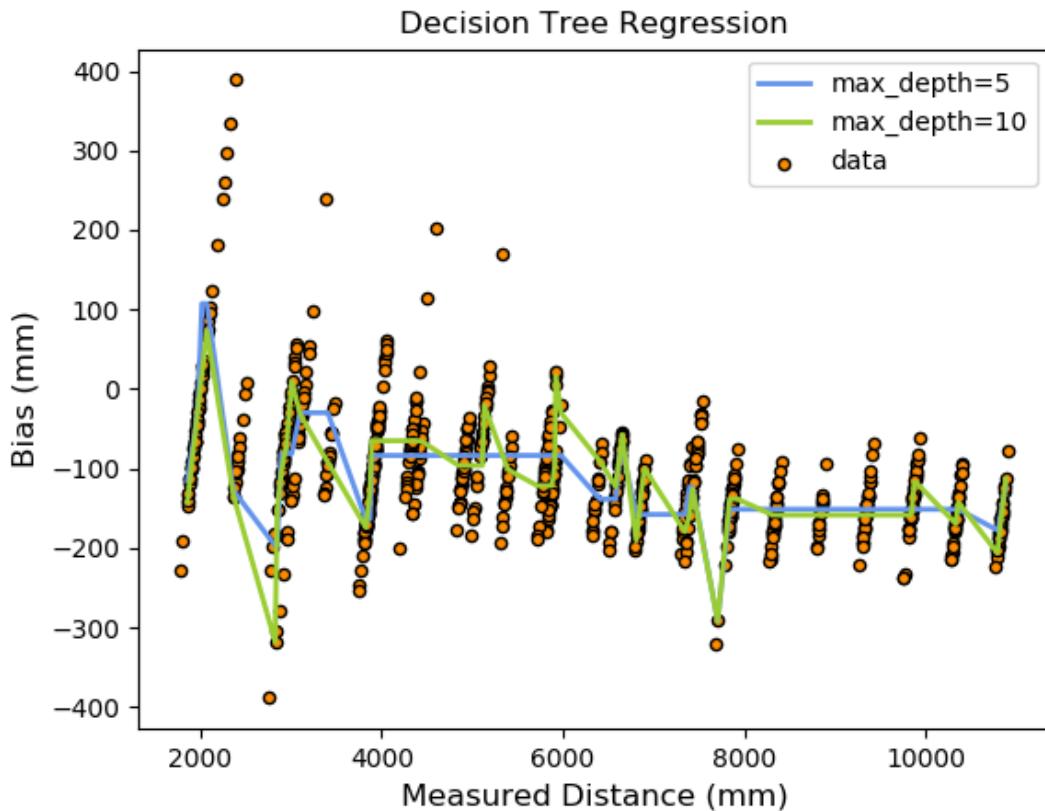


Figure 6.7: Influence of different depth lengths of decision trees applied on the LOS data

The solution to this overfitting is the ensemble of n multiple deep uncorrelated decision trees ($h_1(X), h_2(X), h_3(X), \dots, h_n(X)$), denoted as random forest. The regression prediction is then formed by taking the average of the individual predictions of the n trees. The decrease in variance is only guaranteed if the trees are uncorrelated. This is achieved by the generation of random vectors that govern the growth of each tree in the ensemble [49].

Various methods for generating these vectors exist, of which bootstrap aggregating is the most popular. In this method, also known as bagging, a random selection (without replacement) is made from the samples in the training set [49].

Hyperparameter Tuning

The hyperparameters will be discussed in order of their importance. To find the best values of these hyperparameters, a grid search over them is commonly done. This grid search is an exhaustive searching through a manually specified subset of the hyperparameter space of a learning algorithm. In order to find appropriate subsets and to minimize computing time, one can first visually inspect them. This gives more insight and an educated choice for the range of the hyperparameters.

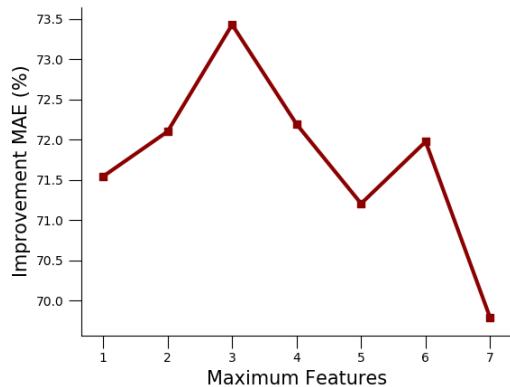
Max number of features The hyperparameter K sets the maximum number of features that are allowed in the individual decision tree. Specifically, it is the number of features that are each time considered when looking for the best split. A high K generally improves the model's performance but leads to a slower speed of the model. A small value of K gives rise to a strong randomization and a fast computation time [50]. A good balance needs to be found.

Number of Trees This is the number of trees you want to build before taking the average of all the individual predictions. Intuitively a higher number of trees increases the model's performance however it also increases computation time. As the number of trees grows, it does not always mean the performance of the forest is significantly better than previous forests (fewer trees). It is possible to state there is a threshold beyond which there is no significant gain, unless a huge computational environment is available [49], [51].

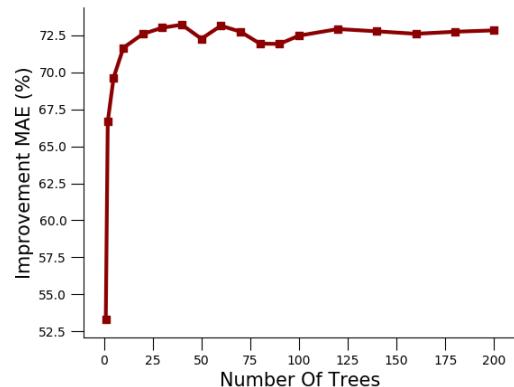
Max. Depth Length This hyperparameter sets the maximum allowed number of levels in each decision tree. The deeper the tree, the more splits it has and the more information it captures about the data. However a too high depth length can lead to unnecessary long computation times. The chance of overfitting is limited due to the ensemble method.

Min. Leaf Size Minimum leaf size specifies the smallest number of samples a node is allowed to have. If the splitting rule results in a child node with fewer samples than this number, the node is not split. A smaller leaf size makes the model more prone to capturing noise, since the tree can generally grow deeper. Increasing the minimum samples leaf limit thus generally decreases variance and increases bias.

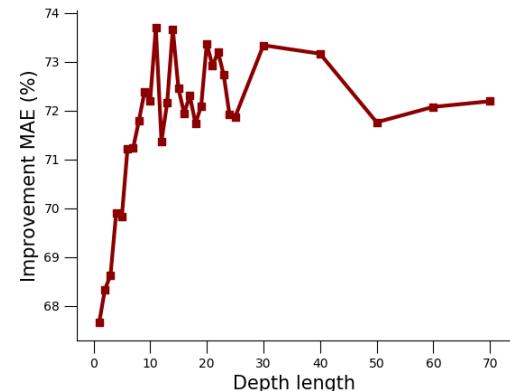
The influences of the hyperparameters on the improvement of the MAE for the LOS mitigation method can be seen in Figure 6.8.



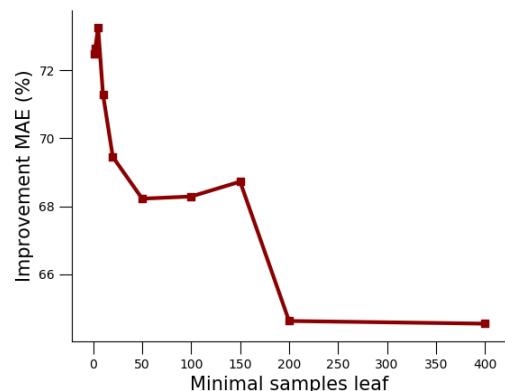
(a) Maximum features to be considered for splitting



(b) Number of trees in the random forest



(c) Maximum depth length for each decision tree



(d) Minimal leaf size for each decision tree

Figure 6.8: Influences of the hyperparameters on the improvement of the MAE for the LOS mitigation method.

6.3.2 Support vector machine

Support vector machine (SVM) algorithms can be used for classification and regression analysis, as well for the detection of outliers. The main reason SVM methods are chosen is due to their effectiveness in high dimensional spaces. Since our data has a relatively small amount of data samples with a high number of features, SVM is thus a good consideration. The

main drawback of SVM methods is that they are distance based methods. This has the consequence that features with a broader range of values will be more dominant than smaller range features. Therefore data rescaling is needed. The method of standardization is used, see 6.2.1. This is the most widely used rescaling method for SVM and has been proven to perform well in a high variety of problems.

Principle

In short, the goal of support vector machine algorithms is to find the optimal separating hyperplane, which maximizes the margin of the training data. Firstly, a hyperplane is defined as a subspace whose dimension is one less than that of the ambient space. Thus specifically in SVM, a hyperplane has a n-1 dimension when n features are taken into account. Mathematically this hyperplane can be described with a linear equation of the following form:

$$\vec{w}^T \vec{x} = b \quad (6.11)$$

With \vec{w}^T the weight vector, \vec{x} the input vector and b the bias.

An optimized hyperplane is a hyperplane for which the margin is maximized. The margin is defined as the separation distance between the hyperplane and the closest data samples. The higher the margin the less the method is prone to errors. Thus logically the maximal margin will be searched. These closest data samples are called support vectors, they are the points that are most difficult to classify or predict regressively. They have direct bearing on the optimum location of the decision surface.

Classification example To make this all more clear, an optimized hyperplane is shown for a simple classification example, see Figure 6.9. In this two dimensional feature space, the hyperplane will be a line. To calculate this optimized hyperplane, firstly the boundary planes B+ and B- will be defined such that for each i-th sample (x_i, y_i) :

$$\begin{cases} \vec{w}^T \cdot \vec{x}_i + b \geq 1 & \text{if } y_i = 1 \\ \vec{w}^T \cdot \vec{x}_i + b \leq -1 & \text{if } y_i = -1 \end{cases} \quad (6.12)$$

In short each i-th sample thus needs to satisfy:

$$y_i(\vec{w}^T \cdot \vec{x}_i + b) \geq 1 \quad (6.13)$$

The planes B+ and B- are the boundaries and will thus be defined as:

$$\begin{cases} \vec{w}^T \cdot \vec{x} + b = 1 & \text{for } B+ \\ \vec{w}^T \cdot \vec{x} + b = -1 & \text{for } B- \end{cases} \quad (6.14)$$

By linear algebra, the margin, the distance between these hyperplanes, is calculated by:

$$M = \frac{2}{\|\vec{w}\|} \quad (6.15)$$

The maximum margin can thus be found by minimizing $\|\vec{w}\|$. The parameters \vec{w} and b of the optimized hyperplane will be found by solving following objective function using quadratic programming.

$$\begin{cases} \min \frac{1}{2} \|\vec{w}\|^2 \\ y_i(\vec{w}^T \cdot \vec{x}_i + b) \geq 1 \quad \forall x_i \end{cases} \quad (6.16)$$

However, in most cases a perfect separation is not feasible. In these cases the SVM finds the hyperplane that maximizes the margin and minimizes the misclassifications. The objective function then becomes:

$$\min \frac{1}{2} \|\vec{w}\|^2 + C \sum_{i=1} \xi_i \quad (6.17)$$

With C the trade-off parameter and ξ_i the error misclassification. This trade-off parameter C is a hyperparameter that needs to be tuned.

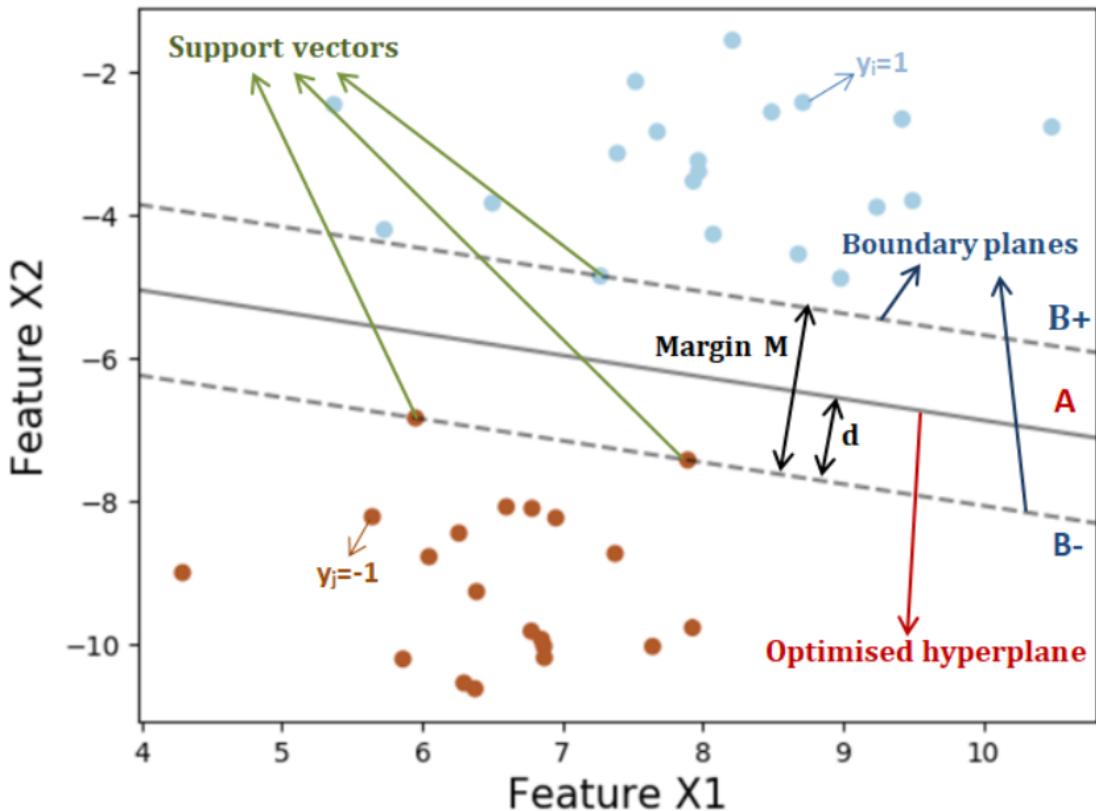


Figure 6.9: SVM classification example

So far we have seen that the groups can be linearly separated, however sometimes nonlinear regions separate the different classes more accurately. This nonlinear separation is done by

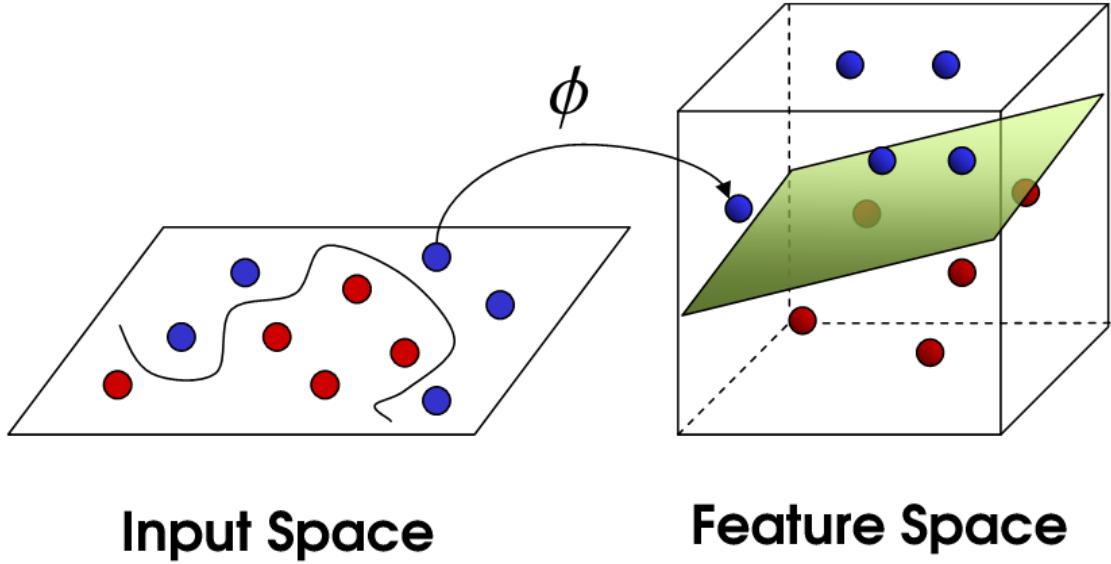


Figure 6.10: Kernel function [52]

the use of a kernel function. A kernel function transforms the data points in a much higher dimensional feature space. The normal linear separation procedure can then be performed like discussed previously. After the linear solution is obtained, it is transformed back to the original feature space. This will result in a nonlinear separation. In machine learning this is also called the ‘Kernel trick’, see Figure 6.10.

