Tampere University

Carlos Mariano Castillo Mateos

RF-BASED AUTOMATED UAV ORIENTATION AND LANDING SYSTEM

Faculty of Information Technology and Communication Sciences (ITC)
Master Thesis
May 2019

ABSTRACT

Carlos Mariano Castillo Mateos: RF-based automated UAV orientation and landing system
Master Thesis
Tampere University
Master Degree Programme in Communication Systems and Networks
May 2019

The number of Unmanned Areal Vehicle (UAV) applications is growing tremendously. The most critical applications are operations in use cases like natural disasters and rescue activities. Many of these operations are performed on water scenarios. A standalone niche covering autonomous UAV operation is thus becoming increasingly important. One of the crucial parts of mentioned operations is a technology capable to land an autonomous UAV on a moving platform on top of a water surface. This approach could not be entirely possible without precise UAV positioning. However, conventional strategies that rely on satellite positioning may not always be reliable, due to the existence of accuracy errors given by surrounding environmental conditions, high interferences, or other factors, that could lead to the loss of the UAV. Therefore, the development of independent precise landing technology is essential.

The main objective of this thesis is to develop precise landing framework by applying indoor positioning techniques based on RF-anchors to autonomous outdoor UAV operations for cases when a lower accuracy error than the provided by Global Navigation Satellite System (GNSS) is required.

In order to analyze the landing technology, a simulation tool was developed. The developed positioning strategy is based on modifications of Gauss-Newton's method, which utilizes as an input parameter the number of anchors, the spacing between them, the initial UAV position, and the Friis-transmission formula to calculate the distance between the anchors and the UAV. As an output, a calculated position of the UAV with an accuracy in the range of tens of centimeters is reached.

The simulation campaign shows the dependencies of the effects of the anchor's number and corresponding spacing on positioning accuracy. Also, the simulation campaign shows Gauss-Newton's method parameter value that maximizes the system performance. The results prove that this approach can be applied in a real-life scenario due to achievements of both high accuracy achieved and close to perfect estimated landing trajectory.

Keywords: UAV, Positioning, Automatic Landing, Simulation

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

PREFACE

This thesis work is one of the essential parts of long-going research and development on automated UAV orientation and landing system, performed in collaboration with aColor project and W.I.N.T.E.R. group at the Faculty of Information Technology and Communication Sciences, Tampere University, Finland.

First and foremost, I would like to express my deep gratitude to my supervisor, Dr. Alexander Pyattaev, for his guidance, valuable ideas, fruitful critique, and constructive advice, without whom it would have been impossible to become the engineer that I am today. Under his guidance, I have gained a wealth of experience that will help me in the future.

I am particularly grateful to Dr. Aleksandr Ometov for his great support not only as a supervisor but a friend and his ability to shine a light on the complicated situations and make life better.

I would also like to acknowledge Prof. Evgeny Kucheryavy and Asst. Prof. Sergey Andreev, for supporting, encouraging and leading the group.

I wish to acknowledge my colleagues at YL-Verkot, especially Dmitry Solomitskiy, and Andrey Samuylov for supporting me and having a great time in the office. I would also give my appreciation to Dr. Mikhail Gerasimenko, Roman Kuvalchukov and Aleksei Ponomarenko-Timofeev for their support with field tests during the Finnish spring.

I would also express my gratitude to my new colleagues from the Nokia SoC Prototyping and Verification team for welcoming me so kindly, and for supporting and encouraging me during the very-needed coffee breaks.

I would like to express my deepest feelings to my family, for supporting me, for believing in me, for loving me from far away and making me feel that they are very close. And in a very special place of my heart, Anastasia, thank you for your endless energy to help me, for believing in me, and most importantly for holding on to me.

And finally I could not be more grateful to the family that I have chosen, my friends, with them my life in Finland has been easy, full of good moments and joy.

Tampere, 13th May 2019

Carlos Mariano Castillo Mateos

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

3GPP 3rd Generation Partnership Project

aColor Autonomous and Collaborative Offshore Robotics

GCS Ground Control Station

GLONASS Global Navigation Satellite System

GNSS Global Navigation Satellite System

GPS Global Positioning System

IEEE Institute of Electrical and Electronics Engineers

LOS Line-of-Sight

NLOS Non-Line-of-Sight

NS3 Network Simulator 3

RFID Radio Frequency Identification

SIR Signal to Interference Ratio

SNR Signal to Noise Ratio

TCP Transmission Control Protocol

TUNI Tampere Universities

UAS Autonomous Unmanned Aerial Vehicle Systems

UAV Unmanned Aerial Vehicle

UDP User Datagram Protocol

UE User Equipment

USV Autonomous Unmanned Surface Vessel

W.I.N.T.E.R. Wireless Intelligence for Networking Technology by Engineering and Research

WINTERsim W.I.N.T.E.R. Simulation tool

WLAN Wireless Local Area Network

GLOSSARY

5G is the 5th generation of communication technology designed by 3rd Generation Partnership Project (3GPP).

Euclideand-Distance is the distance between two points given by the Pythagorean formula.

Friis transmission formula is used in telecommunications to measure the ratio between the transmitted and the received power, given some parameters. These parameters are: the aperture of the transmitter and receiver antennas, the radio wavelength and the distance between the receiver and the transmitter.

Free-Space Path Loss This formula is derived from the Friis transmission. It expresses the signal loss between transmitter and receiver antennas, given the working frequency and the distance between them.

H2020 program started in 2014 in which Europe is investing almost €80 billion for Research and Investigation. It will last until 2020 [1].

LTE is the 4th generation of communication technology designed by 3GPP.

Multipath propagation phenomenon that results in a radio signal reaching the receiver through two or more paths.

Path-Loss Power attenuation of an electromagnetic wave.

Python is an interpreted, high-level, general-purpose programming language.

Wi-Fi Networking technology that uses radio-wave signals to provide Internet and networks connections. Designed by the Institute of Electrical and Electronics Engineers (IEEE).

1 INTRODUCTION

In this chapter, the purpose of the project will be analyzed, focused on the project motivation and goals. After the introduction, the structure of the thesis will be presented, to ease the reading and understanding of each chapter.

1.1 Motivation

Interestingly, the Unmanned Aerial Vehicle (UAV) operation is a topic that has been under careful research community attention for more than a decade [2, 3, 4]. While its use has been a spreader, more applications have been found for the UAVs in life-rescuing and natural disaster scenarios [5, 6]. These involve border surveillance to rescue people in the water [7, 8], where the performance of the UAVs has to be as perfect as possible [9].

However, new limitations appear with the new drone-based applications together with the need to overcome those challenges. Commonly, UAVs are controlled by the operator having direct sight to the UAV [10, 11, 12, 13]. In the case, the performance degradation may bring a mission failure when video transmission experience network delays, or if the UAV is not in the direct line of sight. In order to prevent it, modern UAVs are equipped with Global Navigation Satellite System (GNSS) receivers [14] that can estimate the position of the UAV and can trigger the UAV to "return home" mode in the emergency scenario, i.e., to return to the original position from which the UAV was launched or a preprogrammed location. However, these mostly used Global Positioning System (GPS) trackers still face an error of approximately $1-10$ meters [15]. GPS with less positioning error modules are also present on the market (or could be mitigated by utilizing multiple receivers [16]), but the price for this category of devices increases the value of the UAV.

Autonomous and Collaborative Offshore Robotics (aColor) [17, 18, 19] is a project started on 2017 in Tampere University of Technology. The aColor system is a collaborative project designed to perform missions and tasks. The core of this project is to achieve a shared intelligence between different offshore vehicles (vessels, UAVs, submarines, etc.), as well as the situational awareness of these subsystems. The ultimate aColor's goal is to build a system, whose main strengths are reliability and availability. The system will be independent and will use completely autonomous navigation and path planning.

Moreover, in the air scenario for the aColor system, a UAV has been deployed initially to perform some videos of the vessel's field development. This UAV will be used in the future for water surveillance, tracking disruptions in the water surface, and also as a communications relay if the distance between the vessel and the shore station is greater than the distance capability of a direct radio link. However, one essential key is missing, to be a fully autonomous system is needed a landing system for the UAV that can be performed on top of a moving surface without human aid.

1.2 Objective

In this regard, this master thesis aims to introduce a safe method to operate the UAV without any form of vision as part of the Autonomous Unmanned Aerial Vehicle Systems (UAS) paradigm.

This thesis goal is based on the desired outcome to transform the whole aColor concept into a fully autonomous system. The thesis is focused on fulfilling the missing landing solution that does not require a human controlling the UAV to perform a high accuracy landing in a moving platform. The correct performance of the UAV has to be independent of weather conditions, as very dense fog, rain or wind, which are especially common for next to the shore and maritime operation. These cases describe the drawback of having a ground operator, because of the lack of visual aid the operator cannot anticipate the behavior of the UAV and prevent a crash.

UAVs are almost fully autonomous, with the exception of lack in a generalized landing method, that would make UAVs truly wholly autonomous. To make this happen, it is proposed to relay on radio-frequency signals to perform the landing. These signals will be transmitted from antennas on the ground and received by the UAV that will process the signal strength and approximate the distance from the antennas it is located.

To achieve the concept of a fully autonomous UAV with no aid from a ground crew, a localization system is needed with very low error occurrence. The reduction of the location error is needed when the UAV has to land or recover something in a space where there is no margin of error, for example, land on top of a small surface or recover an object from the ground. At any other scenario, the Global Positioning System (GPS) tracking is enough to provide information about the location of the UAV.

An initial architecture concept of the problem is shown in Figure 1.1. This architecture is composed of four anchors forming a surface in which center the drone is supposed to land, implementing Wi-Fi technology operating at a frequency band of 2.4 GHz and an autonomous UAV.

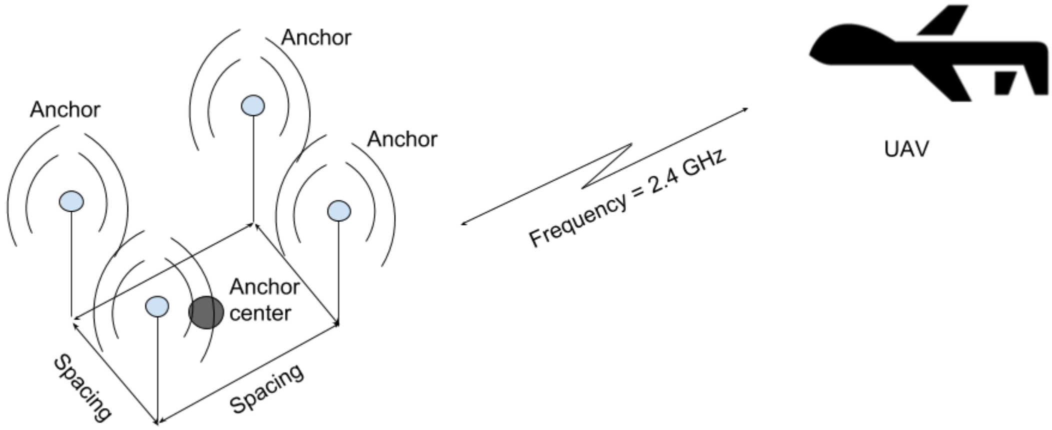


Figure 1.1. *Concept architecture.*

The main objectives of this master thesis are:

- To study a viable solution to perform an autonomous landing for a UAV.
- To study different mathematical approaches to locate objects.
- To design landing trajectories for different types of UAVs.
- To develop an overall system design.
- To create a simulation tool that will assist to develop the real-life system.
- To study different parameters that will improve the performance of the system based on the simulation tool.
- To perform a simulated UAV landing.
- To propose a solution with the calculated parameters that will maximize the system performance.

1.3 Structure of the Thesis

In this section, the structure of this thesis and a brief introduction to every chapter will be presented. This thesis is structured as follows:

- Chapter 1, Introduction. In this chapter, the motivation and the objective of the project will be presented. It presents the steps taken by the author to develop a solution and obtain optimized results.
- Chapter 2, Theoretical Background. The theoretical background is presented in order to explain the essential ideas of the thesis. The following concepts will be introduced:
  - UAV's types and design options.
  - The behavior of radio waves propagation.

  - Antenna diversity techniques to mitigate the attenuation in the radio communication.
  - Different already-implemented landing types.
  - Positioning, different types of positioning technologies and their performance.
  - Gauss-Newton algorithm as a mathematical tool utilized to locate an object with minimal error.