Regression example So far we have discussed a classification example. The regression SVM algorithm is similar to the classification in the sense that it has the same properties of the margin maximization and kernel trick for nonlinear mapping [53]. The optimized hyperplane however is now found by minimizing $\frac{1}{2}\|\vec{w}\|^2$ subject to, see Figure 6.11:

$$\begin{cases} y_i - \vec{w}^T \cdot \vec{x} - b \leq \varepsilon \\ \vec{w}^T \cdot \vec{x} + b - y_i \leq \varepsilon \end{cases} \quad (6.18)$$

Hyperparameter Tuning

The hyperparameters will be discussed in order of their importance. To find the best values of these hyperparameters, a grid search over them is commonly done.

Kernel The most important hyperparameter to choose is the type of kernel function. The most widely used are linear, polynomial, sigmoid and radial based function (RBF) kernel functions.

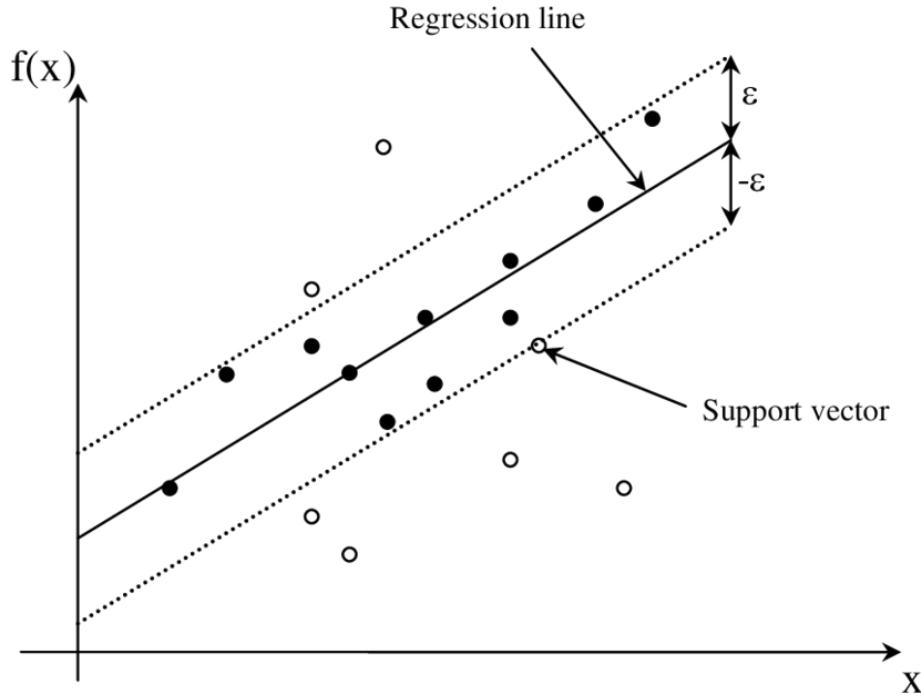


Figure 6.11: SVM Regression principle [54]

Regularization The regularization parameter, mostly termed as C , affects the trade-off between complexity and proportion of non-separable samples and must be selected by the user [55]. A small C allows more errors thus produces a larger margin (high bias). A small margin, also called a hard-margin, is obtained when C goes to infinity (high variance).

Gamma For the polynomial, sigmoid and radial based kernels, another hyperparameter needs to be tuned. The kernel parameter Γ is used to control the locality of the kernel function. A grid search for both the Gamma and Regularization for an RBF kernel function is shown in Figure 6.12.

6.3.3 Univariate regression

Hypothesis

The first parametric method that will be discussed is the univariate regression. Univariate means that only one feature $x = x_i$ is taken into account in the hypothesis $h(x) = h(x_i)$. The hypothesis takes following general polynomial form:

$$h_{\Theta}(x_i) = \Theta_0 + \Theta_1 x_i + \Theta_2 x_i^2 + \Theta_3 x_i^3 + \dots + \Theta_n x_i^n \quad (6.19)$$

$$= \Theta * \phi(x_i) \quad (6.20)$$

With $\Phi(x_i) = [1, x_i, x_i^2, x_i^3, \dots, x_i^n]$ and $\Theta = [\Theta_0, \Theta_1, \Theta_2, \dots, \Theta_n]$. The parameter vector Θ is tuned with the help of the k cross validation of the training set explained in subsection 6.2.

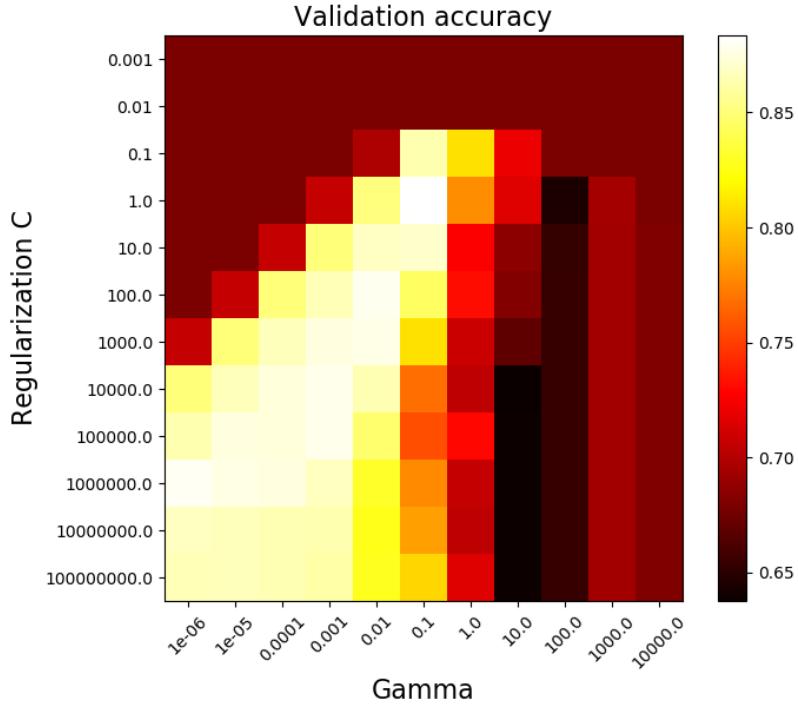


Figure 6.12: Grid search over the regularization and the gamma parameters. The brighter the higher the validation accuracy.

Visual inspection

When visually inspecting the data, we see that for the LOS training data, there is a possibility of a good linear or a low order polynomial relation between the distance bias and the measured distance. This relation is also appointed in DecaWave’s user manual [28]. For insight, some polynomials are calculated with different degrees, see Figure 6.13.

Finally, to see if other features have reasonable polynomial forms, the improvement on the MAE is calculated in function of the degree of polynomial. A high degree of polynomial inherently leads to a high overfitting on the training samples, thus when performing k-cross validation, the accuracy will naturally decrease for high degrees, since the error increases for the validation samples. This effect of the maximum polynomial degree on the improvement of the MAE for each separate feature for both the LOS and the NLOS mitigation method can be seen in respectively Figure 6.14 and 6.15.

6.3.4 Multivariate regression

In multivariate regression, we take into account multiple features in the regression problem. An example of a hypothesis for two features x_1, x_2 with interaction with maximum polynomial degree 2 has the form:

$$h(x) = \Theta_0 + \Theta_1 * x_1 + \Theta_2 * x_2 + \Theta_3 * x_1 * x_2 + \Theta_4 * x_1^2 + \Theta_5 * x_2^2$$

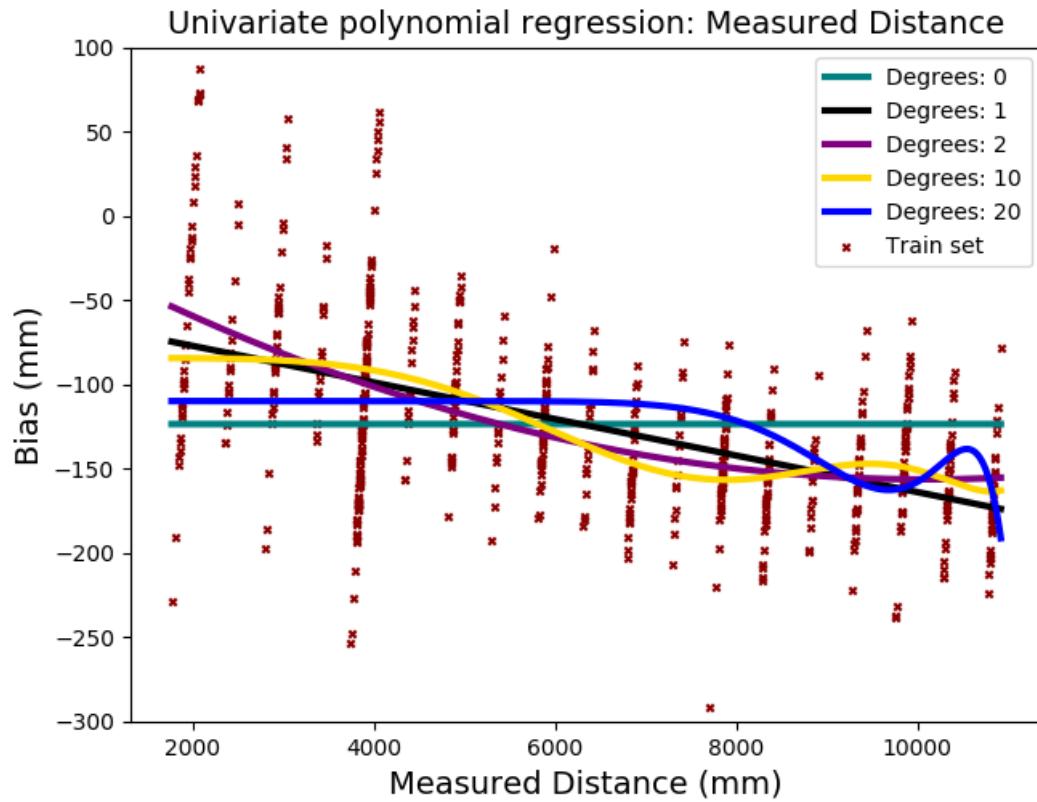


Figure 6.13: Univariate regression on the 'Measured Distance' feature

With a high number of features, a high number of possible regression forms is possible. In order to find the optimal regression form, a grid search is performed. Here, the amount and the selection of features, the interaction term and the maximum polynomial degree are varied.

6.3.5 Logistic regression

Unlike the name of the method, logistic regression methods are used for classification problems. It is called a regression method because it basically outputs the probabilities of the different discrete classes labels. It is a special case of the generalized linear model and thus analogous to linear regression.

Consider a general biclassification problem with $y \in \{0, 1\}$. In general, a predictor $h(x)$ can be a simple construction of a multivariate polynomial regression. However these will generally perform poor classifications and $h(x)$ will predict values higher than or lower than the discrete class labels. To solve this we change $h(x)$ to only take values between the two classes $0 \leq h(x) \leq 1$. This is done by letting the hypothesis $z = h(x)$ be the independent

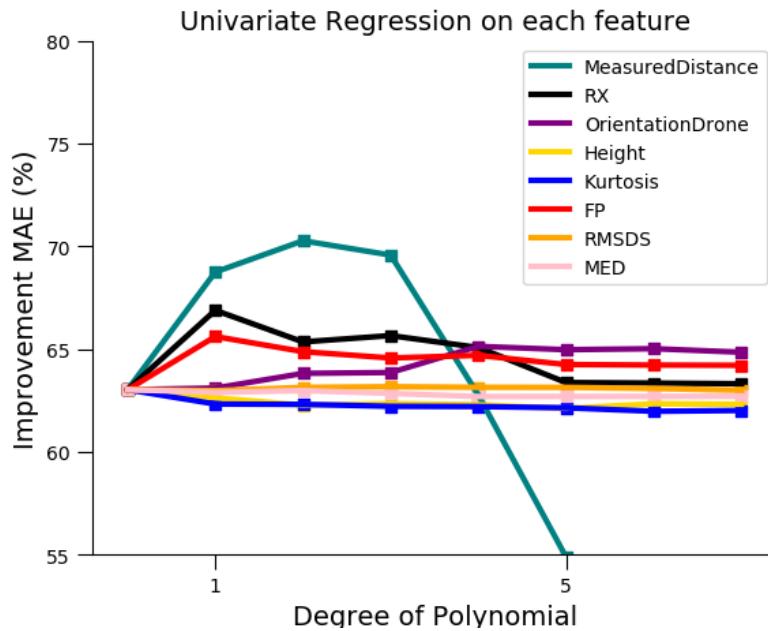


Figure 6.14: The improvement on MAE for each feature dependent on the maximum polynomial degree for the LOS mitigation method

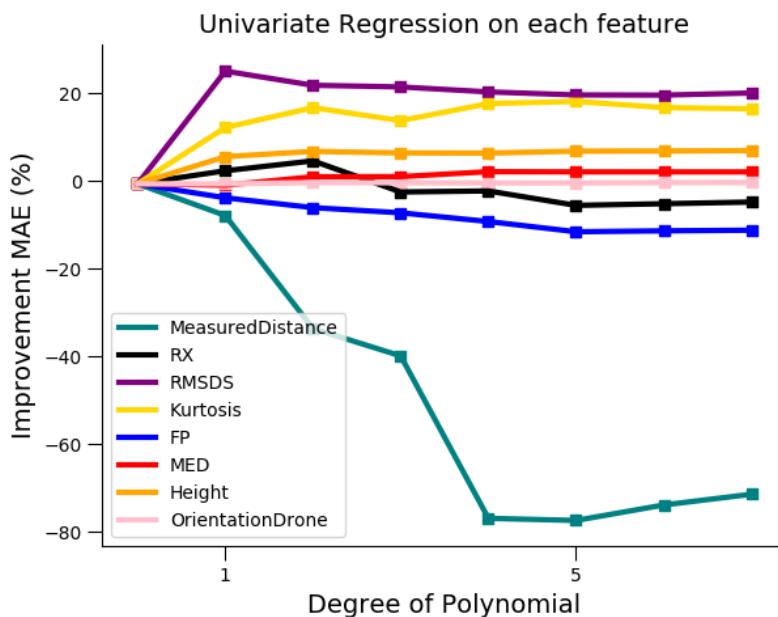


Figure 6.15: The improvement on MAE for each feature dependent on the maximum polynomial degree for the NLOS error mitigation method

variable of the Sigmoid function $g(z)$. This Sigmoid function is also known as the logistic function, see Figure 6.16. As can be seen any real number is mapped to the $(0, 1)$ interval. It is formulated as:

$$g(z) = \frac{1}{1 + e^{-z}} \quad (6.21)$$

The function will thus be the new updated hypothesis $g(h(x))$:

$$g(h(x)) = \frac{1}{1 + e^{-h(x)}} \quad (6.22)$$

In order to get discrete class labels $y \in \{0, 1\}$, the output of the hypothesis function is translated as:

$$\begin{cases} g(h(x)) \geq 0.5 \rightarrow y = 1 \\ g(h(x)) < 0.5 \rightarrow y = 0 \end{cases} \quad (6.23)$$

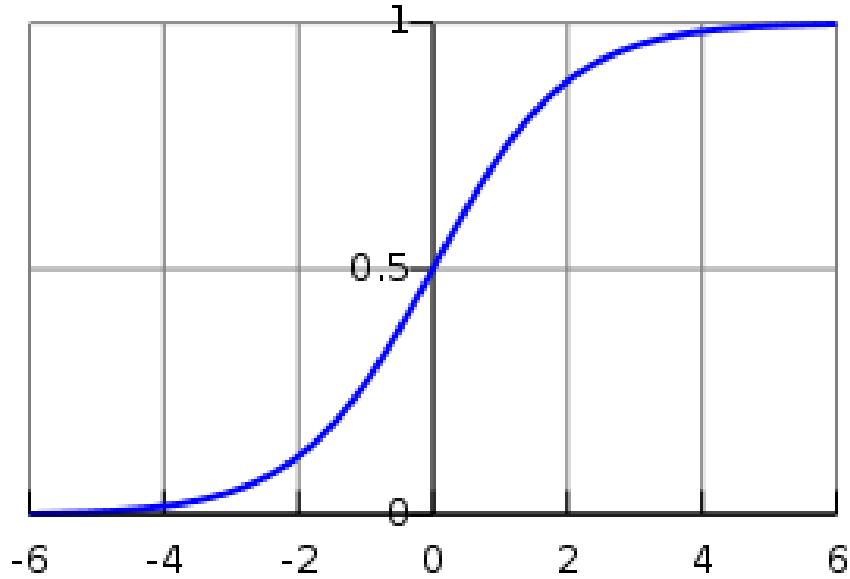


Figure 6.16: The logistic function

6.4 Feature Selection

Feature selection is defined as the process of selecting a subset of relevant features for use in model construction. It is used to eliminate unneeded, irrelevant and redundant features from data that do not contribute to the accuracy of a predictive model or may in fact decrease the accuracy of the model [56]. Next to this, removing features results in a faster and more cost effective predictor. At last it provides a better understanding of the underlying process [57].

Variable Importance A particular advantage of the random forest machine learning method is the possibility of ranking the importance of the different features in a natural way. This is done with the help of bootstrap sampling, explained in Subsection 6.3.1. The evaluation of the algorithm beneath is done with the help of the out-of-the-bag error (OOB). This is the mean prediction error on each training sample X , using only the trees that did not have X in their bootstrap sample.

A short explanation of the algorithm:

- Each decision tree is assigned sample data according to the bootstrap method.
- Around 2/3 of the sample data per tree is used to fit the tree to its greatest depth.
- For each tree, the out-of-the-bag error (OOB) is calculated using the rest of the sample data. Next, for each feature in the tree, the feature variables are permuted and the OOB error is again computed. An increase of the OOB is an indication of the variable's importance.
- Finally the feature importance is measured by averaging the increase in OOB error over all trees.

For both the mitigation methods and the NLOS/LOS identification method, the importance of the features is calculated. See respectively Figures 6.17, 6.18 and 6.19.

It can be seen that for the LOS mitigation method the distance is the most important feature. This relation is also appointed in DecaWave's user manual [28]. However, other features have some minor importance too. The first path power, the orientation of the drone and the standard deviation of the noise are also correlated to the bias occurring in Line-of-Sight situations. For the NLOS mitigation method, two features are considered to be very important. These are the measured distance, which is also an important factor in LOS situations, as well as the received power. The extra occurring NLOS bias is thus most likely to be correlated to the received power.

The most interesting feature ranking is for the NLOS/LOS classification problem. As discussed in Chapter 5, the received power, the distance, RMSDS and the standard deviation of the measured distance could be good indicators. These indicators are also proven to be good indicators by the feature ranking, except for the standard deviation that is given a very small importance.

Feature Filtering

Based on the feature importances, the features are ranked. The accuracy of adding an extra feature, starting from highest variable importance score to lowest, to the set of features is

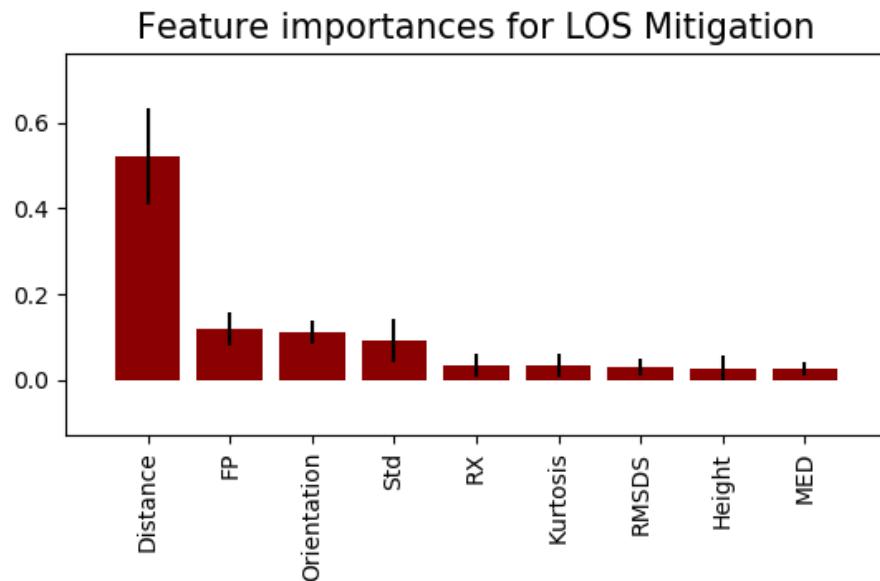


Figure 6.17: Ranking of feature importances for LOS mitigation method

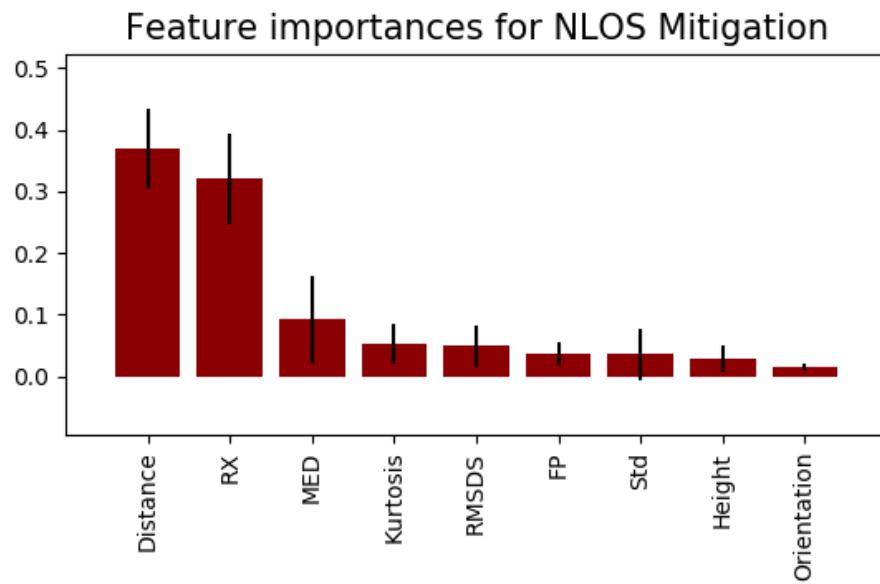


Figure 6.18: Ranking of feature importances for NLOS mitigation method

calculated. As long as the increase in accuracy is satisfied, the feature is added to the feature set. The final set will be then used to further tune the hyperparameters.

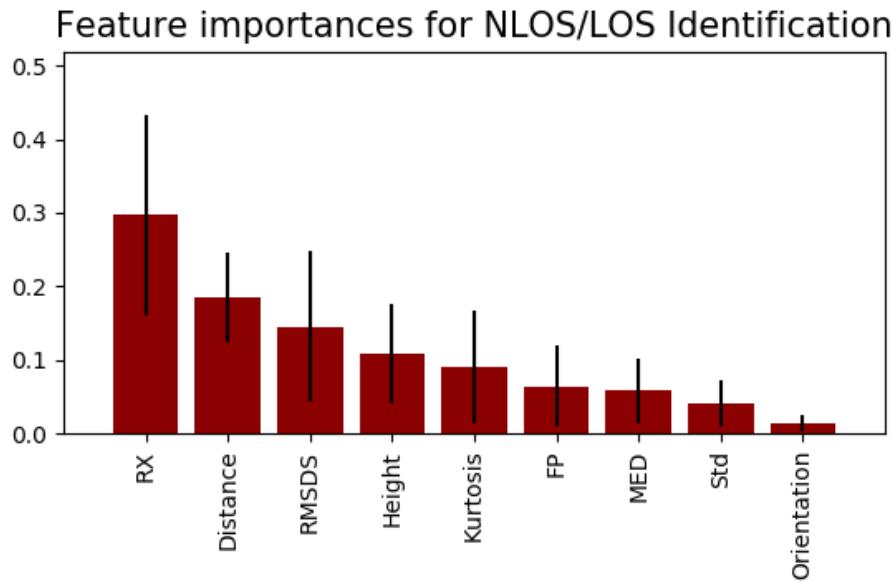


Figure 6.19: Ranking of feature importances for LOS/NLOS identification

6.5 Performance Comparison

6.5.1 Classification

In Table 6.4, the results of the different classification methods applied on an unseen test set can be seen. All the methods have a reasonable amount of accuracy and a small False Positive Rate. It can be noticed that when a method has a good accuracy, the FPR rate is higher. Since Logistic has a low FPR and requires a minimum amount of computation time, the logistic regression method is preferred.

Table 6.4: Evaluation of the different classification methods on the unseen test set

	Logistic	Random forest	SVM
Accuracy (%)	89.6	91.6	94.2
FPR (%)	1.3	4.4	9.5

6.5.2 Regression

LOS mitigation method

In Table 6.6, the results are shown of the different regression methods applied on an unseen NLOS test set. Next to this, for visual insight, the cumulative distribution functions of the bias of the corrected distances are plotted in Figure 6.21.

As can be seen, the random forest has the best improvement on both MAE and MSE. The SVM and the univariate regression methods also show reasonable results. With the best method, the random forest, a mean accuracy of 3.7cm can be acquired.

Table 6.5: Evaluation of the different regression methods on the unseen test set

	Univariate	Multiple	Random forest	SVM
MAE (mm)	49.4	62.8	37.2	47.5
MSE (mm^2)	3784.6	6424.9	2400.6	3262.0
Improvement MAE	52.3%	39.3%	64.0%	54.0%
Improvement MSE	73.3%	54.7%	83.1%	77.0%

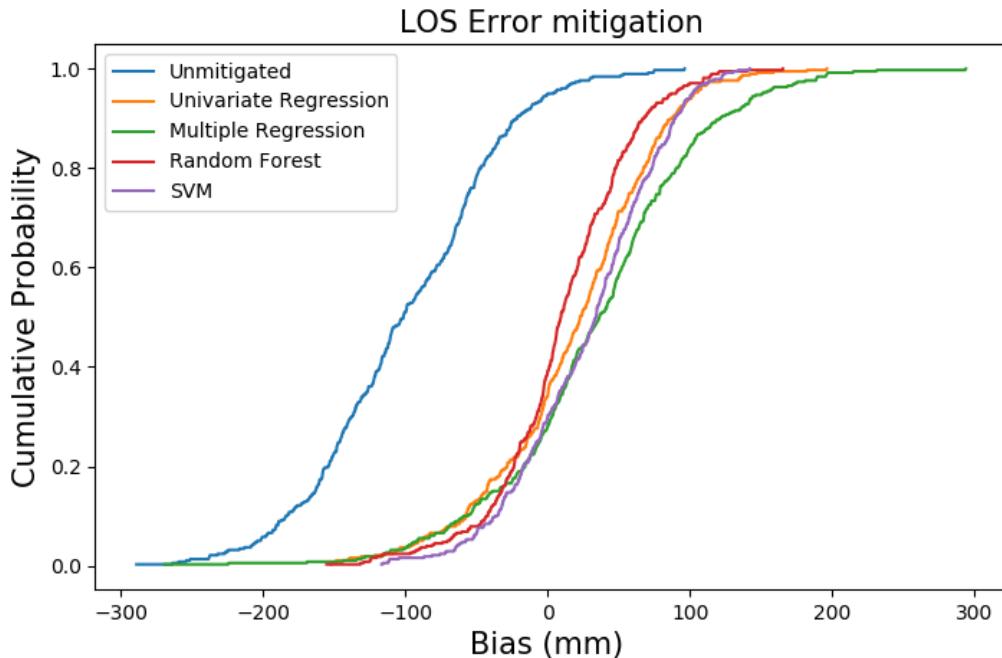


Figure 6.20: The cumulative distribution function of the bias of the corrected distance from an unseen LOS test set.

NLOS mitigation method

In Table 6.6, the results are shown of the different regression methods applied on an unseen NLOS test set. Next to this, for visual insight, the cumulative distribution functions of the bias of the corrected distances are shown in Figure 6.21.

As can be noticed, the parametric regression methods are not able to improve the unmit-

igated bias, instead they even deteriorate. The random forest method and SVM regression method give quite similar improvements. Low mean improvements of 42.3% can be achieved. Based on these two methods, a mean accuracy of 17cm can be achieved. This is significantly lower than the LOS mitigation test. This can be explained by the immense amount of unforeseeable factors that induce the extra NLOS bias. It is thus important in the drone procedure to always give preference to LOS corrected measurements and to give a higher uncertainty factor to NLOS corrected measurements in the weighted nonlinear least squares algorithm.