- Chapter 3, System Design. In this chapter, the scenario of interest is introduced. It explains how to land a drone autonomously on a moving surface. Next, a solution to locate and control the UAV based on Wi-Fi technology operating in 2.4 GHz band is proposed due to its broad market adoption [20]. Finally, the approach to developing the simulation tool and the physical restrictions of the system are given.
- Chapter 4, Simulation Development. First, it is shown the procedure of simulation tool development, as well as code implementation. Next, the algorithms used to calculate the positioning and the landing trajectory are presented.
- Chapter 5, Results and Analysis. In this chapter, the results obtained with the simulation tool, and the optimal values of the parameters to maximize the performance of the system are explained. The chapter is divided into two parts. In the first one, the performance results of locating the UAV and the values of parameters needed to improve these results are given. In the second one, the results of the simulation for the landing of two different UAV with a fixed-wing and a multi-rotor are given.
- Chapter 6, Conclusions and Future Work. In this chapter, the conclusion of the thesis is drawn after evaluating the overall result. Besides, it provides the lines for future work.
- Appendix. All implemented code for development the simulation tool is listed in the appendix. The code could be used to improve already existent tools. Some modules from the W.I.N.T.E.R. Simulation tool (WINTERsim) are needed for its correct function.

2 THEORETICAL BACKGROUND

This chapter depicts the required background needed to understand the proposed solution, and main reasons why it has been chosen to develop the technology in such a manner.

2.1 Introduction and Classification of Unmanned Aerial Vehicles

After the introduction of the project aColor on which this thesis is based on, one of the main devices, UAV, is described along with its types and the mission they are developed to perform.

A simple definition of a UAV is an unmanned aerial vehicle that can be controlled by the ground crew through a radio link [21]. The UAV set is composed by a Ground Control Station (GCS) that contains the control operators; a communication system that sends the control commands to the UAV, which in turn returns the payload and other data to GCS; support equipment (consisting of telemetry equipment, video and audio systems), and object carriers.

UAVs are commonly used in the civil and military markets. The utilization in the civil scenarios implies aerial photography, agriculture, environment protection, maintenance of high voltage cabling, etc. [22, 23, 24, 25]. In the militarily field, UAVs are used to track enemy fleet, retransmission of radio signals, area recognition, minefield location, radar jamming, and many other options. [26, 27, 28]

The element that determines the size of the UAV is the payload size. Usually, a UAV carrying a camera to stream video and a normal antenna to receive control commands can be of few kilograms and with a wingspan less than one meter. For a propeller UAVs, the separation between the propellers are about one meter, but the main determining factor is the lifting power [29].

Every UAV is designed for a particular mission. A deeper understanding of the most common activities that UAV perform are [30]:

- *Environmental applications*. Can be used in monitoring environmental disaster like chemical or nuclear pollution, where the health of human beings performing these actions might be harmed [31, 32].

- *Risk assessment*. Used in recognizance missions of a well-defended enemy field. Due to the smaller size, the devices might be undetected and harder to counter-attack. Those can take more risky decisions without putting human lives in danger. In civil scenarios, those can be used to monitor forest activities and to prevent the fire spread [33, 34].
- *Research purposes*. UAVs are employed in the aerospace field to replicate human-crewed aerial vehicles and to perform simulations and analysis in real conditions [35].
- *Environmental impact*. It is a common thought that UAV might cause a small impact on the environment due to its smaller mass. However, to have a lighter body does not necessarily mean that petrol consumption, gas and noise pollution will be reduced if the UAV is utilized as delivery transport. It has to be taken into consideration that UAVs cannot carry the same package amount and weight than a truck. The articles [35, 36, 37] present different environmental impact depending on the UAV type and mission.
- *Economical applications*. UAVs are smaller and usually cheaper than manned planes and can execute almost all the previously listed missions. Moreover, since UAVs need less number of human operators, the cost to deploy the UAV is cheaper.

This thesis focuses on the environmental control and life rescue usages of a UAVs [21]. There are multiple publications related to the use of drones to assist during natural disasters [6, 38, 39], scenarios modeling of prevention and facing of these situations [7], also covering how UAVs can help to improve the communication when the network is overloaded or unavailable [5, 40, 41] and for other advanced cases [42, 43, 44]. Works [45, 46] shown that the mmWave can improve the Signal to Interference Ratio (SIR) when the average altitude of the UAV, acting as a relay, is 40 meters above the User Equipment (UE) by a 3D-modeling approach.

Another straightforward differentiation relies on UAV design. It can be (i) a multi-rotor that will take-off and land vertically with the power of rotors, or (ii) a fixed-wing that will take-off and land diagonally and it is also capable of sky glide. In this thesis, both UAV types will be taken into consideration for the different types of landing procedures. These two UAV types will be acquired for the project aColor.

***(a)** Multi-copter, reproduced from [47].*

***(b)** Fixed-wing, reproduced from [48].*

***Figure 2.1.** Main UAVs used in aColor project.*

2.2 Attenuation in Radio Wave Propagation

The design of the control system for autonomous vehicles, as Autonomous Unmanned Surface Vessel (USV), UAV, is by default a communication channel between a ground or shore station and the vehicle [49]. This communication channel employs the use of radio link from a transmitter in the ground station and a receiver in the vehicle, allowing sending control commands to the vehicle in case. But during this control communication, it may happen that the connection gets disrupted preventing the receiver to capture the control command. These disruptions might be given in the radio channel.

The radio wave propagation in a communication link can be attenuated because of many reasons [50]:

- **Distance**. The separation between the transmitter and receiver must be in the range that the sensitivity of the receiver antenna can capture the signal and process it.
- **Frequency**. The higher frequencies are less likely to overpass an obstacle in such a way the signal strength does not get affected or absorbed by the object.
- **Antenna height**. It affects directly to the radio-wave as it influences the radiation angles. Also, in order to have a good radio-link, it is preferred to have a Line-of-Sight (LOS) between the transmitter and the receiver antenna.
- **Atmospheric conditions**. The weather conditions, such as rain or fog, may have a negative effect on the radio wave deteriorating the link.

Environmental effects on the radio signals have been broadly studied, for example, the radio wave attenuation produced by vegetation in wireless sensor networks [51], water vapor and snow [52], and forest environments [53]. Document [54] represents two figures with a significant effect of rain over one hour period on the $1 - 5$ GHz link.

Satellite to ground communication is one of the types that most suffer from link mitigation [55]. Those are affected from Free-Space Path Loss, where the distance between the transmitter and receiver is one of the principal parameters together with the frequency; curvature of the Earth, when the angle of the radio link is very low; and satellite-to-ground communications are also affected due to the ionosphere and atmospheric conditions.

Another type of signal mitigation is present in a short-range communication type. As the density of devices using radio waves increases, the physical medium starts to run short in resources. The USV will carry and use multiple devices and technologies that will increase the density in the physical medium. A real-life example of this phenomenon is taken from the wearable devices concept. Work [56] presents an investigation towards the efficiency of wearable connectivity, studying different technologies, short-range communications, and spatial reuse.

In this proposed scenario of this thesis, if the UAV attempting to land in a moving platform on top of the water surface, is being controlled through a radio link communication, the effects explained in this section can cause significant damage on the UAV. This is why it is needed to find an approach to mitigate these effects and minimize the use of control commands to land the UAV.

2.3 Antenna Diversity Review

One of the most common disruptive effects in radio communications is interference. When a system is composed of multiple transmitters and receivers the interference between those separated signals is very high. Interference provokes a very low Signal to Noise Ratio (SNR) that influences directly in the quality of the communication in a negative way. One valid approach is to increase the transmitted power affecting directly the spectrum efficiency, however, it will not solve the problem if the rest of the transmitters will also increase their transmission power.

Antenna diversity is a widely studied physical solution that helps to improve the SNR of a transmitted radio signal, and therefore improve the quality and reliability of communications [57, 58]. Following [59], orthogonality can be achieved using multiple identical antennas but separated with distance. Work [57] states that the spacing between monopoles does not need to be very big, at a close to GHz frequency, the spacing between the monopoles can be of 0.05 meters, and the correlation coefficient is acceptable. If more monopoles are added to the system, then the spacing between them has to be increased. One of the drawbacks is to have every branch separated, requiring a greater space to deploy the antenna, and it is not useful for low frequencies. However, for higher frequencies, it is possible the use of omnidirectional antennas. There can be multiple channel diversity if the separation between the antennas is sufficient. With the usage of two monopoles, it is needed a 0.15 λ, where λ is the signal wavelength in meters, between them, and if the number of monopoles increases, to improve the channel diversity, then the separation has to increase also (for 3 monopoles is needed a separation of $0.25\ \lambda$).

In work [60], it has been found that the interference between different signals is mitigated due to spacing. The combination between M antennas and N interferes provides the same average error as $M - N + 1$ antennas and no interferes. This means that the error rate of a user is not affected by the other users even if the number of users is the same as the number of antennas.

Spatial diversity is achieved by having a separation between the antenna branches. There are two basic ways to separate the branches, horizontally and vertically, depending on the scenario to be deployed. Having antennas separated farther than half a wavelength effectively de-correlate signals contributing to improve the noise reduction. A communication link with low noise allows having an automated landing system with few losses capable to use radio-frequency signals to locate accurately a UAV.

2.4 Study of Landing Systems

The study of the behavior of the radio link used for the control communication between the ground station and the UAV is followed by the study of different autonomous landing approaches.

In conventional aircraft, the landing is the most critical operation since passengers might get injured if the landing is not performed correctly. Today passenger aircraft is not yet unmanned, and a crew is controlling the flight on board and from the ground station. Over the years, significant improvements have been developed in this matter to aid the pilots and the ground crew, creating an approach to a more autonomous landing system [61].

In UAVs, the autonomous landing capability has been studied and tested in more details, UAV presents an advantage due to the lack of direct risks. These techniques usually employ GPS positioning and own UAV's autopilot to land on a steady base station [62]. However, in this approach, positioning error brought by the GPS might be expected.

In order to mitigate the landing accuracy given by GPS, a visual analysis technique is commonly applied [63, 64, 65]. This hybrid landing system consists of taking the UAV to specific GPS coordinates and, then, the UAV will recognize the visual pattern given in the landing point using its cameras and signal processing.

However, all these previous solutions are not scalable to moving platforms. In this thesis scenario, the landing points will have continually changing GPS coordinates compared to ones provided at the beginning of the mission. To overcome this situation, the UAV requires a stable communication link to the landing platform to receive the new coordinates, or another technique has to be employed. Visual aid provided by some given pattern (e.g., light combination or QR-codes) in the landing platform can continue to be used to minimize the location error in the surface target position.

In the scenario of interest, the moving platform is on the water surface, thus creating a risk of damaging or losing the UAV. In order to land a UAV on a moving vessel, the use it's equipment to capture and retrieve should be applied. Document [66] provides a design to approach this mechanical solution.

2.5 Positioning Technologies Overview

All the different approaches to autonomous UAV landing methods assume that the locations of the UAV and landing platform are known. The default localization system commonly used by many devices is the Global Navigation Satellite System (GNSS) [67, 68, 69]. Due to these technologies, it is possible to locate any device that has a receiver in any part of the planet having LOS to the satellites. European Union has invested in its own positioning technology, Galileo, more than €10 billion and it is estimated that the total investment will be of €22 billion [70] by the end of the project. Also, within the framework of Galileo project, the European Union is investing in H2020 a budget of €100 million for R&D funding [71]. Apart from Galileo in Europe, other countries and organizations have their own satellite positioning technologies, the most important are: United States with GPS, one of the most widely used in the world, and Russia with its own positioning system called Global Navigation Satellite System (GLONASS) [72]. These three satellite positioning technologies, Galileo, GPS and GLONASS are not the only ones deployed but are the most broadly adopted ones.

Because of these positioning systems, it is possible to locate any device with very high accuracy. In the GPS case, as stated by [15], it is possible to have an accuracy of 4.9 meters radius in LOS using a GPS-enabled smartphone. This accuracy gets worse if Multipath is present due to buildings, trees, bridges, etc., surrounding the receiver.

Satellite-based localization requires the presence of four or more satellites. These satellites follow an orbit that it might imply that it is possible to have a "dead-zone" for some localization technologies. If the desired positioning error is in the magnitude of a few centimeters, then, satellite positioning technologies are not the best choice. Even though these technologies can reduce that error, the expense of a receiver increments the cost of the project greatly. For this purpose, it is necessary to find a technique that provides a better outdoors accuracy, and the cost is not very significant.

In indoor scenarios, high accuracy is more desired than outdoors because an error of few meters might mean that the object is in another floor. Because of the fact that satellite signals cannot be used inside the buildings, therefore, other techniques were developing over the past years. The most common approach for indoor positioning is the use of low range radio signals [73]. A transmitter acts as an anchor and the device that is meant to be located will measure the received signal estimating the distance to the anchor. If the indoor plan is known, the position estimation has an error of 0.5 meters measured in [74]. Authors of [75, 76] have studied the possibility and related challenges to use not only Radio Frequency Identification (RFID) signals but Wireless Local Area Network (WLAN) signals for indoor localization. One of the main advantages of using WLAN is the presence of high dense network deployments, reducing the cost of a positioning system greatly.

2.6 Localization Based on Modified Gauss-Newton's Method

In this thesis, the indoor positioning-based method is analyzed as an optimal solution for outdoor positioning. In the proposed scenario, the idea implies that the layout is not known beforehand and could dynamically be changed due to the mobility of the vessel and the effects of waves. Therefore, a method to calculate the position of an object in an unknown and potentially moving space is the problem statement. In this thesis, it is proposed a modification of the Gauss-Newton method for non-linear models. This method is iterative, meaning, the calculated intersection will be a better approximation to the root of the function than the original guess. This is why it is not known beforehand how many iterations are required to find the root of the function.

The calculation of the position of the UAV can be accomplished using the knowledge of Friis transmission distances (applying Free-Space Path Loss) from the anchor point to the UAV location. The position coordinates are non-linear, creating an extra difficulty to the UAV location calculation. One of the most studied methods to calculate non-linear regression by least squares is the use of the Gauss-Newton algorithm. This solution allows solving non-linear least square problems [77]. Applying some modifications to the algorithm, it can be adjusted to multiple fields as well [78, 79, 80, 81]. Based on the previously listed works and [82], the following mathematical development modifying Gauss-Newton's method is shown to prove that the system is robust.

The goal is to allow modeling a system by a non-linear function $y = f(x, a_1, a_2, \dots)$ composed by a set of parameters $a = [a_1, a_2, \dots]^T$ able to minimize the residual error between the actual location and the calculated position of the UAV:

$$\epsilon(a) = \sum_{i=1}^{N} r_i^2 = \sum_{i=1}^{N} [y_i - f(x_i, a)]^2 = \sum_{i=1}^{N} [y_i - f_i(a)]^2 \rightarrow min, \tag{2.1}$$

where a is the set of parameters that will define the system, $\epsilon(a)$ is the total residual error depending on the parameter set a, r_i is the residual error in every iteration, N is the number of data points $(x_i, y_i), (i = 1, ..., N)$, y_i is the real location of the UAV, and $f(x_i, a), (i = 1, ..., N)$ is the location calculated in every iteration. It has been defined $f_i(a)$ as $f(x_i, a)$.

The residual error is given by the cumulative sum of every iteration of the system. The minimization of this residual error is the ultimate goal. The residual error expressed in vector form to represent the general approach is as follows:

$$r = \begin{pmatrix} r_1 \\ \dots \\ r_N \end{pmatrix} = \begin{pmatrix} y_1 - f_1(a) \\ \dots \\ y_N - f_N(a) \end{pmatrix} = \begin{pmatrix} y_1 \\ \dots \\ y_N \end{pmatrix} - \begin{pmatrix} f_1(a) \\ \dots \\ f_N(a) \end{pmatrix} = y - f(a), \tag{2.2}$$

where $y = [y_1, ..., y_N]^T$, $f(a) = [f_1(a), ..., f_N(a)]^T$.

Therefore, the general equation in vector form is:

$$\epsilon(a) = \sum_{i=1}^{N} r_i^2 = r^T r = ||r^2|| = ||y - f(a)^2||, \tag{2.3}$$

where the total residual error $\epsilon(a)$ is the absolute value obtained in the difference between the real and the calculated location of the UAV.

Once it is obtained the method to find the residual error, it is necessary to find a that minimizes $\epsilon(a)$. In order to do so, the equation where the gradient of the vector is equal to zero is used.

$$\frac{\delta}{\delta a_j}\epsilon(a) = \frac{\delta}{\delta a_j}\sum_{i=1}^{N}[y_i - f_i(a)]^2 = -2\sum_{i=1}^{N}[y_i - f_i(a)]\frac{\delta f_i(a)}{\delta a_j} = -2\sum_{i=1}^{N}[y_i - f_i(a)]J_{ij} = 0, \quad (j = 1, \dots, M), \tag{2.4}$$

where J is the Jacobian matrix defined as:

$$J_{ij} = \frac{\delta f_i(a)}{\delta a_j}, (i = 1, \dots, N, j = 1, \dots, M), \tag{2.5}$$

where the Jacobian is the first derivative with respect to the paramaters wanted to be minimized (a in this case), $i = 1, \dots, N$ defines the number of iterations, and $j = 1, \dots, M$ defines the number of parameters a.

However, it might happen that equation (2.5) do not have a closed solution for a. Finding the optimal parameters $a = [a_1, \dots, a_M]$ that will minimize the residual error $\epsilon(a)$ when the previous equations do not have a solution, can be done with the use of iteration:

$$a_{n+1} = a_n + \Delta a, \tag{2.6}$$

where it is required to find $\Delta a = a_{n+1} - a_n = [\Delta a_1, \dots, \Delta a_M]^T$. If Taylor expansion is considered $f_i(a_{n+1})$ at a_n, then the following equation is found (in vector form):

$$f(a_{n+1}) \approx (a_n) + J\Delta a. \tag{2.7}$$

After substituting (2.7) in (2.4):

$$\sum_{i=1}^{N} J_{ij} \sum_{k=1}^{M} J_{ik}\Delta a_k = \sum_{i=1}^{N} J_{ij}[y_i - f_i(a_n)], (j = 1, \dots, M). \tag{2.8}$$

Adapting equation (2.8) to matrix form and solving for Δa:

$$\begin{gathered} \Delta a = a_{n+1} - a_n = (J^T J)^{-1} J^T (y - f(a_n)) = J^-(y - f(a_n)), \\ a_{n+1} = a_n + \Delta a = a_n - J^-(f(a_n) - y), \end{gathered} \tag{2.9}$$

where $J^- = (J^T J)^{-1} J^T$ is the pseudoinverse, and is obtained the iteration:

$$a_{n+1} = a_n + \Delta a = a_n - J^-(f(a_n) - y). \tag{2.10}$$

Finally, it is possible to see that the solution to the modeling problem is (by Newton's method for solving the multivariate non-linear equations):

$$f'(a) = f(a) - y = 0. \tag{2.11}$$

The situations may appear when the system may not converge at any iteration, this is why a parameter $0 < \gamma < 1$ is introduced to reduce the step size of the iteration. This parameter γ has to be calculated in order to maximize the performance of the system. This performance measures if the system has found the solution, and how long it has taken to find the solution.

The following would be further used to estimate the UAV position:

$$a_{n+1} = a_n + \gamma \Delta a, \tag{2.12}$$

where a_n in the first iteration is the initial guessed position, γ is the step parameter to smooth the step Δa obtaining a value a_{n+1} closer to the actual position of the UAV. The residual error is minimized at every iteration,

As it has been widely studied, “the [Gauss-Newton] method is attractive because it converges rapidly from any sufficiently good initial guess” [83]. In this thesis, the initial guess position is analyzed and provided an optimal solution as well as the parameter γ.

	Notation	**Description**
Input parameters	a	Set of data parameters. Initial guess position, anchor position and measured distance.
	γ	Step size to get a softer approach towards the desired output.
	$f_i(a)$	Positioning calculation in each iteration.
	y_i	Real position of the UAV
Output parameters	r_i	Residual error between the real and calculated position depending on the iteration.
	$\epsilon(a)$	Sum of every residual error.
	$a_n + 1$	Optimal parameter that minimizes the error between the real and the calculated UAV coordinates.

Table 2.1. *Notation used in the mathematical development.*

3 SYSTEM DESIGN

After Chapter 2, all the necessary knowledge required for the system design is provided: the definition of the problem to be solved, design of an accurate autonomous landing system on a moving platform, location of a UAV, using a new algorithm based on Gauss-Newton's method, and different landing technologies. In this chapter, it is proposed a procedure to create a system that allows a UAV to land on a moving platform.

3.1 Target Scenario

The scenario is described as follows. An autonomous vessel is deployed in the sea or in a lake to perform missions that may involve the integrity of underwater mines checking or rescue of the boat crew that is overboard due to bad weather conditions, or boat malfunction, or collision, etc.

The autonomous vessel is communicating with the ground station through a radio link that can provide control commands in case of a malfunction of the system, otherwise, the vessel is operating autonomously. This radio link can also be used for telemetry, sending all the data generated by the sensors deployed in the vessel, or streaming the data from cameras. The radio link can be a LOS backhaul from the vessel to the ground station, or if the shore is very far away, the radio link could be set through satellite communication. Another possibility to connect the vessel to the ground station is through a backbone link via the UAV, in order to achieve that, a corresponding design have been developed previously by our group and published in [84]. Simply, a UAV can be deployed from the vessel to do a visual recognition to avoid potential harm to the crew, prevent collisions, or even be able to command the route modification of the vessel. The drone will be deployed from the vessel surface, perform its mission, and return to the vessel. As the vessel is on top of the water surface, the "return home" feature that most of the UAV have in their system is not acceptable as the vessel might have changed its coordinates.

3.2 Proposed Solution

The solution proposed in this thesis is based on object localization using radio signals. A group of anchor-antennas will transmit radio signals that will be received by the UAV, which will calculate the Friis transmission distance to those antennas based on the antennas' transmitted power and the power it has received (among other parameters). To calculate the object position, the following parameters will be used:

- The Friis distance, calculated with the Euclidean distance from the anchors to the UAV coordinates, and the Free-Space Path Loss;
- A guessed initial position used to calculate a virtual distance;
- Error tolerance. If the error between the virtual distance and the actual distance is less than a given value, then the UAV position has been correctly calculated, otherwise, the UAV guessed coordinate would be modified.

After the UAV coordinates are calculated it is necessary to design a trajectory to the landing platform. This trajectory will describe a parabola from the UAV located point to the center of the anchors. The anchors will be situated on the front of the USV, together will form the landing platform. The anchors will picture the shape of the surface, in which center the UAV will land. Moreover, as two different types of UAV are going to be used in aColor, a fixed-wing, and a multirotor UAV, two different landing approaches are needed. In the fixed-wing landing case, the parabola is followed until the landing point, employing a trap to prevent the UAV to crash or fall-off the vessel. For the landing trajectory of the multirotor UAV, a slightly different approach is used, they perform their landing in a vertical manner, this means that the UAV will be located perpendicularly on top of the desired landing spot before starting the descent.

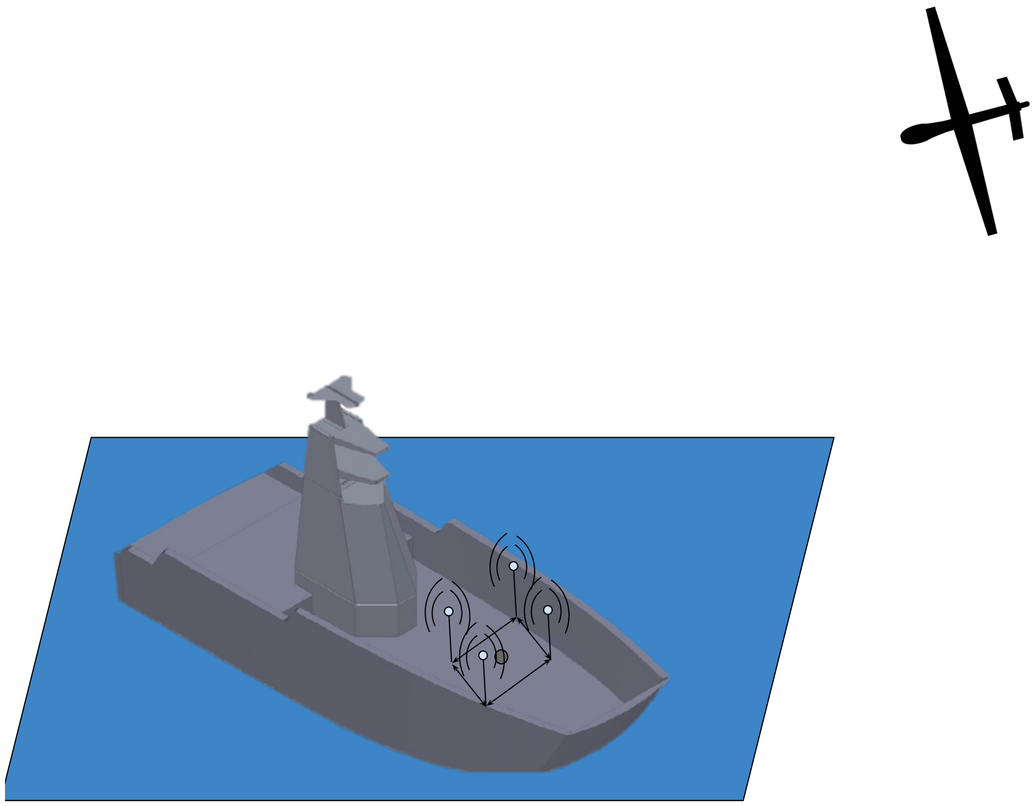


Figure 3.1. *Proposed solution.*

3.3 System Simulation and Physical Restrictions

The first step of the landing system development is to create a simulation tool. This simulation tool will recreate the scenario and will perform the initial analysis of the solution. The simulation has to be as precise as the real case scenario, including propagation errors and randomness in the initial location of the UAV.