Table 6.6: Evaluation of the different regression methods on the unseen test set

	Univariate	Multiple	Random Forest	SVM
MAE (mm)	-	-	173.5	188.6
MSE (mm²)	-	-	54549.9	59695
Improvement MAE	< 0%	< 0%	42.3%	37.3%
Improvement MSE	< 0%	< 0%	57.9%	52.9%

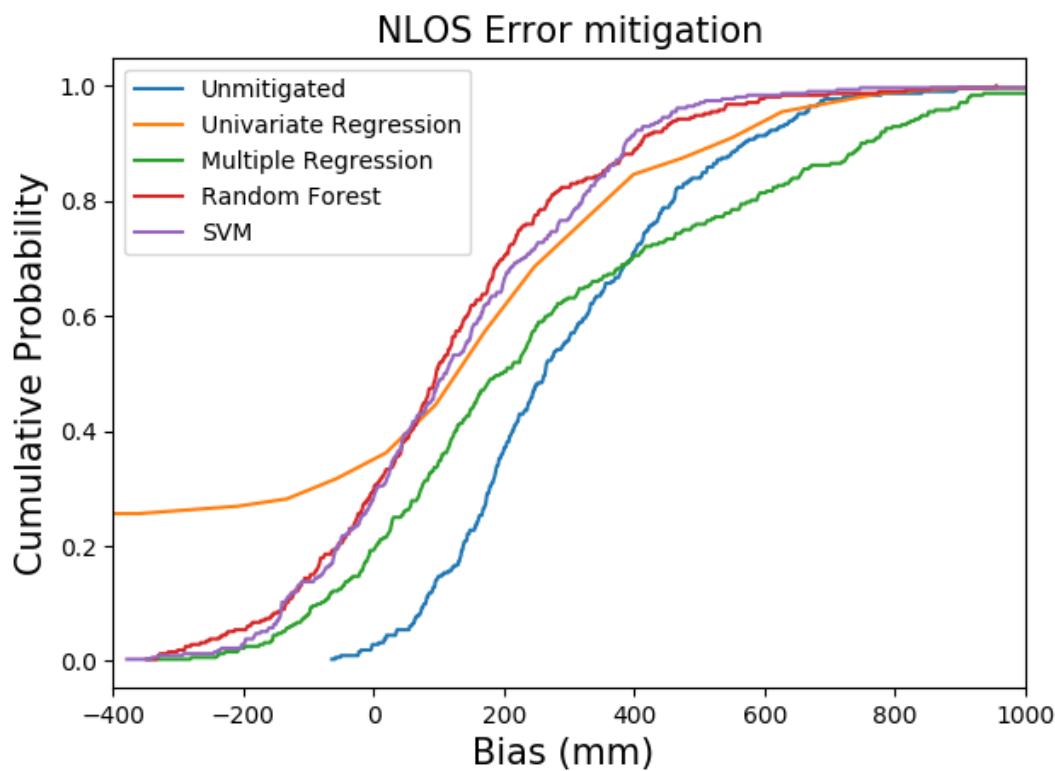


Figure 6.21: The cumulative distribution function for each machine learning method applied on unseen NLOS test set.

Chapter 7

Localization Algorithm

In this chapter a mathematical framework of the localization algorithm is presented based on the principle of multilateration, discussed in Chapter 2. First, a simple framework is given for the ideal case where no measurement errors occur. After this, errors are taken into account and the localization framework becomes more complex to solve. This complex localization is a nonlinear weighted least squares algorithm that will be further used to estimate the global coordinates of the fixed anchor. A weighted algorithm solves the problem of the different error variance for NLOS/LOS situations after applying mitigation methods. For simplicity, the algorithm will be explained for 2D localization problems. The extension to 3D is trivial.

7.1 Ideal Linear Least Squares

Suppose at n different points, distance measurements to the anchor were taken. When no measurement errors are assumed, Pythagoras' theorem is valid:

$$\begin{bmatrix} d_1^2 \\ d_2^2 \\ \dots \\ d_n^2 \end{bmatrix} = \begin{bmatrix} (x_1 - x_a)^2 + (y_1 - y_a)^2 \\ (x_2 - x_a)^2 + (y_2 - y_a)^2 \\ \dots \\ (x_n - x_a)^2 + (y_n - y_a)^2 \end{bmatrix} \quad (7.1)$$

With (x_i, y_i) the coordinates of point i , (x_a, y_a) the coordinates of the to be localized anchor and d_i the distance between point i and the anchor. This can be transformed into a linear form by subtracting the last equation from the other equations and reordering:

$$A = 2 \begin{bmatrix} (x_1 - x_n) & (y_1 - y_n) \\ (x_2 - x_n) & (y_2 - y_n) \\ \dots & \dots \\ (x_{n-1} - x_n) & (y_{n-1} - y_n) \end{bmatrix} \quad (7.2)$$

$$b = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_n^2 - d_1^2 \\ x_2^2 - x_n^2 + y_2^2 - y_n^2 + d_n^2 - d_2^2 \\ \dots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_n^2 - d_{n-1}^2 \end{bmatrix} \quad (7.3)$$

$$x = \begin{bmatrix} x_a \\ y_a \end{bmatrix} \quad (7.4)$$

When $n > 2$ this over determined system can be solved using an ordinary least squares estimation with solution:

$$x = (A^T A)^{-1} A^T B \quad (7.5)$$

An example can be seen in Figure 7.1.

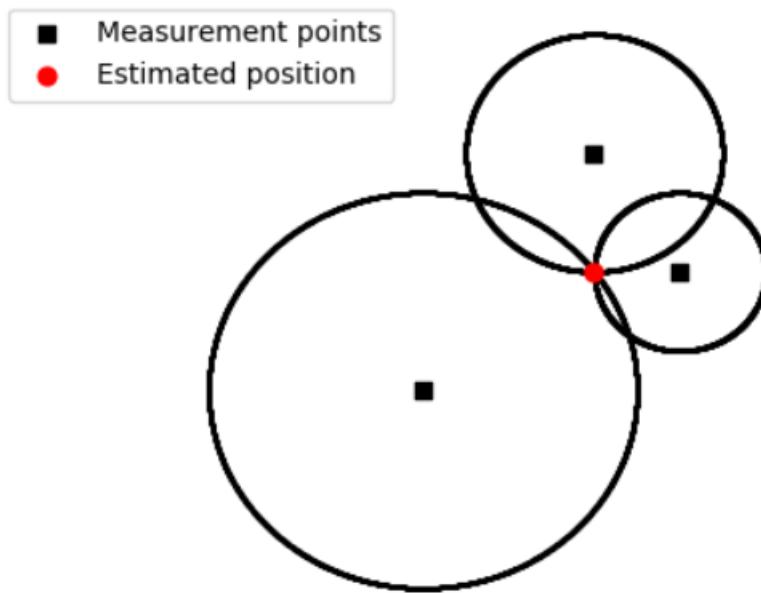


Figure 7.1: Ideal linear least squares.

7.2 Nonlinear Weighted Least Squares

In reality however, there are inaccuracies on the distance measurement and on the position measurements.

7.2.1 Errors

Distance error

For each separate measurement, the general form holds:

$$d_{i,measured} = d_{i,exact} + d_{i,bias} + n_{thermal} \quad (7.6)$$

With $d_{i,exact}$ the exact distance from point i to the anchor, $d_{i,measured}$ the measured distance, $d_{i,bias}$ the occurring distance bias and n the noise Gaussian distributed with mean 0, see Chapter 4. When averaging out the distance measurements made at a certain position, the noise is eliminated. By performing error mitigation a prediction $d_{i,bias,estimated}$ is calculated in order to cancel out $d_{i,bias}$. However small errors e_d will still exist.

$$d_{i,estim} = d_{i,exact} + e_d = d_{i,exact} + (d_{i,bias} - d_{i,bias,estimated}) \quad (7.7)$$

Position error

The coordinates of the measurement points are measured by the RTK module, however these measurements have inaccuracies leading to an error e_{pos} .

Thus taken these errors into account:

$$\begin{bmatrix} d_1 \\ d_2 \\ \dots \\ d_n \end{bmatrix} = \begin{bmatrix} \sqrt{(x_1 - x_a)^2 + (y_1 - y_a)^2} \\ \sqrt{(x_2 - x_a)^2 + (y_2 - y_a)^2} \\ \dots \\ \sqrt{(x_n - x_a)^2 + (y_n - y_a)^2} \end{bmatrix} + \begin{bmatrix} e_{pos,1} + e_{d,1} \\ e_{pos,2} + e_{d,2} \\ \dots \\ e_{pos,n} + e_{d,n} \end{bmatrix} \quad (7.8)$$

7.2.2 Nonlinear weighted squares problem statement

Due to this introduction of unknown errors, the problem has shifted from finding an exact solution to finding the most plausible solution, which has the smallest error. This solution can be stated in the form of the minimization of the sum of the squares of the errors:

$$(x_a, y_a)^* = \operatorname{argmin}_S S(x_a, y_a) = \operatorname{argmin} \sum_{i=1}^n e_i^2 = \operatorname{argmin} \sum_{i=1}^n (d_i - \sqrt{(x_i - x_a)^2 + (y_i - y_a)^2})^2 \quad (7.9)$$

Here, S is the sum of the squares of the errors and n the amount of different distance measurements. This form is formally known as the nonlinear least squares problem statement. Various methods have been developed to solve this iteratively, which will be discussed later. However this general form assumes that all the errors have the same variance. As was shown in previous Chapter 6 , the Non-Line-of-Sight error mitigation has a larger correction uncertainty and thus should logically be appointed a higher error variance. An updated version, holding into account these different error variances between measurements, is a weighted nonlinear least squares approach:

$$(x_a, y_a)^* = \operatorname{argmin} \sum_{i=1}^n W_i e_i^2 = \operatorname{argmin} \sum_{i=1}^n W_i (d_i - \sqrt{(x_i - x_a)^2 + (y_i - y_a)^2})^2 \quad (7.10)$$

Here, W_i stands for a weighted factor depending on the error variance of the distance and position measurement at point i.

7.2.3 Weight selection

The accuracy of the nonlinear weighted method strongly depends on the careful choice of the weighting factor per measurement point. The position error variance, caused by the inaccuracy of the RTK module, denoted as σ_{pos}^2 is assumed constant over all measurement points. The distance error variance however strongly depends on LOS/NLOS conditions. Since the LOS correction accuracy is much higher than the NLOS correction, LOS has a lower distance error variance σ_d^2 . The weighting factor is calculated as [58] :

$$W_i = \frac{1}{\sqrt{\sigma_{pos}^2 + \sigma_{d,i}^2}} \quad (7.11)$$

Using the results of the machine learning error mitigation methods, we can assume that the mean variance of the error for LOS is around $2401mm^2$, while the variance of the error for NLOS is $54550mm^2$. The variance of the error of the RTK module can be estimated knowing that the RTK promises an accuracy around $20mm$, thus the variance can be estimated to be around $400mm^2$. However it is important to note that these weights are just initial guesses. These weights can further be fine tuned by performing some test experiments.

7.2.4 Solution methods

As previously noted, the nonlinear weighted least squares problem is specified as finding the minimum of the weighted sum of the squares of the residuals, denoted as:

$$(x_a, y_a)^* = \text{argmin}_S(x_a, y_a) \quad (7.12)$$

$$= \text{argmin} \sum_{i=1}^n W_i e_i^2 \quad (7.13)$$

$$= \text{argmin} \sum_{i=1}^n W_i (d_i - \sqrt{(x_i - x_a)^2 + (y_i - y_a)^2})^2 \quad (7.14)$$

$$= \text{argmin} \sum_{i=1}^n W_i (d_i - f_i(x_a, y_a))^2 \quad (7.15)$$

For further mathematical ease we will use vector notation:

$$\mathbf{a} = [x_a, y_a]^T$$

$$\mathbf{f}(\mathbf{a}) = [f_1(\mathbf{a}), f_2(\mathbf{a}), \dots, f_n(\mathbf{a})]^T$$

$$\mathbf{d} = [d_1, d_2, \dots, d_n]^T$$

$$\mathbf{W} = [W_1, W_2, \dots, W_n]^T$$

Thus:

$$\mathbf{a}^* = \operatorname{argmin}_{\mathbf{a}} \sum_{i=1}^n W_i (d_i - f_i(\mathbf{a}))^2 \quad (7.16)$$

This minimum value is found where the gradient is zero. This means:

$$\frac{\partial S(\mathbf{a})}{\partial a_j} = \frac{\partial}{\partial a_j} \sum_{i=1}^n W_i (d_i - f_i(\mathbf{a}))^2 \quad (7.17)$$

$$= -2 \sum_{i=1}^n W_i [d_i - f_i(\mathbf{a})] \frac{\partial f_i(\mathbf{a})}{\partial a_j} \quad (7.18)$$

$$= -2 \sum_{i=1}^n W_i [d_i - f_i(\mathbf{a})] J_{ij} \quad (7.19)$$

$$(7.20)$$

Here \mathbf{J} is the Jacobian matrix and its matrix is defined as:

$$J_{ij} = \frac{\partial f_i(\mathbf{a})}{\partial a_j}$$

In vector form, we can denote 7.19:

$$-2\mathbf{J}^T \mathbf{W}(\mathbf{d} - \mathbf{f}(\mathbf{a})) = \mathbf{0} \quad (7.21)$$

However since $f(\mathbf{a})$ is nonlinear function, an analytically closed form solution can not be obtained. The optimal solution \mathbf{a}^* can be found by nonlinear optimization methods iteratively improving the solution. These numerical search methods require a good initialization in order to avoid converging in local minima instead of the global minimum [58]. This means an estimation of the anchor position should be known before any localization can be performed. Many methods such as the Levenberg–Marquardt method, Powell’s Dogleg method etc. have been proven to be effective [59], [60], however only the Gauss–Newton Method will be mathematically discussed here. This method is the basis for the other methods.

In the Gauss–Newton method, at each iteration k the position \mathbf{a}_k is refined:

$$\mathbf{a}_{k+1} = \mathbf{a}_k + \Delta \mathbf{a} \quad (7.22)$$

with $\Delta \mathbf{a}$ the shift vector. This vector is found by linearizing $f_i(\mathbf{a})$ between the steps. This is done by an approximation to a first-order Taylor polynomial expansion:

$$f_i(\mathbf{a}_{k+1}) \approx f_i(\mathbf{a}_k) + \sum_{j=1}^2 \frac{\partial f_i(\mathbf{a}_k)}{\partial a_j} \Delta a_j \quad (7.23)$$

$$\approx f_i(\mathbf{a}_k) + \sum_{j=1}^2 J_{ij} \Delta a_j \quad (7.24)$$

Or written in vector form:

$$\mathbf{f}(\mathbf{a}_{k+1}) \approx \mathbf{f}(\mathbf{a}_k) + \mathbf{J}\Delta\mathbf{a} \quad (7.25)$$

Now, when substituting 7.24 in 7.19 or more easily in vector form substituting 7.25 in 7.21 following is obtained:

$$(\mathbf{J}^T \mathbf{W} \mathbf{J}) \Delta \mathbf{a} = \mathbf{J}^T \mathbf{W} (\mathbf{d} - \mathbf{f}(\mathbf{a}_k)) \quad (7.26)$$

Now we can easily solve for $\Delta\mathbf{a}$:

$$\Delta\mathbf{a} = \mathbf{a}_{k+1} - \mathbf{a}_k = (\mathbf{J}^T \mathbf{W} \mathbf{J})^{-1} \mathbf{J} \mathbf{W} (\mathbf{d} - \mathbf{f}(\mathbf{a}_k)) \quad (7.27)$$

In Figure 7.2, this Gauss-Newton method is applied on a simple example.

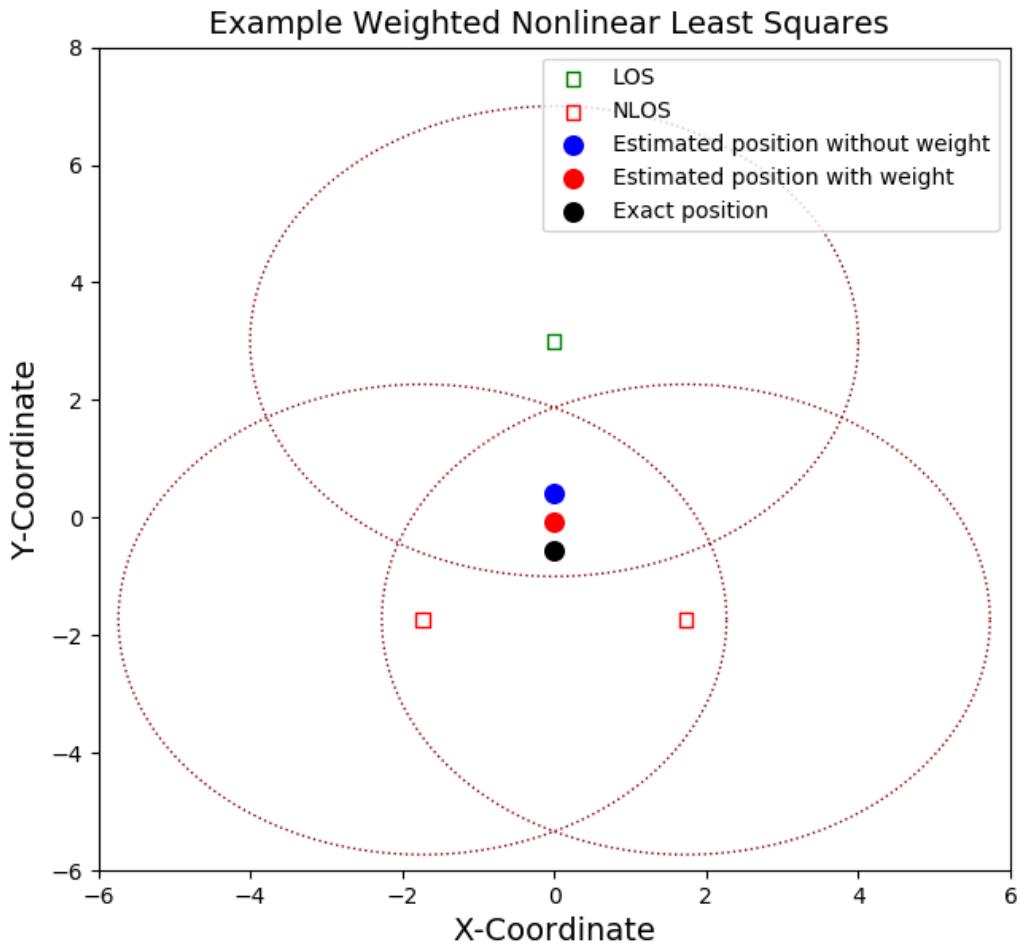


Figure 7.2: Example nonlinear weighted least squares for LOS and NLOS

Chapter 8

Drone Procedure

In this chapter, a drone procedure is proposed based on the findings in this thesis. The procedure needs the user to set a few waypoints after which the drone can perform the procedure. From the moment of lift off, no human intervention should be needed to complete the full procedure. Therefore obstacle avoidance, detection and decision structures to decide where to fly next are present. Next, to achieve accurate positioning, it is important to achieve a low GDOP. First, this concept will be introduced whereafter the small set-up will be explained. Finally, the proposed drone procedure will be discussed and its performance will be tested by initial simulations.

8.1 Geometric dilution of precision

In the previous chapter, a nonlinear weighted least squares algorithm is used to estimate the final position of the anchor. However not only the errors of distance and position have an influence on final position error, but also the specific geometry of the positions where measurements were made. It is thus stated that:

$$\Delta \text{AnchorPosition} = \text{GDOP} * \Delta \text{Measurements} \quad (8.1)$$

With:

- $\Delta \text{Measurements}$ the additive effect of the distance measurement errors still present after corrections and the RTK errors
- GDOP the geometric dilution of precision factor

First the principle of this GDOP will be explained from its original use in satellite positioning systems. After this, a mathematical framework will be given. Finally the interpretation of GDOP will be altered to the localization problem of the anchors.

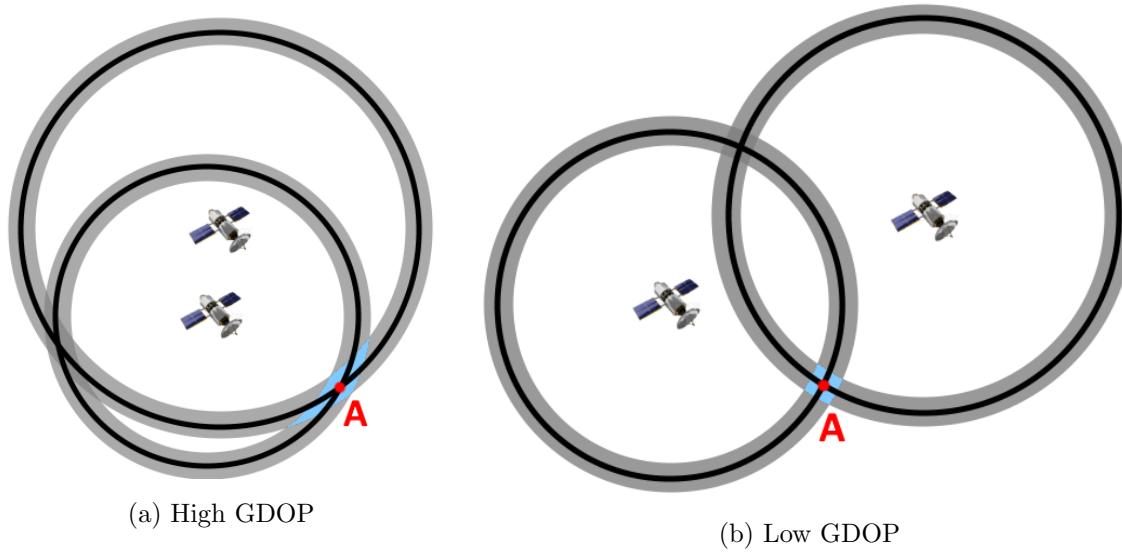


Figure 8.1: Two examples of measurement error in satellite navigation

8.1.1 Principle

Geometric dilution of precision (GDOP), a term used in satellite navigation, describes the additional multiplicative effect of the navigation satellite geometry on positional measurement precision. Looking at a 2D example in Figure 8.1, two satellites are used to locate an object A. Here the error on the distance measurements and on the positions of the satellites is represented by the grey area around the black circle. With the two satellites, the object A is assumed to be at the red dot, but could also be at any place in the blue area due to the errors. For a satellite arrangement as in Figure 8.1a, the blue area is widely extended and the error on position can be enormous, while in Figure 8.1b, the arrangement makes for a small error area. A good satellite geometry can thus significantly lower the error on the final estimated position. Smaller values for the GDOP signify better geometry placements.

8.1.2 Mathematical formulation GDOP

In order to calculate the GDOP value of a certain object, the coordinates of the object anchor and the satellites are used. First, unit vectors from satellite i to the object are calculated, denoted as v_i :

$$v_i = \left(\frac{x_i - x_a}{R_i}, \frac{y_i - y_a}{R_i}, \frac{z_i - z_a}{R_i} \right) \quad (8.2)$$

With (x_a, y_a, z_a) the coordinates of the object, (x_i, y_i, z_i) the coordinates of satellite i and R_i the distance between satellite i and the object formulated as:

$$R_i = \sqrt{(x_i - x_a)^2 + (y_i - y_a)^2 + (z_i - z_a)^2} \quad (8.3)$$

Now, a matrix M is formed using the calculated unit vectors:

$$M = \begin{bmatrix} \frac{x_1-x_a}{R_1} & \frac{y_1-y_a}{R_1} & \frac{z_1-z_a}{R_1} & 1 \\ \frac{x_2-x_a}{R_2} & \frac{y_2-y_a}{R_2} & \frac{z_2-z_a}{R_2} & 1 \\ \dots & \dots & \dots & \dots \\ \frac{x_n-x_a}{R_n} & \frac{y_n-y_a}{R_n} & \frac{z_n-z_a}{R_n} & 1 \end{bmatrix}$$

The covariance matrix G is then found by:

$$G = (M^T M)^{-1} \quad (8.4)$$

The GDOP is then:

$$GDOP = \text{tr}(G) \quad (8.5)$$

8.1.3 Applied on our procedure

This principle of GDOP can also be applied in our procedure in order to find optimum positions to measure the location of the anchor. Here, the satellite positions are now the positions at which the drone already has measured and a next possible candidate point. It is important to note that the mathematical framework needs the coordinates of the anchor. In the procedure an estimate of these coordinates is given to the algorithm.

8.2 Procedure set-up

Given that the drone should not start randomly flying around when powered on, minor inputs from the drone operator are needed. The first one is a set of waypoints. These are the points that should be followed by the drone and give a rough overview where the drone will fly. Following this, the user can show the contours of the building where the anchors are and help the calibration procedure by putting more waypoints where anchors are located.

For example, let us look at Figure 8.2, where a fictive building with anchors to be calibrated is shown. Based on the ground plan of the building, 24 waypoints are appointed by the operator. Also, the drone operator can choose to make the waypoint distribution more dense in places where it is known more anchors are located. In this way, more measurements will be performed in these areas. If this knowledge of the anchor location is not known, all waypoints could be assigned a fixed interval.

8.3 Procedure

A flow chart of the proposed procedure is shown in Figure 8.3. The consequent steps are shown with full lines, while the data transfers are shown in dotted lines. Each of the bold

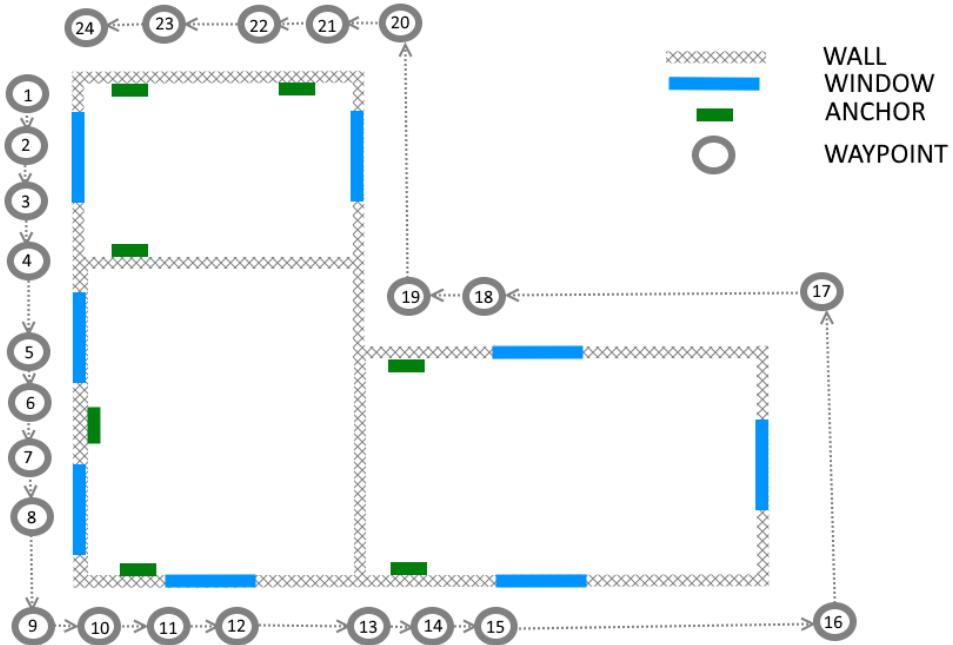


Figure 8.2: Schematic overview of waypoints

actions will be explained in the following subsections.

8.3.1 Start at waypoint 1

This is the first step of the procedure. The drone flies to the first waypoint and starts the rest of the procedure as the waypoint is reached.

8.3.2 Search anchors

In this step, the Pozyx device attached to the drone searches for the anchors. If no anchor is found, the drone will investigate the direct environment in the “move around”-action. If at least one anchor is found, the ranging can be initiated.

The remainder of the explanation will be done as if only one anchor is present. However, with multiple anchors, the procedure runs parallel from the “perform ranging”-step.

8.3.3 Move around

If no anchor is found, the drone will attempt to find an anchor by moving in all three directions. This step is to anticipate situations where the signal to the receiver for an undetermined reason is blocked in a small region, but is not in the region next to it.

If after this still no signal is found, the drone engages the “move towards next waypoint”.