This simulation tool is built with Python language [85]. Python is a widely used programming language due to its properties:

- Very robust and powerful.
- It can be deployed in any operating system.
- User-friendly.
- Open source.

Two libraries have been used to develop the simulation tool in Python:

- *numpy* [86] is a built-in library designed for scientific usage. It is a powerful N-dimensional array object, useful for linear algebra, Fourier transforms and randomized calculations.
- *WINTERsim* is a custom library developed by the Tampere Universities (TUNI) research group Wireless Intelligence for Networking Technology by Engineering and Research (W.I.N.T.E.R.) [87].

WINTERsim can be compared to other commercial simulation tools like Network Simulator 3 (NS3) [88]. The main difference is that WINTERsim is a modular platform designed for simulating wireless systems in a more simple manner. WINTERsim allows to use already made wireless technology modules, like Wi-Fi [89], LTE [90], etc., and Internet protocols Transmission Control Protocol (TCP), User Datagram Protocol (UDP), etc., to create fast and straightforward network simulation. Also, it allows the developers to create their own protocols or add custom technologies based on the specifications, like 5G [91].

In the simulation campaign, *numpy* allows to create and perform vector operations in a simple way, as well as, randomize the system. The numpy has already built-in distributions, like Gaussian, used for the calculation of the antenna transmission error. *WINTERsim* has provided the methods to calculate free-space-path-loss and Friis-distances, as well as all the built specifications for Wi-Fi.

During the system developing, all the restrictions pointed by the aColor team have to be taken into account when developing the simulation tool and implementing the hardware as well. These restrictions will define how the hardware will be implemented, as the space on the USV is limited. The landing platform formed by the anchors picturing the surface shape is a constraint to the vessel dimensions. Due to this limited space, it is needed to foresee a scenario where it might happen the loss or crash of the UAV. To prevent this scenario, a capture system will be deployed, the UAV will be captured and secured to the vessel to avoid damages. All these safety measures will be implemented by the aColor team with collaboration from the W.I.N.T.E.R. research group.

Another physical limitation is the height of the vessel's mast. The mast is approximately two meters tall and can produce blockage to the LOS, affecting the radio-link between the anchors and the UAV. This thesis proposes to use omnidirectional antennas in order to locate the UAV, with the disadvantage of shorter link range, because of directional antennas will be affected on a greater scale if the radio-link between UAV and the anchors is in Non-Line-of-Sight (NLOS).

The mast height has to be taken into consideration when performing the landing of the different UAVs. For the fixed-wing UAV, the landing trajectory has to be done aiming the front part of the vessel to avoid a collision to the mast. This can be done either orienting the vessel to face the UAV or sending a control command to the UAV to modify its position to start the landing. In the multi-rotor UAV case, it is needed to prevent the UAV to fly in a height lower than two meters before reaching the landing point. Once the obstacle has been avoided, and the UAV is positioned on the landing point at two meters height, the UAV can begin the descent.

4 SIMULATION DEVELOPMENT

In this chapter, the development of the simulation tool following the directives from Chapter 3 is presented. This simulation tool takes into consideration the restrictions with respect to the antenna spacing and elaborates the modified Gauss-Newton's algorithm to calculate the UAV coordinates.

In order to estimate the location fo the UAV the following elements are needed:

- Antennas that are acting as anchor points;
- The distance between anchors and UAV;
- Gauss-Newton's method estimation;
- Residual of the actual target location and estimated location.

4.1 Anchors, Positioning and Center Calculation

In the scenario, the anchors are antennas forming a surface in which center the UAV will land. These anchors will be transmitting at a frequency of 2.4 GHz. The anchor-antennas are isotropic radiating in every direction as initially the location of the UAV is unknown.

The antennas acting as the anchor will form a flat surface. The center will be equidistant to every antenna, and the antennas will have the same distance between each other for the simplest case. The first intuitive idea was to utilize only four antennas but when the simulations were executed – it was decided to analyze the performance of the system using different sets of antennas. The decision on the number of anchors generally depends on the error performance given by the simulation output. The anchor is positioned around $[x, y] = [0, 0]$ depending on the number of antennas.

The center of the anchor will be calculated and given as a parameter to the landing algorithm as it will be one of the points of the parabola.

- *Three anchors:* The anchors form a triangle. To calculate the center, for every axis, it is needed the sum of every antenna coordinate corresponding to that axis and divided by 3.
- *Four anchors:* The anchors form a rectangle. The middle point of the diagonal formed by the anchors will act as the center of the landing surface.

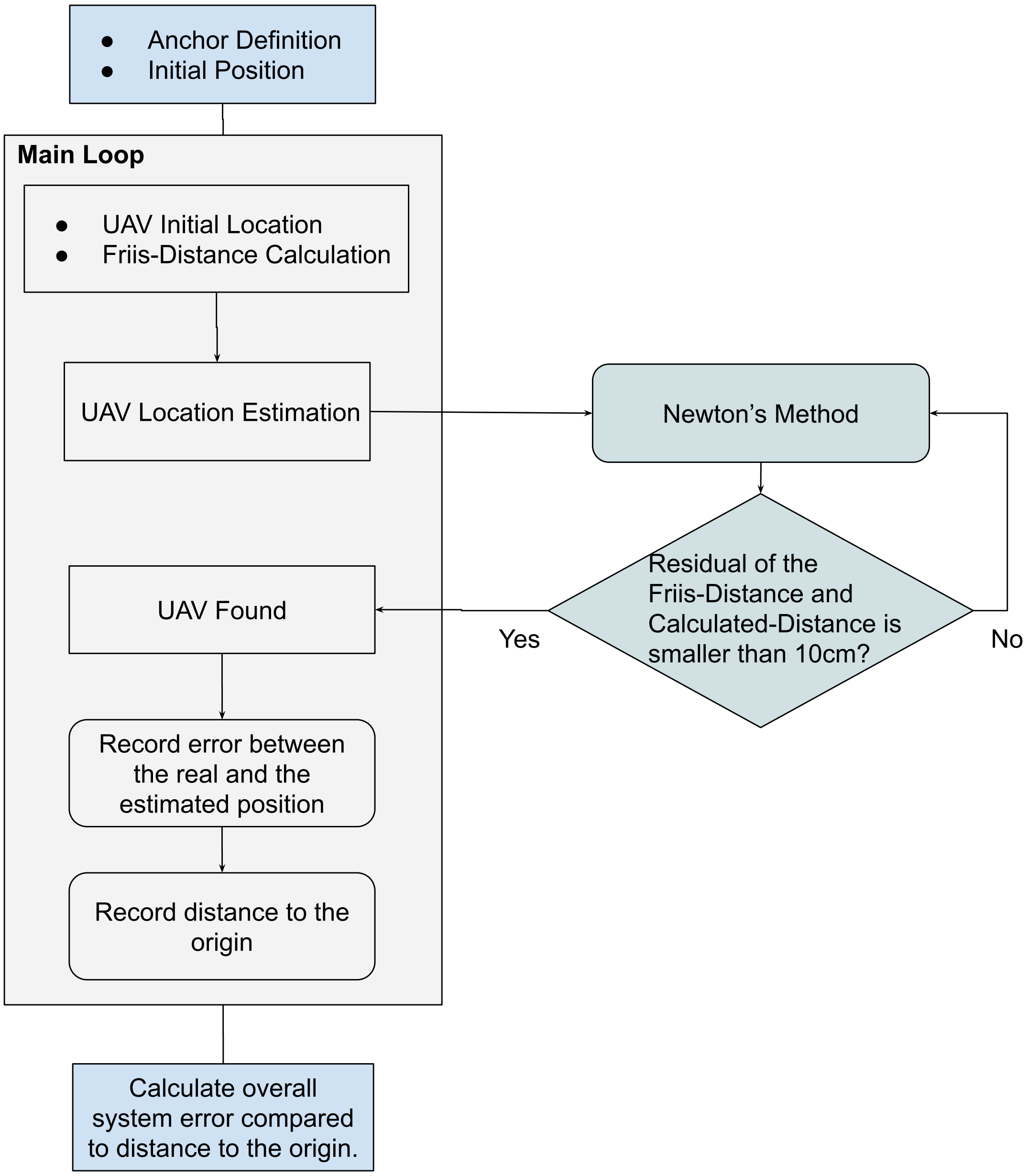


Figure 4.1. *System flow chart.*

4.2 Drone-Anchors Distance Calculation

In order to know the distance between every antenna that forms the anchor and the UAV, Friis distance formula was used. First, the Euclideand-Distance was calculated between every antenna and the target. Then, Free-Space Path Loss was obtained using as parameters the calculated distance and a frequency of 2.4 GHz (this frequency was chosen because the purchase of Wi-Fi antennas operating at this frequency is inexpensive). From the antenna specification, it is known that it has an error that follows a normal distribution of 1dB, this error is also added to the path losses. Once the path loss is known, it is proceeded to calculate the Friis distance. In order to calculate the Friis transmission distance, a combination of Python and WINTERsim libraries was used.

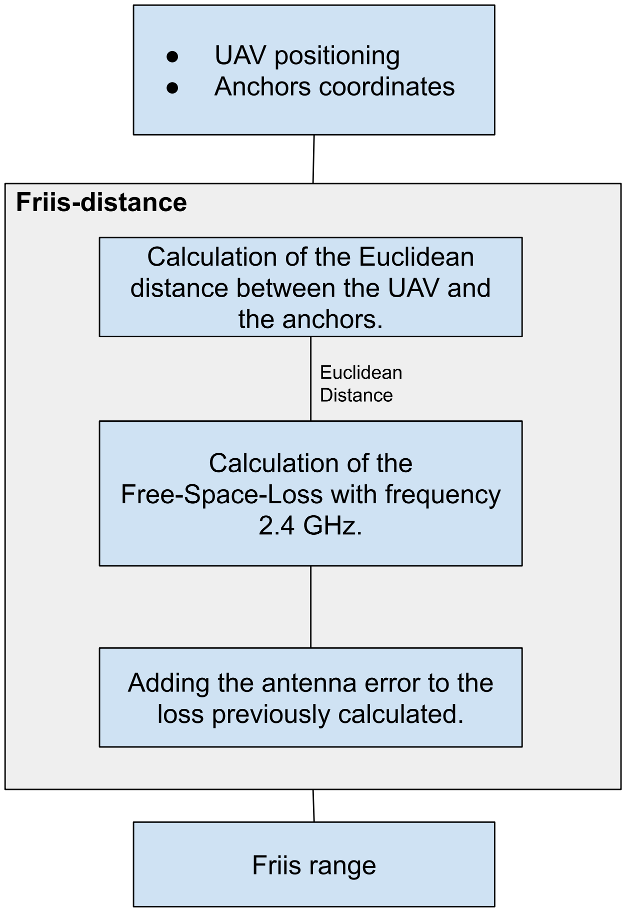


Figure 4.2. *Flow to calculate the Friis range.*

Figure 4.2 represents the steps to be taken to calculate the Friis distance between the anchor's coordinates and the UAV position. First, it is required to find the Euclidean distance between the anchors and the UAV to calculate the Free-Space-Loss. Second, the FSPL output will be the input to the Friis-range equation.

The next step is to randomize the initial location of the UAV. The UAV will appear in a volume of 1000 cubic meters (10 meters in each of the three coordinates). A vector of 5000 different UAV positions will be created in order to be able to check the performance of the system.

Once the actual position of the UAV is obtained, as well as the knowledge on how to calculate the Friis transmission distance to each antenna, the values will be submitted to customized Gauss-Newton's algorithm in order to estimate the UAV location.

4.3 Localization Based on Gauss-Newton's Method

This algorithm is a variation of Gauss-Newton's method explained in Section 2.6. It has been implemented manually to suit our purposes, no external library with this algorithm has been used.

To calculate by means of this modified algorithm, the initial coordinate guess should be provided, i.e., the coordinates of the anchor-antennas are situated and the original friss distance between the UAV actual position and the anchor.

As it can be seen later in Chapter 6, the parameters, error tolerance and γ, need to be optimized to improve the performance of the system. The Gauss-Newton parameter γ will define if the system converges. It allows specifying the step to take if the coordinate has not yet found on every iteration. The error tolerance is used as the margin error between the calculated position using Newton's method and the measured position between the anchors and the UAV. If the error tolerance is not fulfilled, then a variation in the guessed position of the UAV will be applied depending on the direction of the vector connecting the anchor and the UAV.