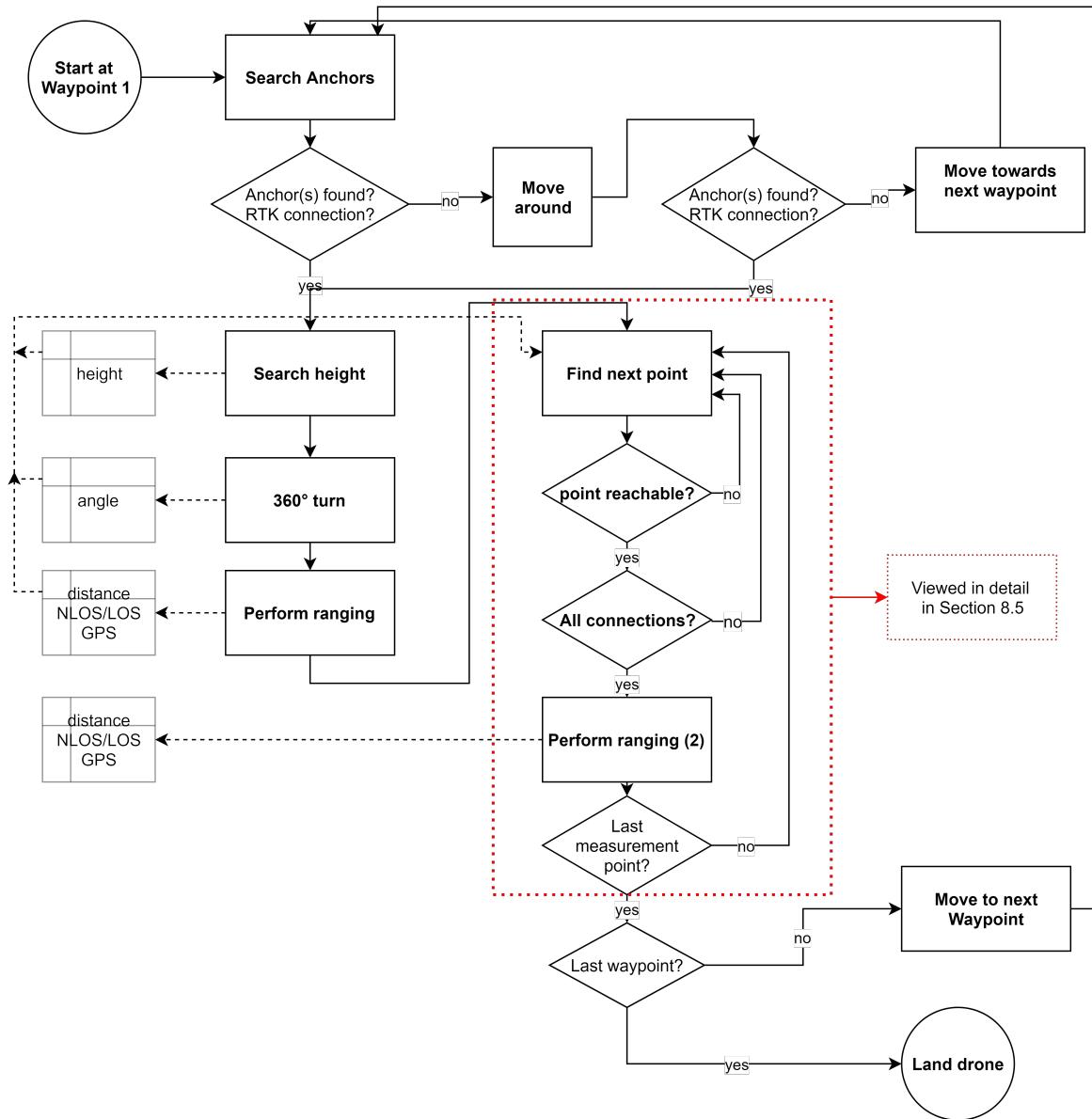


Figure 8.3: Flow chart of the proposed drone procedure

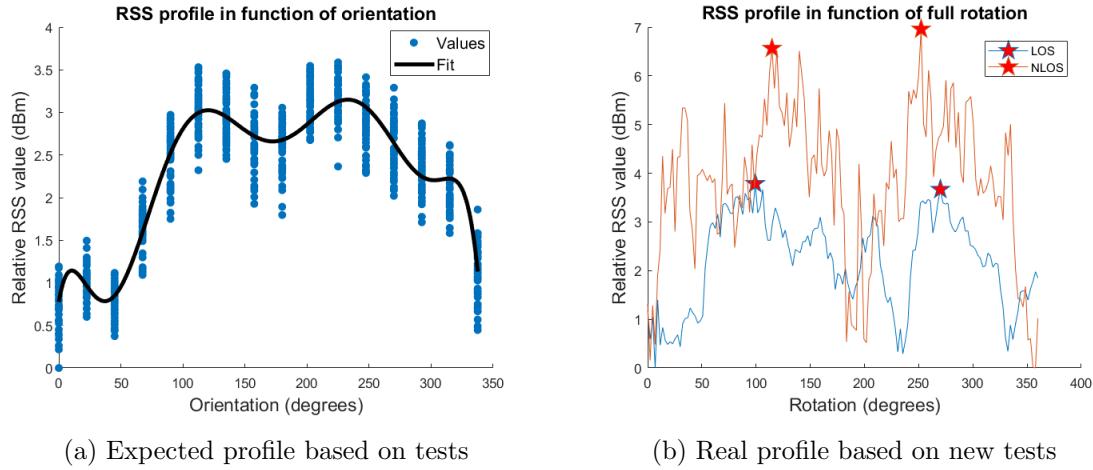


Figure 8.4: Search for RSS profile in function of orientation

8.3.4 Move towards next waypoint

If this step is reached, the former measurement point had no connection with an anchor. In an attempt to solve this, the drone is moved towards a next coordinate with a distance proportional with the distance between waypoint WP_i and WP_{i+1} with a proportionally factor k :

$$(x, y, z)_{next} = \frac{1}{k} \cdot (x_{WP_{i+1}} - x_{WP_i}, y_{WP_{i+1}} - y_{WP_i}, z_{WP_{i+1}} - z_{WP_i}) \quad (8.6)$$

This integer factor k can be chosen arbitrarily and makes sure that after k failed attempts to make connection to an anchor, the drone automatically arrives in the next waypoint.

8.3.5 360° turn

Here, the drone is ordered to make a full turn around its z-axis. This is a consequence of the typical shape that was noticed for the RSS-values in function of the orientation. Independent of height difference or mutual distance, the RSS in function of the orientation always has a similar shape, as can be seen in Figure 8.4a. Here the minimum RSS value per test is subtracted from each measurement to show the course of the profile independent of the mean RSS. This corresponds with the findings in [61]. Based on Figure 8.4a, the values of the RSS appear to have a maximum around 120° and 250°. So, a simple algorithm searching for the peaks could match this profile and provide an estimate for the orientation of the anchor. Both in the LOS and NLOS case, this technique was simulated in Figure 8.4b, showing the peaks of the signal. Based on these profiles, this method should be able to identify the angle of the signal with an accuracy of about 20°.

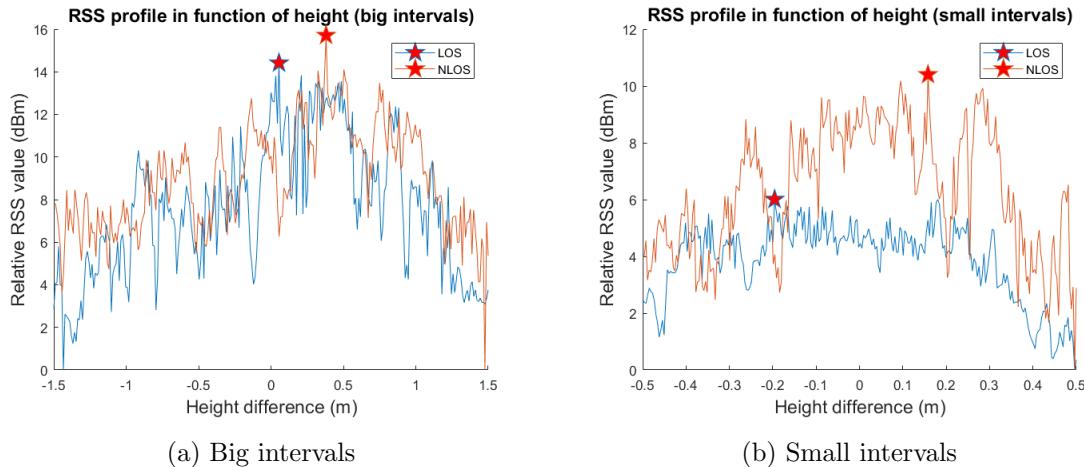


Figure 8.5: Search of maximum value for RSS in function of height displacement

8.3.6 Search height

In this step, the drones moves both upwards and downwards in search for the position with the best RSS value. Regardless the orientation, distance or environment; if the UWB antenna is facing upwards (as was discussed in Section 4.2.2), the RSS value tends to be stronger when the height difference between the device is zero i.e. when they are on the same height. This step assumes the height of the anchor and is very useful to reduce the complexity of the calibration. To prove this assumption, tests were performed for both LOS and NLOS measurements where the height of a receiver was varied from $1.5m$ above the anchor to $1.5m$ under the anchor (Figure 8.5a). One would expect the RSS value to be maximum at $0m$, which is the case in LOS, but for NLOS this method performs less ideal. However, it provides a first estimate.

To test how accurate this method could work, the same procedure was repeated for smaller differences in height, (from $-0.5m$ to $0.5m$). Here, the peaks identified the height with an accuracy up to $20cm$, as can be seen in Figure 8.5b.

8.3.7 Perform ranging

In this step, a connection with an anchor is established and the ranging is started. Next to a value for the measured distance, also the GPS connection and the NLOS/LOS-identification is saved.

8.3.8 Find next point

In this step, the CPU calculates the next point for the drone to fly to. This is the most complicated step and will be explained in its own Section 8.4.

8.3.9 Point reachable

There should be a safety mechanism in place if a point is not reachable due to obstructions or any other reason. Then, the drone moves back to its former point and remembers this point was not reachable. In the former “Find next point”-algorithm, now a new drone position is generated.

8.3.10 All connections

If the point is reachable, but no connection can be made with either the UWB anchors, RTK rover or CPU, the drone flies back to its former position, remembering not to turn back to this point.

8.3.11 Move to next waypoint

When enough valuable datapoints are gathered at a waypoint, the drone is moved to the next waypoint.

8.3.12 Land drone

When all waypoints are visited, the drone can land and the procedure has finished.

8.4 Best point to fly to

In this section, the core part of the drone procedure is viewed in detail, namely the decision on where to fly next to achieve an accurate calibration. A flow chart is available in Figure 8.6 and the different parts will be explained in the following subsections.

8.4.1 Estimate anchor position (1)

After the first measurement, it is possible to make a gross estimate for the anchor position. Given that the coordinates of the drone $(x, y, z)_{drone}$ are known, using the former methods, a first guess can be made for the position of the anchor $(x, y, z)_{anchor}$:

$$\begin{aligned} x_{anchor} &= x_{drone} + d_{meas} \cdot \cos \theta_{meas} \\ y_{anchor} &= y_{drone} + d_{meas} \cdot \sin \theta_{meas} \\ z_{anchor} &= z_{meas} \end{aligned} \tag{8.7}$$

Where d_{meas} is the measured distance from the *Perform ranging*-method, θ_{meas} is the guessed angle from the 360° *turn*-method and z_{meas} the assumed height from the *Search height*-method. It has to be noted that due to inaccuracies, the found position for the anchor will not be accurate. However, it provides an initial guess, which will be used to find the best next position for the drone to fly to.

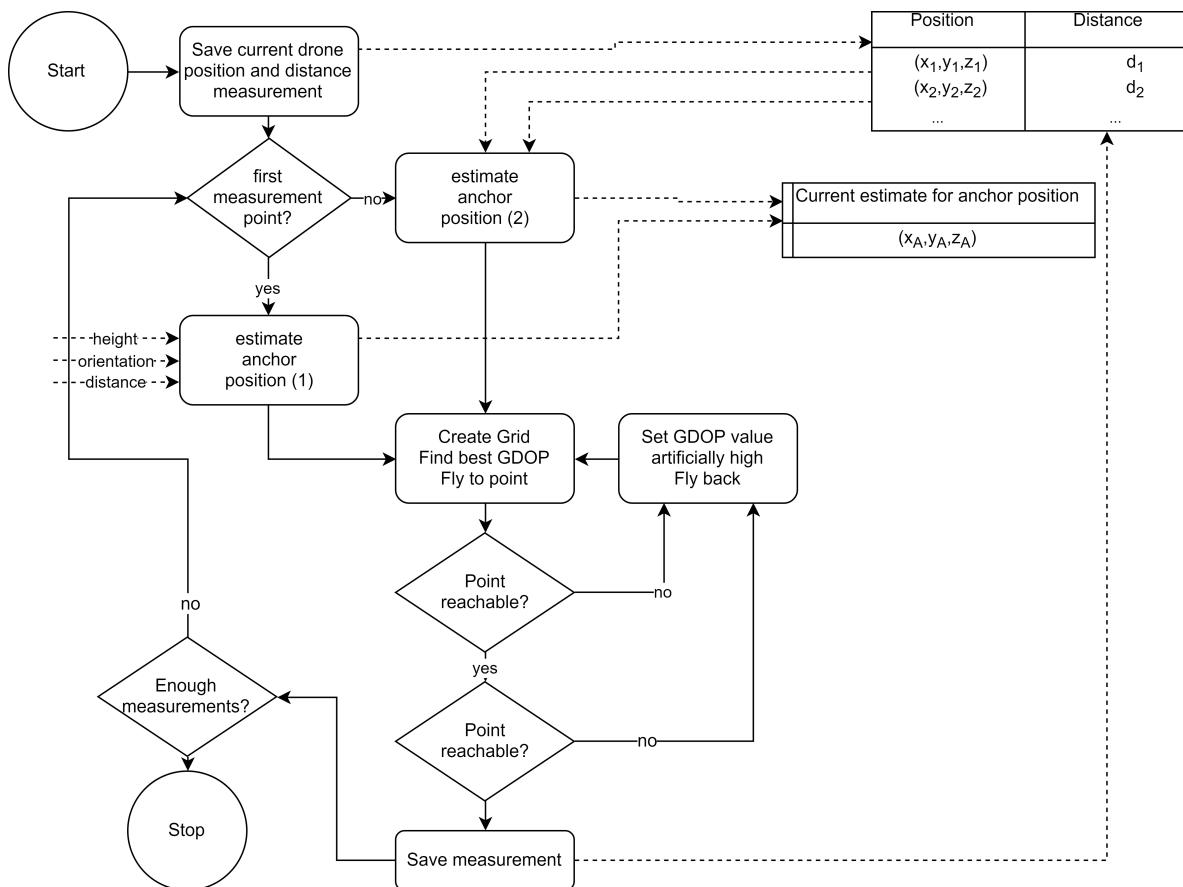


Figure 8.6: Detailed flow chart of the “find next point”-step. The full lines present subsequent steps, the dotted lines datatransfers.

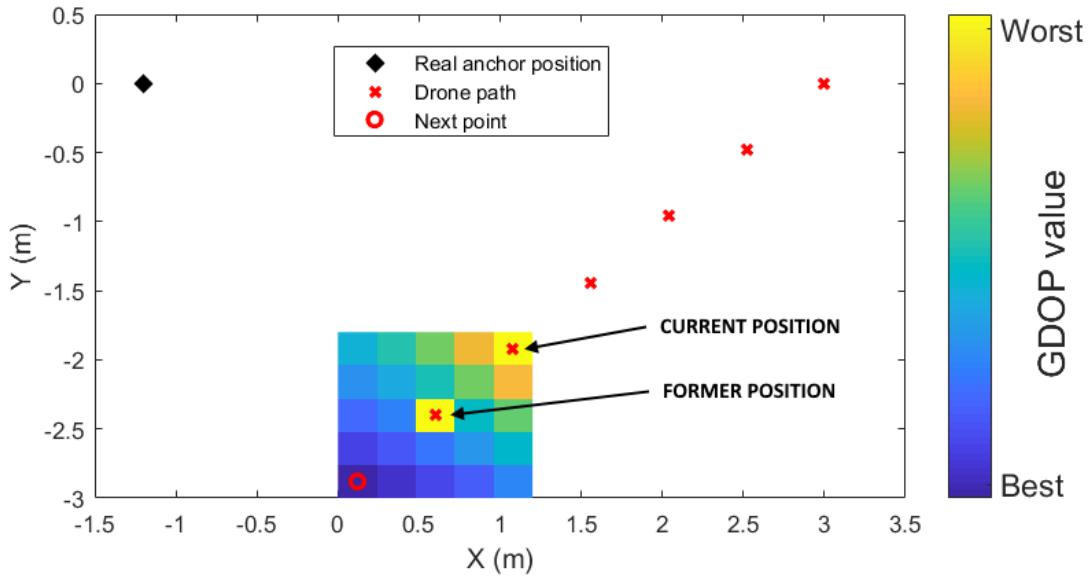


Figure 8.7: Application of the grid to search for the best point to fly to

In the following, it will be assumed that the z_{anchor} is perfectly known, reducing the problem to 2 dimensions for the explanation. The whole procedure is as easily transferred to three dimensions.

8.4.2 Estimate anchor position (2)

As soon as two positions and distances are available, a new anchor estimate can be estimated using the intersection of the spheres. The first initial guess of the anchor position is used to prevent from converging into the wrong intersection.

8.4.3 Create grid

In this step, a grid is created around the current position of the drone. Then, for each point of this grid, the GDOP using the combination of all past positions and the position of this point is calculated. Next, the point which results in the lowest GDOP value is taken. If this is a point that has already been visited, the GDOP value of this point is set artificially high, and the next is taken. If the point has not yet been visited, the drone moves to this point. An example of this step is visible in Figure 8.7. Here, the earlier visited positions are marked with the red cross. The drone is located at the last cross, and a grid is made surrounding this point. For each point, the GDOP value is calculated. The worst GDOP values are found for the current and former positions, and the best is found in the left bottom corner, resulting the drone to fly to this point.

If a point is not reachable or the connection with one of the used technologies can not be established, the drone flies back and the GDOP value for this position is set artificially high, making sure that in the next iteration another point will be chosen.

Next, the measurements are taken and saved and the procedure restarted.

8.4.4 Enough measurements

To prevent the drone from iterating this subprocedure infinitely, an exit procedure is provided. As soon as there are enough measurements captured (either by the total number of iterations or by the GDOP number being sufficiently low), this subprocedure is terminated and the next waypoint in the major procedure is visited.

8.5 Results

To prove the working of the proposed drone procedure, in this section, simple tests will be applied accompanied by some assumptions. As discussed before, no RTK hardware was available, as was a controllable drone. Therefore, for testing purposes, two Pozyx devices were put on tripods at a fixed height, assuming that the proposed *search height* has found the right height. This results in the problem being reduced to 2 dimensions. To simulate the RTK coordinates, a (x, y) -grid was applied. In this grid, the fixed Pozyx acts as a to be calibrated anchor, the movable Pozyx as representation of the drone.

Figure 8.8 shows a visual representation of the procedure and the associated localization algorithm. The drone was initially placed in $(3, 0)$ and the anchor is located at $(-1.2, 0)$. With a rough estimate of the anchor position and a grid size of 0.48, the following point $(2.52, -0.48)$ was visited, based on the maximum improvement of the GDOP. Per iteration, the GDOP was improved, shown in Figure 8.9. A wall was placed at $x = -0.5$, resulting in the drone not being allowed to approach near the wall or equivalently, setting the GDOP values for these positions artificially high, as a test for the *point reachable*-step. The procedure followed accordingly and from iteration 7, the wall was followed instead of crashing against it. A note on this is that the decision for simulating the algorithm in 2D puts restrictions on the best achievable GDOP. In 3D, more grid points will be available and a faster decrease of this GDOP will be obtained.

This exact procedure was followed for both LOS and NLOS measurements. Since in both cases, the initial guess for the anchor position was close to the real position, the same path was followed for both. The results are shown in Figure 8.10 for the LOS case and in Figure 8.11 for the NLOS case. For each iteration, the following is shown:

1. The position error for corrected and uncorrected measurements following from the localization algorithm
2. The distance error and its correction following the error mitigation
3. The identification of LOS or NLOS based on the machine learning classification

LOS For the LOS test, all measurements were correctly identified, except the first one. As this first one was identified as NLOS, the NLOS mitigation method was used, resulting in a corrected distance worse than the uncorrected distance. For the others, each measured distance was correctly improved by the LOS mitigation method. This resulted in a final accuracy of the localization algorithm of 1.3cm . This is an improvement on the accuracy for the position of 6.1cm using uncorrected distance measurements. The initial position error is high since here, only 2 measurements can be used, resulting in an inaccurate position estimate.

NLOS Due to the higher error variance in NLOS cases, the accuracy of the final estimated position is, as expected, lower than the LOS test. However, still a high improvement was made correcting the distance measurements by the NLOS error mitigation method. The accuracy of the position using uncorrected distance measurements is around 26cm , while the accuracy using corrected measurements is 9cm . It is important to note that the identification was perfect and no False Predictions were made, which is to be avoided at all cost.

Conclusion In both cases, an improvement is achieved. However it should be noted that this improvement will be less when also the error of the RTK and vibrations of the drone are present. In these 2 performed tests, the algorithm and procedure work well. However, more tests should be executed to test the procedure on a larger scale of environments.

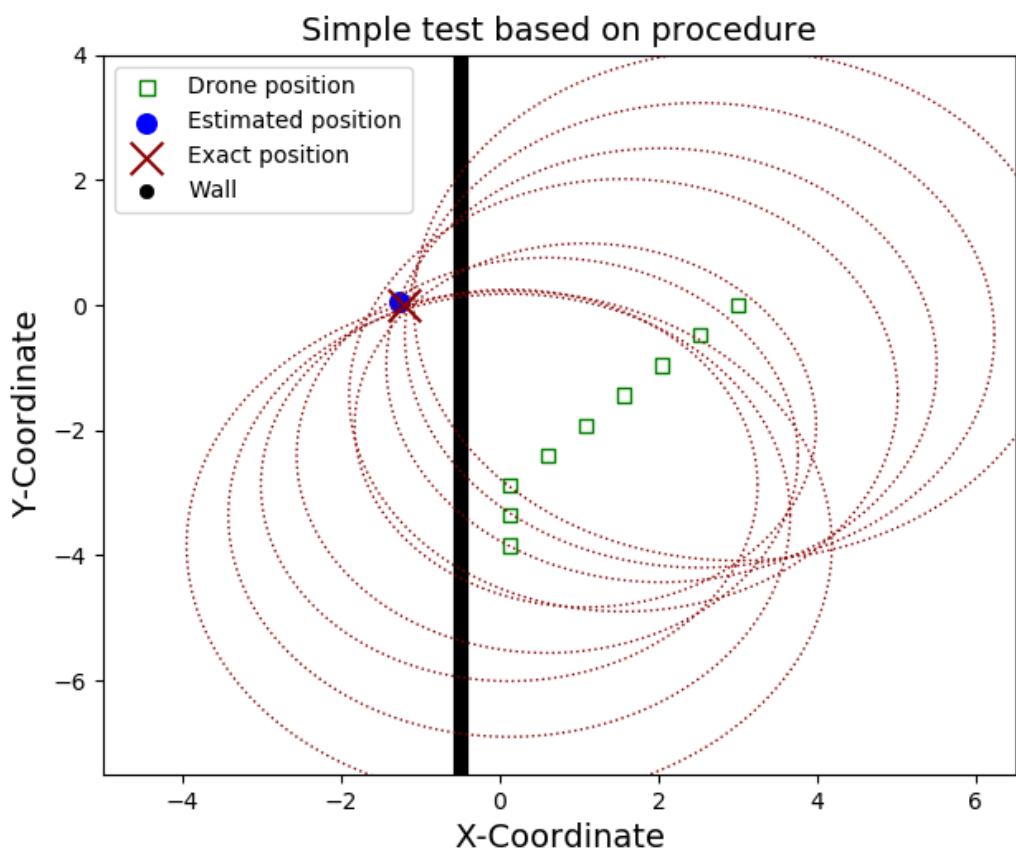


Figure 8.8: Simple test on the proposed procedure

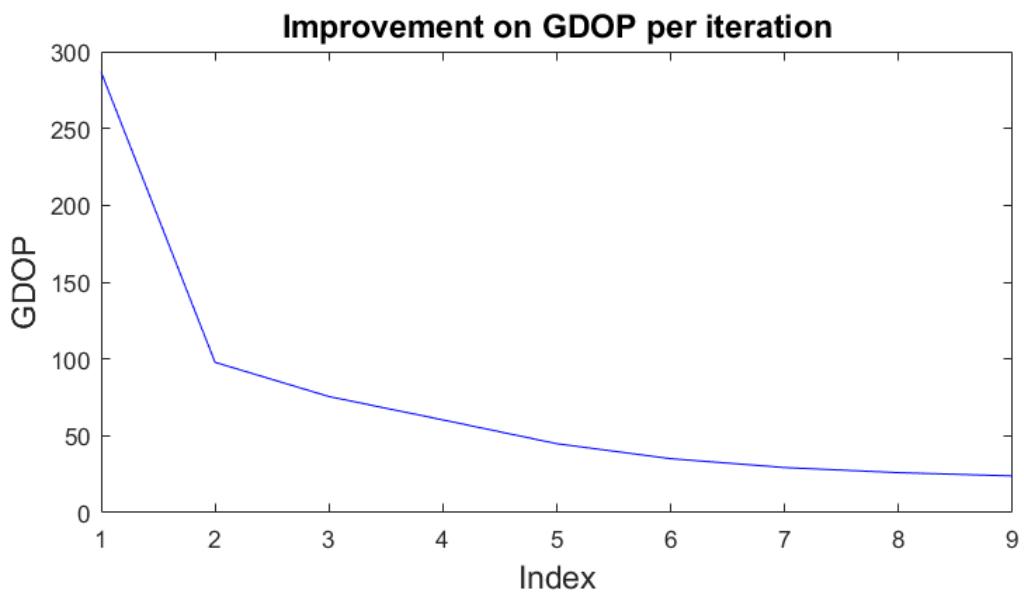


Figure 8.9: GDOP Improvement

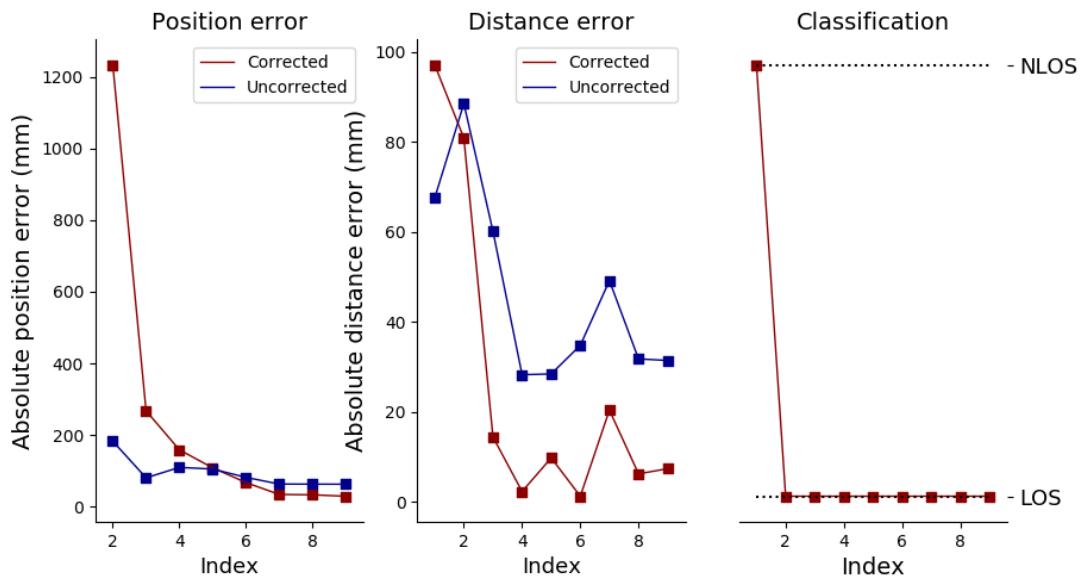


Figure 8.10: Results of the procedure in a Line-of-Sight situation

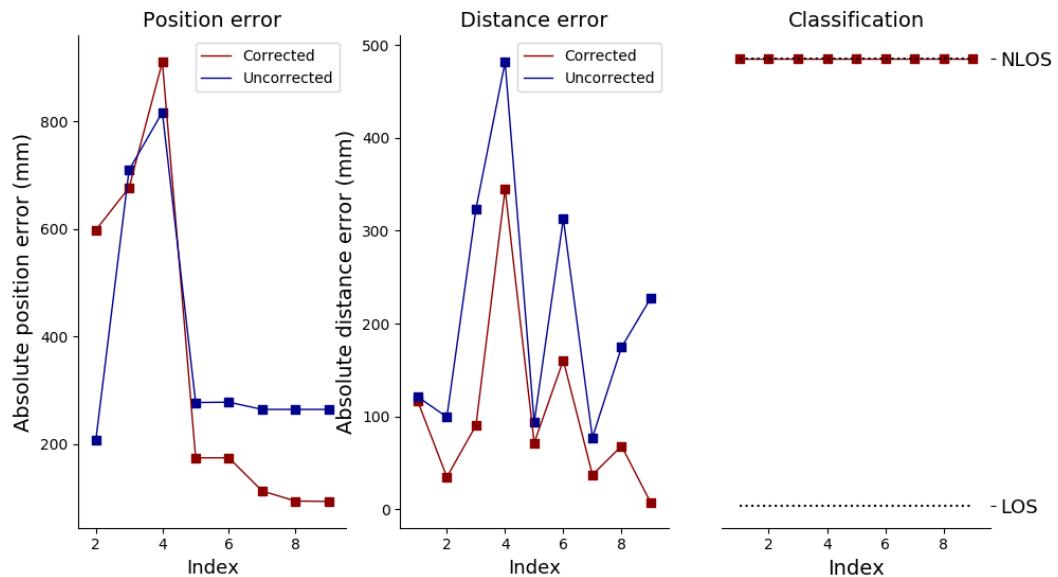


Figure 8.11: Results of the procedure in a Non-Line-of-Sight situation

Chapter 9

Conclusion and future prospects

In this thesis, a first step was taken towards a fully autonomous calibration procedure. First, in order to correct the measurements, a machine learning classification was used to identify an LOS or NLOS measurement based on many identification features. Based on this identification, the measurements are corrected in a machine learning regression algorithm.

Next, a weighted localization method was explained based on the (already corrected) measurements from the drone. Here, the weights are based on the identification of LOS/NLOS. Higher weights are given to LOS measurements because these are more reliable and accurate than NLOS measurements.

Finally, a first procedure was proposed. The reason not the whole procedure could be put on a actual drone was due to it being impossible to automatically control the drone and due to not yet having received the RTK hardware.

However, moving on with our more limited procedure, initial simulations of (parts of) the procedure proved that the bias was significantly improved in LOS measurements and as such an accuracy in the range of 2cm for the position could be achieved (assuming the coordinates are known exactly). For NLOS measurements, a lower accuracy in the range of 9cm was obtained, but these can be improved by generating more training data for the machine learning.

This thesis counts as a sound basis for a fully autonomous procedure. The part of the procedure for having accurate and improved measurements has already been tackled and a first algorithm has been proposed. Future work however is needed to transform the theoretical algorithm into code for a automatically controllable drone and removing the last operator input - the waypoints - from the algorithm without letting the drone fly away to infinity and beyond. Complementary decision blocks will be necessary when using the procedure in real life.