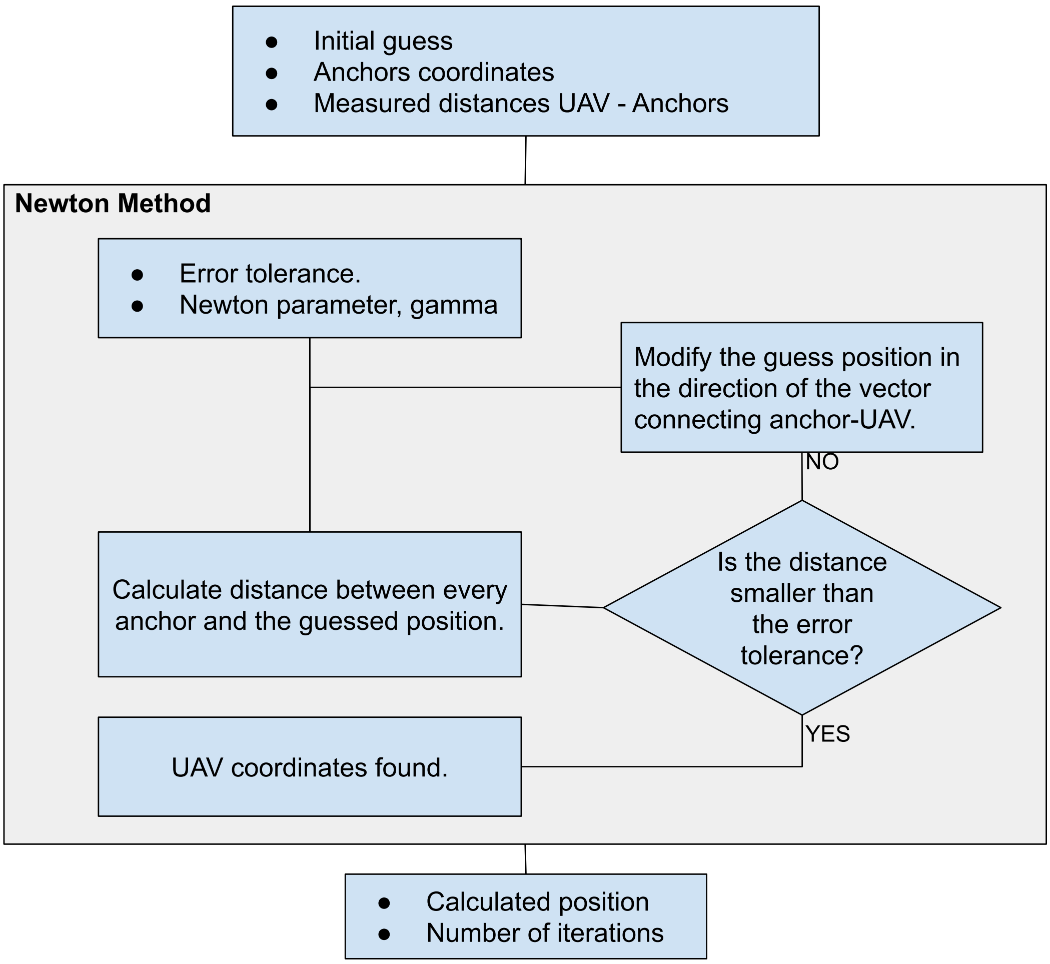


Figure 4.3. *Flow to calculate the Gauss-Newton's method.*

Figure 4.3 represents Gauss-Newton's method algorithm to calculate the UAV coordinates, providing the initial position guess, the Friis-distance from every anchor to the real UAV position, and the anchors' coordinates.

After the allowed error of the calculated and real distances is met, the new position is returned as well as the number of iterations required to find it.

4.4 Unmanned Aerial Vehicle Landing System Development

Once the UAV has been found, the landing procedure should be executed. The landing surface might be moving, which makes the requirement of fast calculation of a trajectory from the current position to the landing point a must. To perform a soft landing, an algorithm should be able to be executed fast enough to estimate the new position taking into account bad weather conditions.

The best solution that fulfills all the criteria is to have a parabolic trajectory between the UAV and the anchor-antennas. In this approach, the center of the anchors will act as one of the points of the parabola and the calculated position of the UAV will act as the vertex of the parabola.

It will be used the parabolic function $y = a(x-h)^2 + k$, where a is to be found, $[h, k]$ is the vertex coordinates, in this case the UAV coordinates, and $[y, x]$ are the coordinates of the center of the anchors.

First, the parameter a needs to be found:

$$a = \frac{y-k}{(x-h)^2}. \tag{4.1}$$

Next, we proceed with generating the parabolic second-degree equation substituting the values a, h and k, and solving by x:

$$y = tx^2 + ux + v, \tag{4.2}$$

where t, u and v are values obtained in the substitution. The z-position of the UAV is used due to the need to trace the parabola in the z-axis, and the values x and y will be linear.

The coordinates are stored as an array where the first position of the coordinate is at 0. With this in mind, anchor[2] corresponds to the z-coordinate of the anchor.

While the UAV is in the air, its altitude is being monitored. In case the next z-position of the UAV is lower than 1 meter, the UAV will maintain the altitude of 1 meter until reaching the landing point. Once the UAV has reached the landing coordinates, it will descend vertically to the center of the anchors.

The developed system, after the first calculation of the UAV position, the UAV has the memory of the calculated locations, provoking a faster convergence of the system in the next taken coordinates.

If the processing power of the UAV is not enough to perform the positioning calculation before the UAV changes its coordinates, then the system can be modified to have fewer iterations to fulfill the positioning error tolerance. This will provoke a higher error in the initial iterations as the system might not converge, however, as the algorithm has the memory of the previously calculated positions in the following iterations will perform better, obtaining a high accuracy position when the UAV is close to the landing point.

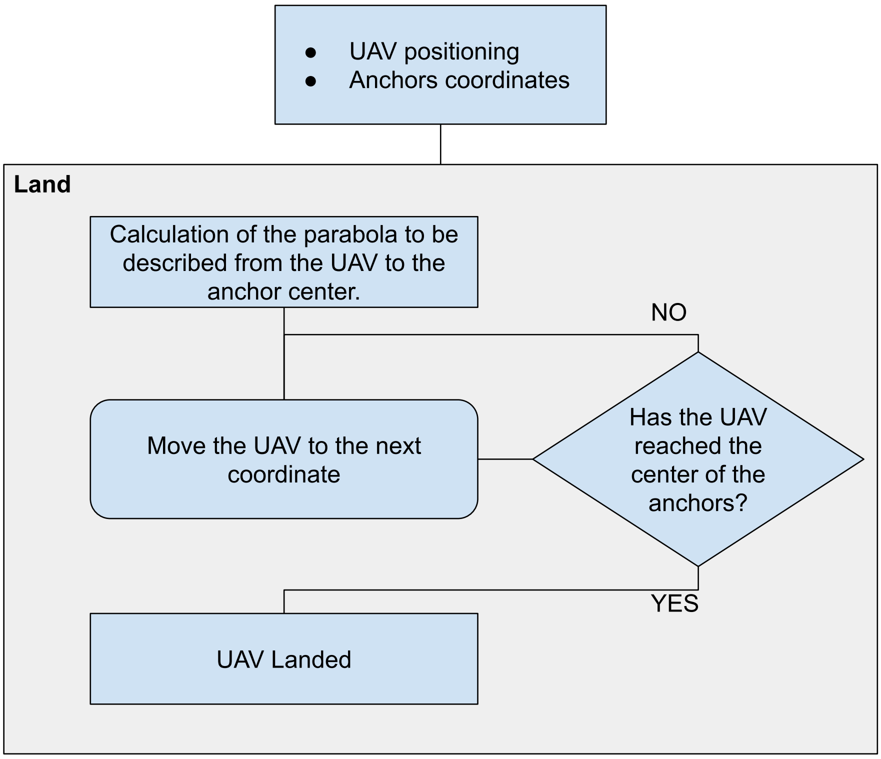


Figure 4.4. *Flow of the UAV trajectory calculation.*

Figure 4.4 represents the logic flow to perform the landing calculating the UAV trajectory from its initial point to the center of the anchor points.

5 RESULTS AND ANALYSIS

In this chapter, the results obtained from the system simulation are presented. The simulation consists of one thousand iterations. On every iteration of this simulation, set 1000 as the limit iteration number, the initial UAV position is generated randomly inside of a cube with a volume of 1000 cubic meters (x, y, and z-axis are 10 meters wide). The output calculation of every iteration has been recorded to be studied and analyzed.

While developing the simulation environment, several assumptions have been made beforehand. However, the results of the simulation proved the incorrectness of those assumptions. Multiple tests have been executed finding a stable solution.

- The first assumption was that the initial guess position would not influence the results. As it can be seen later, the initial guess position adds performance error to the algorithm when calculating the position of the UAV. An optimal initial guess has been proposed in Section 5.1.
- The second assumption was that the γ value in Newton's method would not significantly influence in the system.
- The last assumption was that the system performance would be degraded if the value of the calculated error tolerance is greater than 10 cm.

5.1 Coordinates Initial Guess

In Section 2.5, the influence of choosing a good optimal solution was given. For this reason, it has been used different initial guess solutions at $[x, y, z] = [0, 0, 0]$, $[x, y, z] = [5, 5, 5]$ and $[x, y, z] = [10, 10, 10]$ in order to achieve this goal. In this section, the optimal solution for the system that will be used in the upcoming simulations is proposed.

While executing this experiment, the error tolerance with value 0.1 m was used aiming to achieve the smallest error when calculating the position of the UAV. In order for the system to be able to converge, the error between the calculated distance and the measured distance must be smaller or equal to the error tolerance.

At the same time, the spacing between the anchors will be taken into consideration, which influences the spatial diversity of the antenna, and together with the path loss, the error performance of the system will vary.

During the first calculation, the idea was to represent the maximum positioning error, from the calculated position to the actual location of the UAV, that was calculated by the simulation tool. Every case has been run 1000 times with different UAV location every time. The maximum positioning error was defined as it is the most representative value that will affect directly proposed system.

On every simulation step, the number of iterations needed for convergence was estimated when calculating the UAV coordinates. The meaning of convergence is the probability that the UAV is found with an error smaller than 10 cm. Convergence rate can be improved by allowing to have a higher positioning error between the calculated position and the actual location of the UAV. This might mean that a trade-off between convergence and positioning error should be considered.

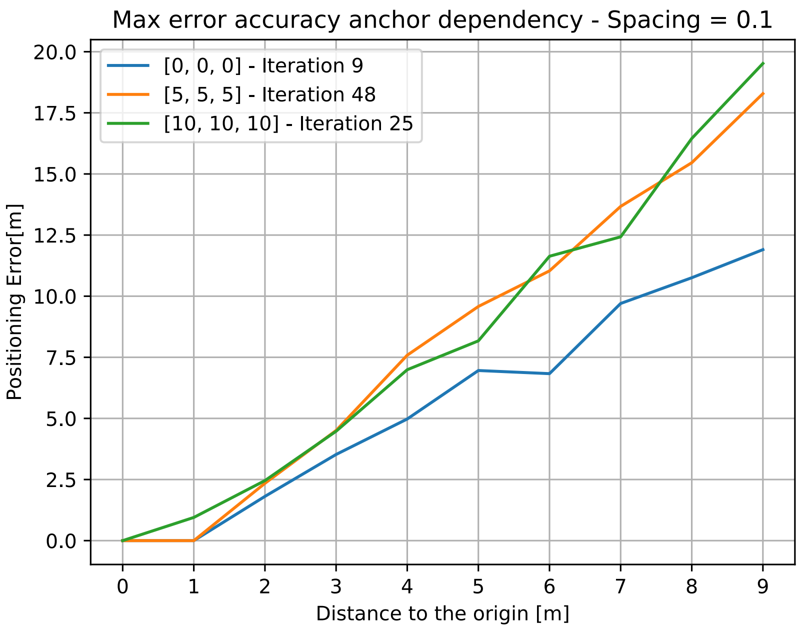


Figure 5.1. *Initial guess performance - Anchor spacing: 10 cm.*

As it is shown in Figure 5.1, the performance of choosing the initial position of the UAV to be at $[x, y, x] = [0, 0, 0]$ is better than the other two initial positions, having a maximum positioning error of 8 meters when it is 9 meters away from the anchor's center. From the other two initial conditions, it can be concluded that both have somewhat similar performance, but position $[x, y, z] = [5, 5, 5]$ have fewer position error in a greater distance from the center of the anchors. The mean number of iterations needed for convergence of the system is represented in the legend next to the initial positioning guess value. The significance of a low number of iterations, for $[x, y, x] = [0, 0, 0]$ is 9 iterations, is that the system is fast. The limit number of iterations has been set to 1000 in the simulation, meaning that for the case when the anchor spacing is 10 cm, the system converges rapidly.