Bibliography

- [1] GPS Accuracy: HDOP, PDOP, GDOP, Multipath & the Atmosphere. (2018, February 24). Retrieved from <https://gisgeography.com/gps-accuracy-hdop-pdop-gdop-multipath>
- [2] Ozsoy, K., Bozkurt, A., & Tekin, I. (2013). Indoor positioning based on global positioning system signals. *Microwave and Optical Technology Letters*, 55(5), 1091-1097.
- [3] IndoorAtlas, VansonBourne (2016). A 2016 global research report on the indoor positioning market.
- [4] Mona Lisa 2.0 (n.d.). Indoor positioning system: Requirement definition.
- [5] Brena, R. F., García-Vázquez, J. P., Galván-Tejada, C. E., Muñoz-Rodriguez, D., Vargas-Rosales, C., & Fangmeyer, J. (2017). Evolution of indoor positioning technologies: A survey. *Journal of Sensors*, 2017.
- [6] Alarifi, A., Al-Salman, A., Alsaleh, M., Alnafessah, A., Al-Hadhrami, S., Al-Ammar, M. A., & Al-Khalifa, H. S. (2016). Ultra wideband indoor positioning technologies: Analysis and recent advances. *Sensors*, 16(5), 707.
- [7] Mautz, R. (2009). Overview of current indoor positioning systems. *Geodezija ir kartografija*, 35(1), 18-22.
- [8] Cheng, T. *Real-time Automated Construction Worker Location Tracking for Spatio-Temporal Safety Analysis and Feedback* (Doctoral dissertation, School of Electrical and Computer Sciences, Georgia Institute of Technology).
- [9] DecaWave, Darmon, L., (2014, April). Zeroing In: UWB Technology Offers Greater Accuracy for Real-Time Location Services. Retrieved from https://www.decawave.com/sites/default/files/psqh_april_2014.pdf
- [10] Kessel, M., Werner, M., Gschwandtner, F., & Linnhoff-Popien, C. (n.d.). Indoor positioning: Tracking and navigation for goods and people.

- [11] Lindemann, A., Schnor, B., Sohre, J., & Vogel, P. (2016, February). Indoor Positioning: A Comparison of WiFi and Bluetooth Low Energy for Region Monitoring. In *HEALTHINF* (pp. 314-321).
- [12] Ghavami, M., Michael, L., & Kohno, R. (2007). *Ultra wideband signals and systems in communication engineering*. John Wiley & Sons.
- [13] Mautz, R. (2012). Indoor positioning technologies.
- [14] Hol, J. D., Schön, T. B., & Gustafsson, F. (2010, September). Ultra-wideband calibration for indoor positioning. In *Ultra-Wideband (ICUWB), 2010 IEEE International Conference on* (Vol. 2, pp. 1-4). IEEE.
- [15] Van de Velde, S., Van Torre, P., & Steendam, H. (2013, December). Fast and robust anchor calibration in range-based wireless localization. In *Signal Processing and Communication Systems (ICSPCS), 2013 7th International Conference on* (pp. 1-6). IEEE.
- [16] Ran, L., Helal, S., & Moore, S. (2004, March). Drishti: an integrated indoor/outdoor blind navigation system and service. In *Pervasive Computing and Communications, 2004. PerCom 2004. Proceedings of the Second IEEE Annual Conference on* (pp. 23-30). IEEE.
- [17] Hausman, K., Weiss, S., Brockers, R., Matthies, L., & Sukhatme, G. S. (2016, May). Self-calibrating multi-sensor fusion with probabilistic measurement validation for seamless sensor switching on a UAV. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on* (pp. 4289-4296). IEEE.
- [18] Vanclooster, A., & De Maeyer, P. (2012). Combining indoor and outdoor navigation: the current approach of route planners. In *Advances in Location-Based Services* (pp. 283-303). Springer, Berlin, Heidelberg.
- [19] Nawaz, S., & Jha, S. (2007). Cooperative relative positioning for wireless sensor networks. In *1st International Workshop on Localized Algorithms and Protocols for WSN. Santa Fe (USA)*.
- [20] Chen, H., Huang, P., So, H. C., Luo, X., & Deng, P. (2008, November). Mobility-assisted cooperative localization scheme for wireless sensor networks. In *Military Communications Conference, 2008. MILCOM 2008. IEEE* (pp. 1-7). IEEE.
- [21] Gentner, C., & Ulmschneider, M. (2017). Simultaneous Localization and Mapping for Pedestrians using Low-Cost Ultra-Wideband System and Gyroscope.
- [22] Strömberg, E. (2017). Smoothing and Mapping of an Unmanned Aerial Vehicle Using Ultra-wideband Sensors.

- [23] Gakstatter, E. (2014). Centimeter-Level RTK accuracy more and more available—for less and less. *GPS World*, 4.
- [24] Breed, G. (2005). A summary of FCC rules for ultra wideband communications. *High Frequency Electronics*, 4(1), 42-44.
- [25] Haraz, O. (2012). Why do we need ultra-wideband?.
- [26] Pozyx. (n.d.). How does ultra-wideband work. Retrieved from https://www.pozyx.io/Documentation/how_does_uwb_work
- [27] McCroskey, R. (2011). Pulsed-rf ultrawideband ranging for the glanser gps-denied emergency responder navigation system.
- [28] Decawave. (2015). Sources of error in DW1000 based two-way ranging (TWR) schemes. Retrieved from https://www.decawave.com/sites/default/files/resources/aps011_sources_of_error_in_twr.pdf
- [29] Decawave. (2015). DW1000 Datasheet. Retrieved from <https://www.decawave.com/sites/default/files/resources/dw1000-datasheet-v2.09.pdf>
- [30] Pozyx. (n.d.). Where to place the anchors. Retrieved from https://www.pozyx.io/Documentation/where_to_place_the_anchors
- [31] Decawave. (2016). Metrics for Estimation of Non Line Of Sight Operating Conditions. Retrieved from https://www.decawave.com/sites/default/files/aps006_part_3_dw1000_diagnostics_for_nlos_channels.pdf
- [32] Dardari, D., Conti, A., Ferner, U., Giorgetti, A., & Win, M. Z. (2009). Ranging with ultrawide bandwidth signals in multipath environments. *Proceedings of the IEEE*, 97(2), 404-426.
- [33] Novatel. (n.d.). An introduction to GNSS. Chapter 5 - Resolving Errors. Retrieved from <https://www.novatel.com/an-introduction-to-gnss/chapter-5-resolving-errors/real-time-kinematic-rtk/>
- [34] Sachs, J. (2013). *Handbook of ultra-wideband short-range sensing: theory, sensors, applications*. John Wiley & Sons.
- [35] Muqaibel, A., Safaai-Jazi, A., Woerner, B., & Riad, S. (2002, August). UWB channel impulse response characterization using deconvolution techniques. In Circuits and Systems, 2002. *MWSCAS-2002. The 2002 45th Midwest Symposium on* (Vol. 3, pp. III-605). IEEE.

- [36] Venkatesh, S., & Buehrer, R. M. (2007). Non-Line-of-Sight identification in ultra-wideband systems based on received signal statistics. *IET Microwaves, Antennas & Propagation*, 1(6), 1120-1130.
- [37] Xiao, Z., Wen, H., Markham, A., Trigoni, N., Blunsom, P., & Frolik, J. (2015). Non-Line-of-Sight identification and mitigation using received signal strength. *IEEE Transactions on Wireless Communications*, 14(3), 1689-1702.
- [38] Güvenç, I., Chong, C. C., Watanabe, F., & Inamura, H. (2007). NLOS identification and weighted least-squares localization for UWB systems using multipath channel statistics. *EURASIP Journal on Advances in Signal Processing*, 2008(1), 271984.
- [39] Xiao, Z., Wen, H., Markham, A., Trigoni, N., Blunsom, P., & Frolik, J. (2013, October). Identification and mitigation of non-Line-of-Sight conditions using received signal strength. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2013 IEEE 9th International Conference on* (pp. 667-674). IEEE.
- [40] Wylie, M. P., & Holtzman, J. (1996, September). The non-line of sight problem in mobile location estimation. In *Universal Personal Communications, 1996. Record., 1996 5th IEEE International Conference on* (Vol. 2, pp. 827-831). IEEE.
- [41] Gururaj, K., Rajendra, A. K., Song, Y., Law, C. L., & Cai, G. (2017, September). Real-time identification of NLOS range measurements for enhanced UWB localization. In *Indoor Positioning and Indoor Navigation (IPIN), 2017 International Conference on* (pp. 1-7). IEEE.
- [42] Kolakowski, M., & Modelska, J. (2017, November). First path component power based NLOS mitigation in UWB positioning system. In *Telecommunication Forum (TELFOR), 2017 25th* (pp. 1-4). IEEE.
- [43] Overfitting. (1930). In OxfordDictionaries.com. Retrieved from <https://en.oxforddictionaries.com/definition/overfitting>
- [44] Ripley, B. D. (2007). *Pattern recognition and neural networks* (pp. 354). Cambridge university press.
- [45] Daniyal, P. (2017). Statistics for Machine Learning: Techniques for exploring supervised, unsupervised, and reinforcement learning models with Python and R.
- [46] Fortmann-Roe, S. (2012). Understanding the bias-variance tradeoff.
- [47] Denil, M., Matheson, D., & De Freitas, N. (2014). Narrowing the gap: Random forests in theory and in practice. In *International conference on machine learning (ICML)*.

- [48] Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning* (Vol. 1, pp. 337-387). New York: Springer series in statistics.
- [49] Breiman, L. (2001). Random forests. *Machine learning*, 45(1), 5-32.
- [50] Bernard, S., Heutte, L., & Adam, S. (2009, June). Influence of hyperparameters on random forest accuracy. In *International Workshop on Multiple Classifier Systems* (pp. 171-180). Springer, Berlin, Heidelberg.
- [51] Oshiro, T. M., Perez, P. S., & Baranauskas, J. A. (2012, July). How many trees in a random forest?. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition* (pp. 154-168). Springer, Berlin, Heidelberg.
- [52] Khanna, D., Sahu, R., Baths, V., & Deshpande, B. (2015). Comparative Study of Classification Techniques (SVM, Logistic Regression and Neural Networks) to Predict the Prevalence of Heart Disease. *International Journal of Machine Learning and Computing*, 5(5), 414.
- [53] Kim. Yu, H., & Kim, S. (2012). Svm tutorial—classification, regression and ranking. In *Handbook of Natural computing* (pp. 479-506). Springer Berlin Heidelberg.
- [54] Bouzgou, H., & Benoudjit, N. (2011). Multiple architecture system for wind speed prediction. *Applied Energy*, 88(7), 2463-2471.
- [55] Cherkassky, V., & Ma, Y. (2002, August). Selection of meta-parameters for support vector regression. In *International Conference on Artificial Neural Networks* (pp. 687-693). Springer, Berlin, Heidelberg.
- [56] Bermingham, M. L., Pong-Wong, R., Spiliopoulou, A., Hayward, C., Rudan, I., Campbell, H., ... & Haley, C. S. (2015). Application of high-dimensional feature selection: evaluation for genomic prediction in man. *Scientific reports*, 5, 10312.
- [57] Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar), 1157-1182.
- [58] Jun, Z. G., Xin, L., Long, X. Z., & Chao, L. H. (2015, September). Weighted least square localization algorithm based on RSSI values. In *Instrumentation and Measurement, Computer, Communication and Control (IMCCC), 2015 Fifth International Conference on* (pp. 1236-1239). IEEE.
- [59] Kabir, M. H., & Kohno, R. (2012). A hybrid TOA-fingerprinting based localization of mobile nodes using UWB signaling for non Line-of-Sight conditions. *Sensors*, 12(8), 11187-11204.

- [60] Madsen, K., Nielsen, H. B., & Tingleff, O. (1999). Methods for non-linear least squares problems.
- [61] Tiemann, J., Pillmann, J., & Wietfeld, C. (2017, June). Ultra-Wideband Antenna-Induced Error Prediction Using Deep Learning on Channel Response Data. In *Vehicular Technology Conference (VTC Spring), 2017 IEEE 85th* (pp. 1-5). IEEE.

Appendix A

Appendix

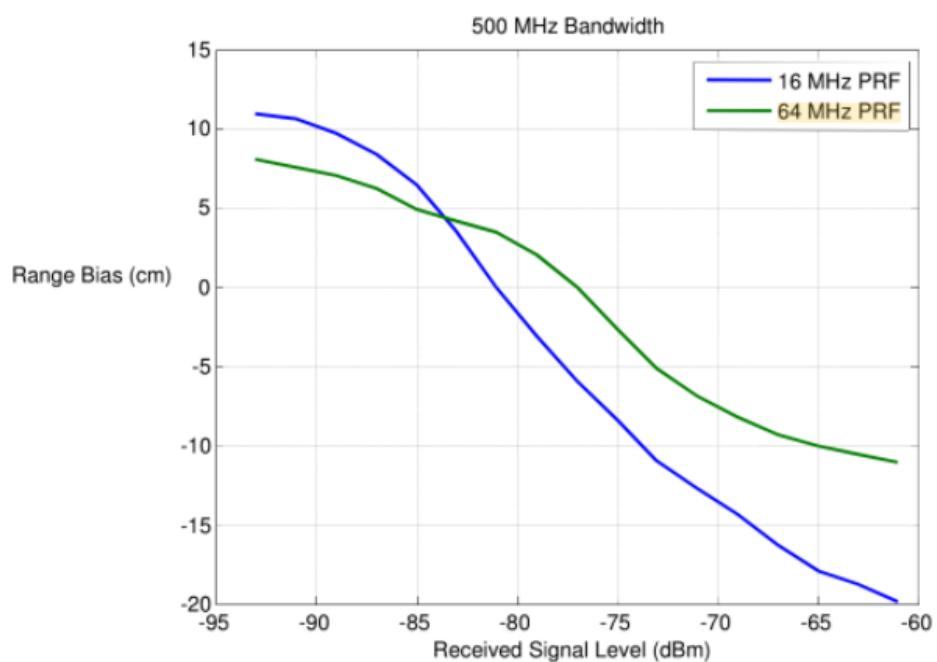


Figure A.1: Relation between bias and signal strength, indicated by DW1000 user manual [28]

Technology	Approx. accu.	Coverage	Cost		Strengths	Weaknesses
			IC	UC		
Technologies with signal encoding						
Infrared	57 cm-2.3 m	Room	H	L	Cheap for user	Sunlight interference
VLC	10 cm	Building (ML)	H	L	Cheap for user, unintrusive	Expens. infrast.
Ultrasonic	1 cm-2 m	Room	H	H	Good precision	Cost, interfer.
Audible sound	Meters	Room	L	L	Low cost	Low precision
Wi-Fi	1.5 m	Building	L	L	Low cost, good precision	Vulnerable to access point changes
Bluetooth	30 cm-meters	Building	L	L	Low cost, good precision	Intrusive; needs signal mapping
ZigBee	25 cm	Building	L	H	Could reuse infrastructure	Low precision; user needs special equip.
RFID	1-5 m	Room	H	L	Very low cost passive side	Very low precision
UWB	15 cm	Building	H	H	High precision	High cost
Passive technologies without signal encoding						
Geomagnetic	2 m	—	L	L	No need for infrastructure, good precision	Requires mapping
Inertial	2 m	—	L	L	Low cost, private	Accumulates error
Ambient sound	Meters	—	L	L	Cheap, not intrusive	Not accurate, sensitive to changes
Ambient light	10 cm-meters	—	L	L	Cheap	Sensitive to sunlight and changes such as a bulb and a window
Computer vision	1 cm-1 m	—	L	L	Low cost, privacy if cellphone camera is used	Sensitive to light conditions

Figure A.2: Indoor positioning technologies [5]