Figure 5.2 shows that the performance of choosing the initial position of the drone to be at $[x, y, x] = [0, 0, 0]$ becomes worse at further distances, while the other two cases

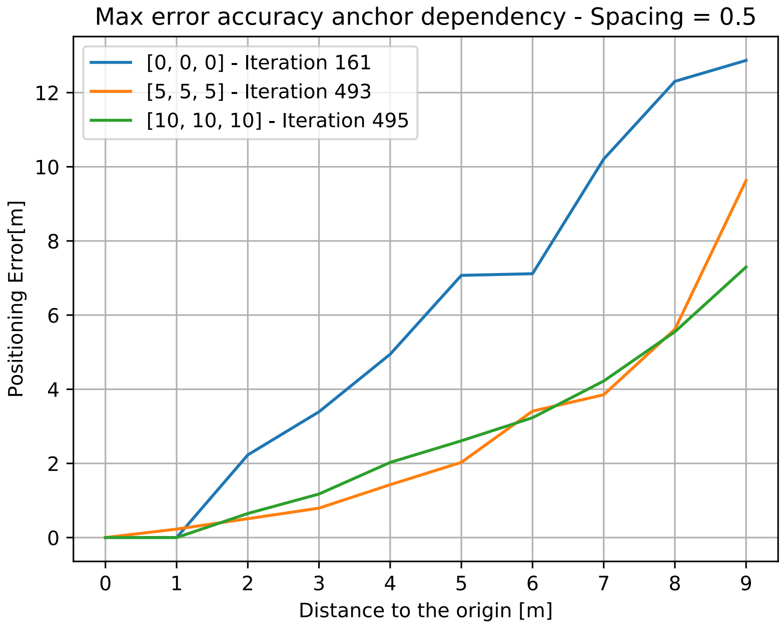


Figure 5.2. *Initial guess performance - Anchor spacing: 50 cm.*

prove that both initial positions have somewhat similar performance, but with a lower position error if $[x, y, z] = [5, 5, 5]$. Even though, this experiment has been ambitious the results seem to have a good performance when the UAV is closer than 4 meters to the anchors. Comparing with respect Fig. 5.1 the number of iterations needed for the system to converge has increased, meaning that the system takes longer to find the optimal solution.

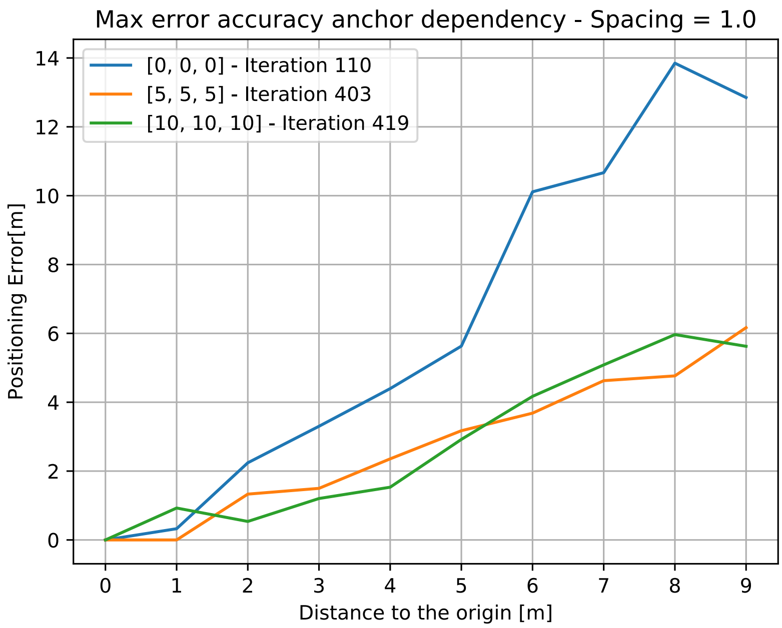


Figure 5.3. *Initial guess performance - Anchor spacing: 1 m.*

Figure 5.3 shows that the number of iterations to convergence with every initial guess increases compared to Figure 5.2 and Figure 5.1. In Figure 5.3, it is also possible to observe less maximum positioning error than in Figure 5.2 supporting the statement about antenna diversity, i.e., with greater anchor spacing – the fewer positioning errors.

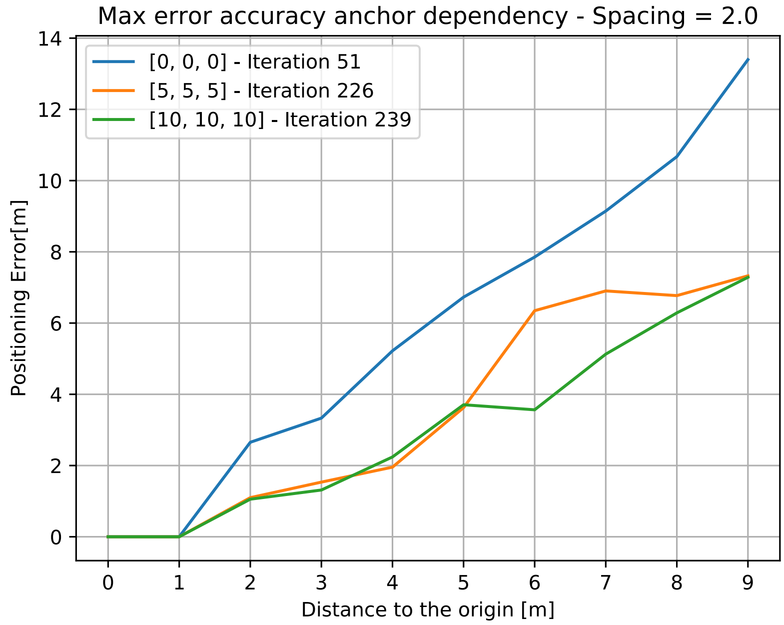


Figure 5.4. *Initial guess performance - Anchor spacing: 2 m.*

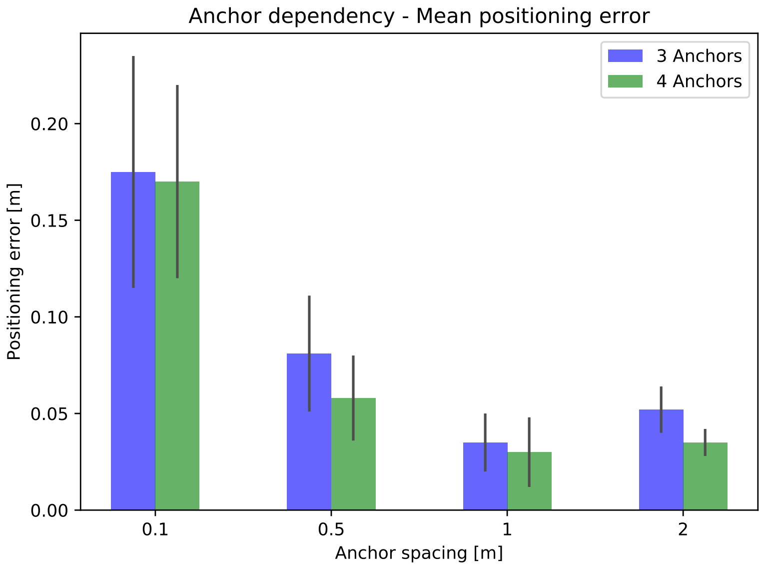
Comparing Figure 5.4 with Figure 5.3, it is possible to spot that the performance of the system does not improve consequently. However, the number of iterations in order to convergence is lower compared to Figure 5.3, making the system faster.

From Figures 5.1 to 5.4, it is possible to conclude that for situations where the anchor spacing is smaller than half a meter, the initial guessed position $[x, y, z] = [0, 0, 0]$ could be used. However, as the proposed simulation tool has to be general, it will be chosen between the other two options, $[x, y, z] = [5, 5, 5]$ and $[x, y, z] = [10, 10, 10]$, for multiple scenarios where the anchor spacing might vary. As the error performance and convergence are a bit better for the case with the initial position $[x, y, z] = [5, 5, 5]$, from now on the simulations will be run with this parameter.

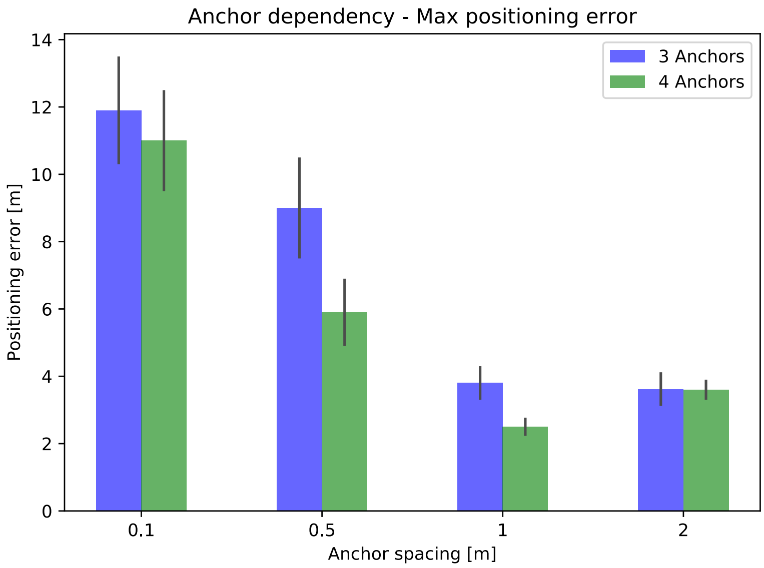
Figures 5.1, 5.2, 5.3, and 5.4 show no positioning errors for cases when the UAV distance to the origin is less than 1 meter. One possible cause is that the simulation has not created the initial position of the UAV in this range. These results are obtained generating 1000 times a random UAV location and calculating the distance error to its calculated position with Gauss-Newton's method, instead of calculating the error while the UAV approaches the center of the platform. Another possible answer to the fact of not having error positioning is due to being a value too small for the figure scale.

5.2 Number and Spacing Between Anchors

It is desired to make the environment portable and save some costs in the real implementation, simulations have been made where two, three and four anchors are used to evaluate the system performance. In this experiment, the error tolerance value was selected as the most restrictive, of 10 cm, in order to make it as accurate as possible. Also, the Newton parameter γ is set to 0.4. Together with the number of anchors, the spacing between the anchors was evaluated in order to get the optimal environment solution.



(a) *Mean positioning error.*



(b) *Max positioning error.*

Figure 5.5. *Anchor's influence on positioning error.*

In this experiment, the performance of the system using a different set of antennas and spacing between them is shown. For Figure 5.5a, the mean of errors at every distance to the center were calculated (as shown in Figure 5.1), and then the overall mean was also obtained. Figure 5.5b shows that the maximum error in the system. With Figure 5.5, it can be seen that, even though the maximum error positioning can be very high, the mean error is lower than 20 cm in every case, asserting that the proposed system is reliable.

Figure 5.5 represents the mean positioning error with the variance represented as a black line and the maximum error with a black line also representing the variance between the test with the lowest error and the one with the highest error. It can be concluded that the positioning error gets reduced having more antennas acting as anchors.

- In case of *three antennas*, a significant improvement in the mean error when the spacing between the antennas is 50 cm compared to two antennas can be observed.
- With *four antennas*, there is an improvement compared to three antennas, but it is not that significant as the previous improvement.

The goal is to have an accurate landing as possible, thus, the positioning error should be minimal when the UAV is close to the center of the anchors. Based on the explained results in Sections 5.1 and 5.2, it can be concluded that the system that best fulfills this criterion is the one with four anchors. Therefore, it is proposed that the system should utilize *four anchors* with a distance between them of 1 m and an initial guess position $[x, y, z] = [5, 5, 5]$.

5.3 Influence of Newton's Method Parameter

It is needed to have a scenario with the most restrictive cases, this means that the parameters – error tolerance and Path-Loss – cannot be modified, but the convergence must be improved. In Section 2.5, the Newton's method was introduced allowing to calculate the initial position of the UAV. The Gauss-Newton's method parameter, γ, in equation (2.12) should improve reaching a solution with a minimized residual error.

The parameters used in the simulation to calculate the influence of γ on the system operation are given in Table 5.1.

Figure 5.6 shows the influence of the parameter γ on positioning error and the number of iterations to achieve convergence. Parameter γ determines what will be the next movement of the drone in every iteration. If γ is too big, the calculated position of the drone will be far from the actual location, and if γ is too small, then the system will take longer to calculate the drone position with an error distance of 10 cm, causing the system to not converge (meaning that the position was not calculated).

According to Figure 5.6, it is evident that the best overall performance is given when it is chosen γ value between 0.3 and 0.5. In all the simulations γ of 0.4 was used to maximize the performance of the system. The relationship between the number of iterations to achieve convergence of the system and the positioning error is also provided in this figure on top of every bar representing the positioning error for the analyzed γ value. With lower

	Notation	Description
Input parameters	Error Tolerance	Allowed positioning error.
	γ	Step size to calculate the next coordinates, Influencing on the speed of convergence of the system.
	Path-Loss Error	Error given by the antenna specification when calculating the free space loss.
	Spacing	Distance between the antennas that forms the anchors.
Output parameters	Positioning error	Residual error between the real and the calculated UAV positioning.
	Convergence	How fast the system has fulfilled that the residual error is smaller than error tolerance

Table 5.1. *Parameters for the calculation of the optimal γ.*

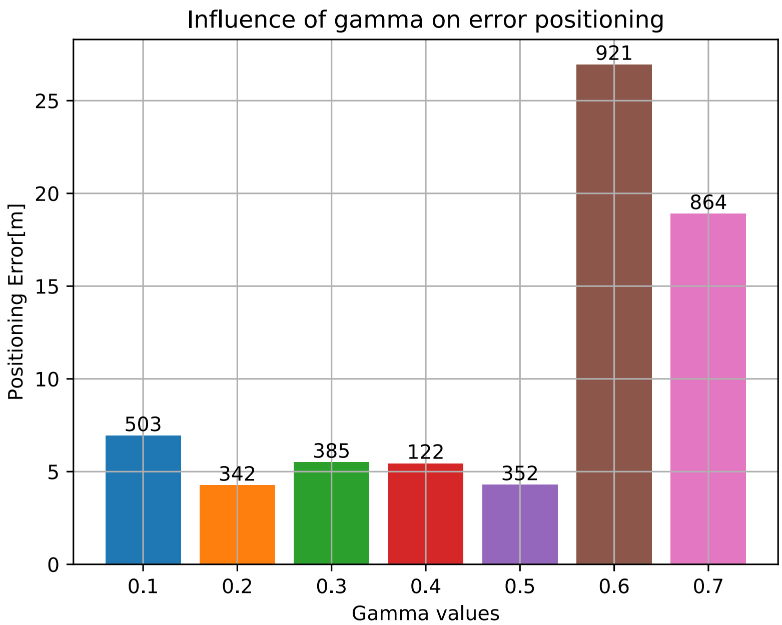


Figure 5.6. *System performance due to Newton's γ.*

positioning error the system converges steadily. However, when γ takes values 0.3 and 0.4 a similar positioning accuracy can be achieved, except the convergence of the system is 30% faster for the case when γ is 0.4. A final analysis is that the needed number of iterations to converge is not directly related to the positioning error, but to the chosen γ value.

5.4 System Performance Depending on the Error Tolerance

When developing the system simulation, the main idea was to be as restrictive as possible when calculating the positioning error. The goal was to have a system with very low positioning error or no error if possible. This is the reason why the system is designed to have a positioning error of 10 cm when calculating the location of the UAV with the aid of the Newton method. However, with the analysis of the results, a low system convergence was noted (the number of times the UAV was found, satisfying the permitted error calculation), of around 60% in most of the cases. The assumed idea of such low convergence, it was a trade-off of having a low positioning error.

Error Tolerance [m]	Init Position	Spacing [m]	Mean [m]	Max [m]	Convergence [Num iterations]
0.1	[0,0,0]	0.1	0.32	8.95	498
		1	0.273	8.98	501
	[5,5,5]	0.1	0.31	16.8	914
		1	0.04	4.13	254
	[10,10,10]	0.1	0.28	16.25	683
		1	0.05	4.5	525
0.3	[0,0,0]	0.1	0.25	8.65	251
		1	0.31	8.99	232
	[5,5,5]	0.1	0.3	17.5	487
		1	0.1	4.87	186
	[10,10,10]	0.1	0.3	16.73	330
		1	0.11	4.51	101

Table 5.2. *Comparison between two error tolerance values.*

The following list of results is detailed in Table 5.2:

- Error tolerance value of 0.1 or 0.3 meters;
- The initial position in Newton's method positioning calculation;
- The antenna spacing that forms the anchor point;
- The overall mean error for 1000 different generated UAV coordinates;
- The maximum error found in the system for 1000 different generated UAV coordinate;
- The convergence of the system, how many times the error tolerance has been satisfied.

The performance of the system related to the mean and the maximum values shows that a chosen error tolerance of 10 cm improves the accuracy but a number of iterations to achieve convergence is very high. On the other hand, the accuracy of the system for error tolerance of 30 cm degrades in less than 10 cm in the mean calculation and around half a meter in the maximum accuracy error, with less number of iterations to reach convergence.

Even though the overall performance of the system is similar for both cases when error tolerance has values of 10 or 30 cm, the convergence is faster for all cases when error tolerance is 30 cm. This analysis does not affect the previously obtained system performance results if an error tolerance of 10 cm was used. However, it was decided to study both cases for implementation of the system in a real-life scenario.

5.5 Unmanned Aerial Vehicle Landing

In this section, the final result of system development is shown. All optimized parameters from Sections 5.3 and 5.4 have been applied for two different landing trajectories.

As this thesis work aims to propose a solution to create an accurate landing system for the aColor project, which will use two different UAVs. One of them is a fixed-wing drone, presented in Figure 2.1b, which landing has to be performed in a diagonal trajectory, as a normal plane would do. The other drone to be considered is a more common and widely used multi-copter, represented in Figure 2.1a. The multi-copters behaves as a helicopter, it performs the take-off and the landing vertically.

The results were calculated with a certain velocity. The velocity is shown as a number of points that the UAV has to overcome in order to reach from its initial position until the center of anchors. For example, in order to increase the speed of the UAV, the number of positions (points) to the center of the anchors will be reduced. In the simulations, it has been chosen a default “velocity”, from the Python library webpage [86]. It can be seen that the default value is 50 points from the initial position of the drone to the center of the anchors. By controlling the speed, the performance of the system can be improved. If the drone travels too fast to the center of the anchors, the system might not converge in time for all the steps that the drone takes. Two examples represented in Figures 5.7 and 5.8, are the simulation proofs that the system can be utilized in real life. When this solution gets applied to the real scenario, a series of tests will need to be done to maximize the performance of the system.

Figures 5.7 and 5.8 show two views, $x, y - plane$, and $z - plane$. The views help to perceive better the 3D image. Blue color represents the actual position of the UAV along the trajectory, and red dots represent the calculated position of the UAV by the algorithm. It is possible to see that when the UAV is far away from the anchor points the calculated position does not coincide with the position of the UAV, but as soon as the Euclidean distance gets reduced the calculated position matches the actual position of the UAV, being almost perfect in the last meters to the landing point.

In Figure 5.7, the UAV appears in the position $[x, y, z] = [5.92, -0.99, 9.73]$ having an initial Euclidean distance of around 11 meters, which is outside the system range. Following Figure 5.5b, the maximum positioning error is given when the system scope is smaller than 10 meters and having a spacing of 1 meter and 4 antennas as anchors are about 2.5 meters. In this case, the approximate error of Euclidean distance is of 3.12 meters. When the UAV reaches approximately the point $[x, y, z] = [4, -0.4, 6]$, having a Euclidean distance of 7 meters to the anchors center, the system becomes almost perfect.

Figure 5.8 shows similar behavior as in Figure 5.7 with an initial location of the UAV at $[x, y, z] = [5.38, 11.55, 11.52]$ and a Euclidean distance of 17 meters to the center of the anchors. In this case, the error of Euclidean distance is approximately 8 meters when the UAV appears in the system range. However, as soon as the Euclidean distance to the center of the anchors is reduced, the positioning error is minimized. When using the multi-rotor type of UAV, a vertical landing shall be performed. This landing starts when the location of the UAV is at $[x, y, z] = [0, 0, 2]$, as it can be seen, the system performance when the UAV is descending is perfect, the red dots match the blue triangles.

The simulation of the system has proven to be almost perfect for the critical points of the different descent methods for both types of UAVs.

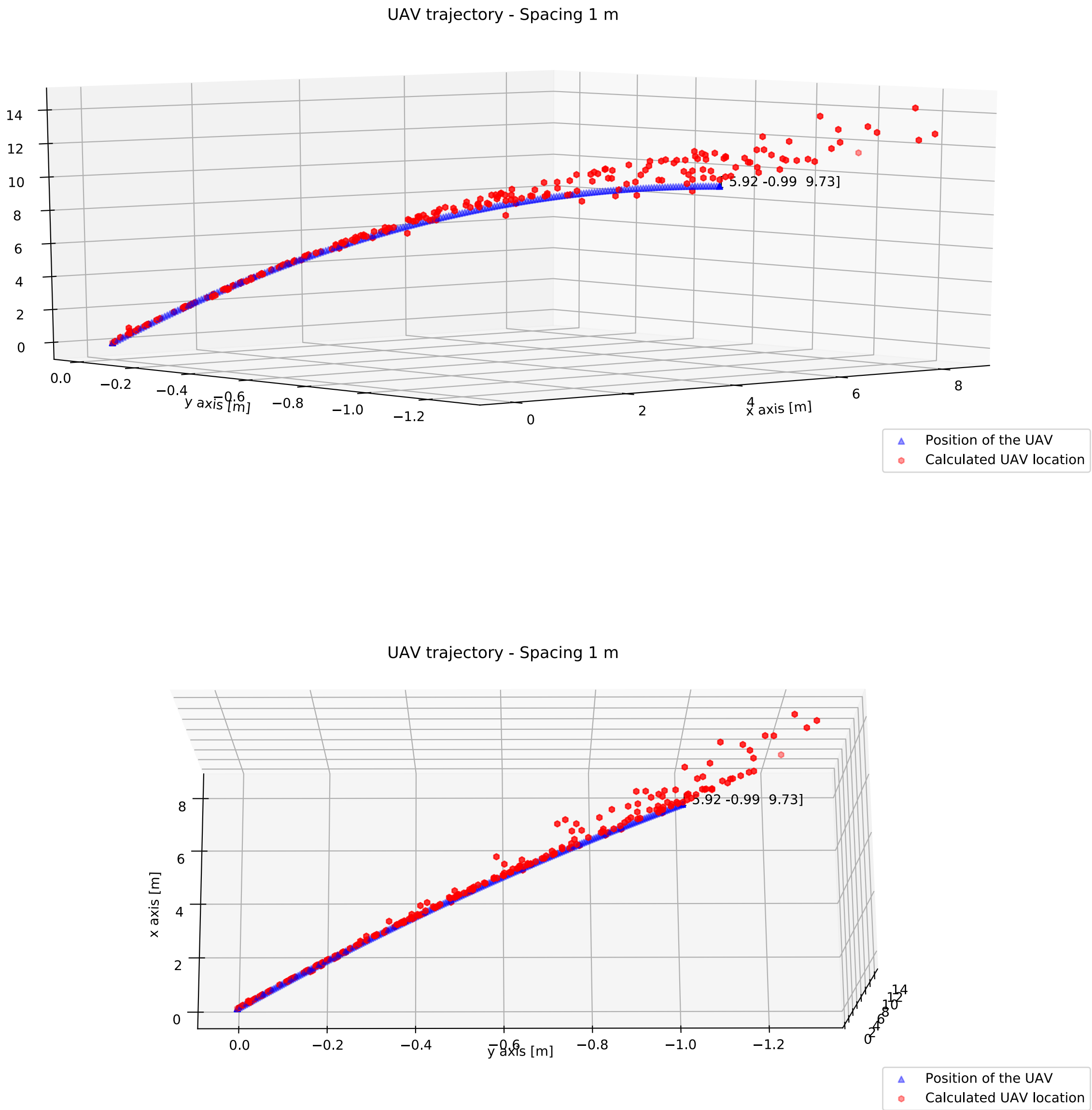


Figure 5.7. *Different views of the landing trajectory – Fixed-wing.*

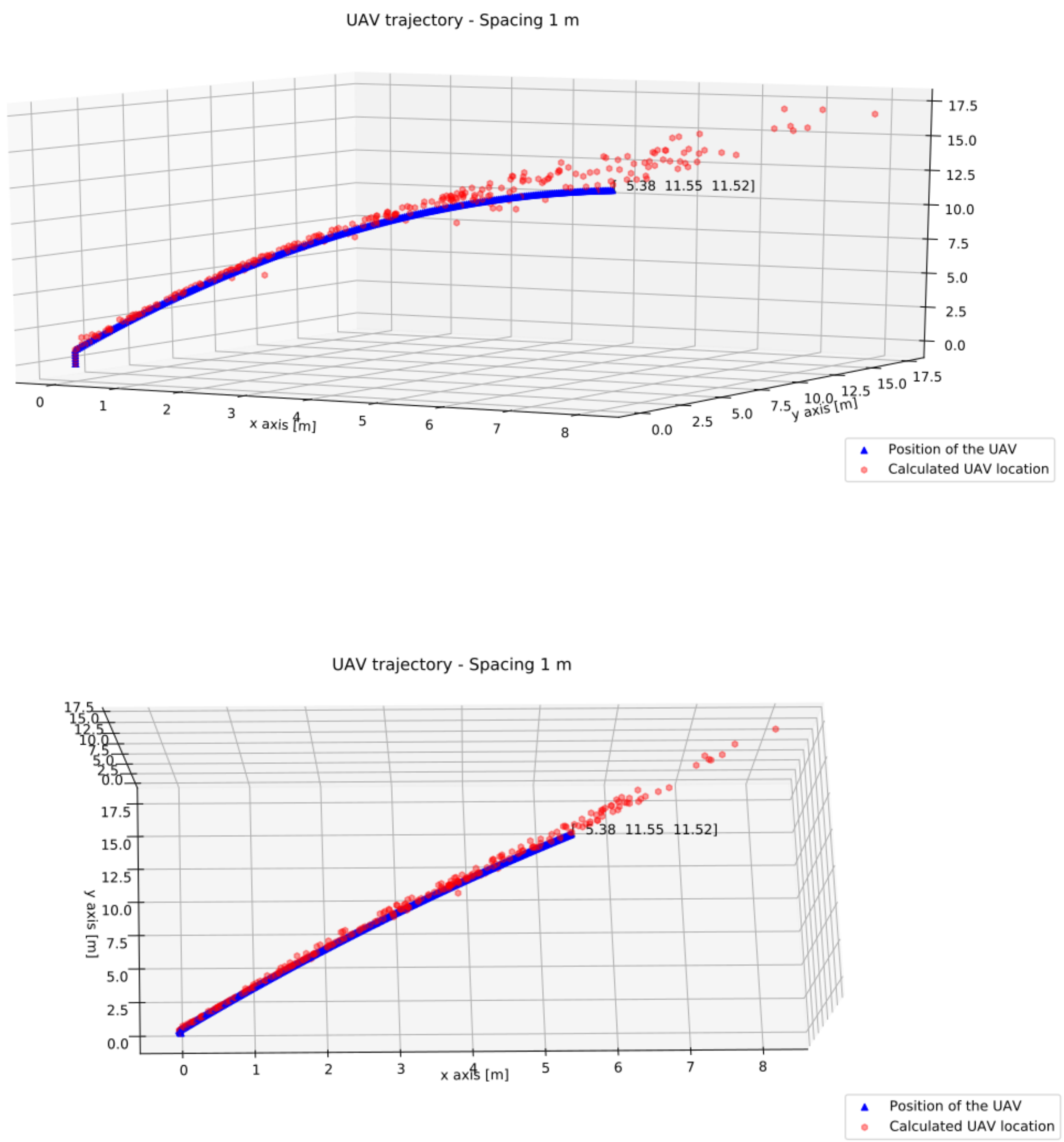


Figure 5.8. *Different views of the landing trajectory – Multi-rotor.*

6 CONCLUSIONS

This Chapter concludes the thesis with a compilation of several significant findings and the author's vision on potential future work.

As a summary of what it has been achieved in this thesis:

- Modeling of a modified Gauss-Newton's method able to find a solution to non-linear equations.
- Creation of a software that can be used as a research tool to simulate localization systems.
- Design of a UAV trajectory in order to perform a soft landing.
- Proposal of a solution to achieve a fully autonomous UAV landing system on a moving platform.

6.1 Conclusions

The number of UAV applications is growing daily. A crucial niche of UAV development is related to Automated UAVs where positioning plays a significant role, especially during landing on moving objects, e.g., vehicles or vessels. In this work, a system for automatic landing support was developed. The thesis focuses on the relationship between positioning errors and system configurations, aiming and keeping the landing surface as small as possible. The moving platform may change its coordinates over time, thus, a pre-decided location cannot be reliable. Moreover, a stable communication link between moving objects and UAV cannot be constantly assumed as there might be some disturbances in the radio link due to environmental and radio factors.

In order to accomplish a fully automated landing system, it was needed a procedure to estimate UAV position coordinates. In order to locate the UAV, it has been used radio signals providing the distance between a set of transmitters and a receiver carried by the UAV. The distance was used as an input parameter to a non-linear equation designed to obtain the UAV coordinates. In order to solve the non-linear equation, a mathematical approach was required for modeling. A modified Newton's method was selected to achieve this goal. It requires an initial parameter and at every iteration, which is modified in order to achieve the solution. Moreover, Newton's method requires a parameter to allow fast convergence. In the proposed system, the method delivered a very low positioning error.

The simulation tool was developed in a user-friendly language allowing the developers to improve the landing systems. The processor capabilities of modern UAVs allow implementing the positioning calculation algorithm that a UAV can process in real time while performing the mission. The system does not require expensive equipment on the transmitter and receiver systems side, as it has been proven by the simulation tool that market-available Wi-Fi equipment is sufficient.

Concerning the take-off and landing procedures, there are two types of UAVs: multi-copter, performing landing vertically, and fixed-wing, performing landing procedure following a curve. In this thesis, these different descent approaches were taken into consideration when designing the landing trajectory for both types. A capture and retrieving system was also proposed in order to avoid damages to the moving platform and the UAV.

Combining all the previous steps, the system designed in this thesis could be proposed as a possible solution to achieve a fully automated UAV because of the accomplishment of a landing system with a high positioning accuracy.

Based on the results of the work, the following conclusions are made:

- In order to achieve faster convergence, an initially assumed positioning has to be studied. In this simulation, it was proposed to use the following coordinates $[x, y, z] = [5, 5, 5]$ as an initial position.
- The number of antennas that compose the anchors should be at least three. However, if there is no restriction, it is advised to use a set of 4 antennas, which results in better performance.
- The anchors spacing should be between 50 cm and 1 meter to maximize the system performance depending on the landing area restrictions.
- Parameter γ in the modified Gauss-Newton's method should be carefully chosen, as γ influences on the pace to reach convergence, and it may happen that the system does not converge. Based on the experiments, it was proposed to use γ between 0.3 and 0.5. The main simulation results were obtained with $\gamma = 0.4$.
- When analyzing the system performance with respect to allowed marginal error, it has been found that a marginal error of 0.1 meters can be used in a very restrictive scenario where there is no possibility of an error. However, if the landing platform is of $1\ m^2$, a marginal error of 0.3 meters can be applied.

The experimental landing trajectory is shown in Figure 5.7 and Figure 5.8. The figures are given under the assumption that the USV is oriented facing the UAV.

6.2 Future Work

Currently, W.I.N.T.E.R and aColor teams are in the final phase of developing a full-scale prototype aiming at testing the developed system in the real-life case.

The author's vision of future development of the thesis results are as following:

- Study of other mathematical methods in order to calculate the UAV coordinates. Evaluation of different algorithms that will assist in increasing the system performance by reaching a quicker convergence.
- Study of other radio technologies and frequencies application for the UAV landing scenario. The use of 5G technology instead of Wi-Fi should be considered as a future direction.
- The study of implementing directional antennas with UAV tracking capabilities to achieve more extended range and lower impact on other network nodes.
- Benchmarking the UAV processor power. Performance analysis of the processor behavior in the calculation process, i.e., how fast the UAV can calculate the coordinates and the trajectory.
- Verification of the simulation tool. The verification implies the comparison of the results obtained by the implemented system and ones produced by the developed simulation tool.
- Development of the UAV capture and retrieving system. As the actual goal of the project is to deploy a fully autonomous system, there is the need to develop a capture mechanism to secure the UAV on the vessel.

REFERENCES

[1] European Comission. *What is Horizon 2020?* Ed. by ec.europa.eu. [Online]. 2015. URL: https://ec.europa.eu/programmes/horizon2020/what-horizon-2020#Article (visited on 04/30/2019).

[2] A. Trotta, M. Di Felice, F. Montori, K. R. Chowdhury, and L. Bononi. Joint Coverage, Connectivity, and Charging Strategies for Distributed UAV Networks. In: *IEEE Transactions on Robotics* 34.4 (2018), 883–900.

[3] H. Ghazzai, M. B. Ghorbel, A. Kassler, M. Hossain, et al. Trajectory Optimization for Cooperative Dual-band UAV Swarms. In: *arXiv preprint arXiv:1807.11578* (2018).

[4] A. Orsino, A. Ometov, G. Fodor, D. Moltchanov, L. Militano, S. Andreev, O. N. Yilmaz, T. Tirronen, J. Torsner, G. Araniti, et al. Effects of Heterogeneous Mobility on D2D-and Drone-Assisted Mission-Critical MTC in 5G. In: *IEEE Communications Magazine* 55.2 (2017), 79–87.

[5] A. Merwaday and I. Guvenc. UAV assisted heterogeneous networks for public safety communications. In: *Proc. of IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. IEEE. 2015, 329–334.

[6] M. Erdelj, M. Król, and E. Natalizio. Wireless Sensor Networks and Multi-UAV Systems for Natural Disaster Management. In: *Computer Networks* 124 (2017), 72–86.

[7] M. Erdelj and E. Natalizio. Drones, Smartphones and Sensors to Face Natural Disasters. In: *Proc. of 4th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*. ACM. 2018, 75–86.

[8] A. R. Girard, A. S. Howell, and J. K. Hedrick. Border patrol and surveillance missions using multiple unmanned air vehicles. In: *Proc. of 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*. Vol. 1. IEEE. 2004, 620–625.

[9] S. G. Gupta, M. M. Ghonge, and P. Jawandhiya. Review of unmanned aircraft system (UAS). In: *International journal of advanced research in computer engineering & technology (IJARCET)* 2.4 (2013), 1646–1658.

[10] S. Saripalli, J. F. Montgomery, and G. S. Sukhatme. Vision-based autonomous landing of an unmanned aerial vehicle. In: *Proc. of IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*. Vol. 3. IEEE. 2002, 2799–2804.

[11] S. Saripalli, J. F. Montgomery, and G. S. Sukhatme. Visually guided landing of an unmanned aerial vehicle. In: *IEEE transactions on robotics and automation* 19.3 (2003), 371–380.

[12] W. Kong, D. Zhou, D. Zhang, and J. Zhang. Vision-based autonomous landing system for unmanned aerial vehicle: A survey. In: *International Conference on Mul-*

tisensor Fusion and Information Integration for Intelligent Systems (MFI). IEEE. 2014, 1–8.

[13] T. Templeton, D. H. Shim, C. Geyer, and S. S. Sastry. Autonomous vision-based landing and terrain mapping using an MPC-controlled unmanned rotorcraft. In: *Proc. of IEEE International Conference on Robotics and Automation*. IEEE. 2007, 1349–1356.

[14] Rieke, Matthes and Foerster, Theodor and Geipel, Jakob and Prinz, Torsten. High-precision positioning and real-time data processing of UAV systems. In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 38.1/C22 (2011).

[15] United States Air Force. *GPS Accuracy*. 2017. URL: https://www.gps.gov/systems/gps/performance/accuracy/ (visited on 03/16/2019).

[16] M. Gowda, J. Manweiler, A. Dhekne, R. R. Choudhury, and J. D. Weisz. Tracking drone orientation with multiple GPS receivers. In: *Proc. of 22nd annual international conference on mobile computing and networking*. ACM. 2016, 280–293.

[17] J. Villa, J. Aaltonen, and K. T. Koskinen. Model-based path planning and obstacle avoidance architecture for a twin jet Unmanned Surface Vessel. In: *Proc. of 3rd IEEE International Conference on Robotic Computing (IRC)*. Feb. 2019, 427–428. DOI: 10.1109/IRC.2019.00083.

[18] Kari T. Koskinen. *Autonomous and Collaborative Offshore Robotics (aCOLOR)*. 2017. URL: https://techfinland100.fi/mita-rahoitamme/tutkimus/tulevaisuuden-tekijat/autonomous-and-collaborative-offshore-robotics-acolor/ (visited on 04/15/2019).

[19] Jose Villa Escusol. *aColor achieves first year goals of the project with the Unmanned Surface Vessel as a main topic*. 2018. URL: https://techfinland100.fi/category/autonomous-and-collaborative-offshore-robotics-acolor/ (visited on 04/15/2019).

[20] E. Vattapparamban, İ. Güvenç, A. İ. Yurekli, K. Akkaya, and S. Uluağaç. Drones for smart cities: Issues in cybersecurity, privacy, and public safety. In: *Proc. of International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE. 2016, 216–221.

[21] P. Liu, A. Y. Chen, Y.-N. Huang, J.-Y. Han, J.-S. Lai, S.-C. Kang, T.-H. Wu, M.-C. Wen, M.-H. Tsai, et al. A review of rotorcraft unmanned aerial vehicle (UAV) developments and applications in civil engineering. In: *Smart Struct. Syst* 13.6 (2014), 1065–1094.

[22] S. M. Adams and C. J. Friedland. A survey of unmanned aerial vehicle (UAV) usage for imagery collection in disaster research and management. In: *Proc. of 9th International Workshop on Remote Sensing for Disaster Response*. Vol. 8. 2011.

[23] F. G. Costa, J. Ueyama, T. Braun, G. Pessin, F. S. Osório, and P. A. Vargas. The use of unmanned aerial vehicles and wireless sensor network in agricultural applications. In: *Proc. of IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 2012, 5045–5048.

[24] A. Danilov, U. D. Smirnov, and M. Pashkevich. The system of the ecological monitoring of environment which is based on the usage of UAV. In: *Russian journal of ecology* 46.1 (2015), 14–19.

[25] A. Pagnano, M. Höpf, and R. Teti. A roadmap for automated power line inspection. Maintenance and repair. In: *Procedia Cirp* 12 (2013), 234–239.

[26] D. Orfanus, E. P. de Freitas, and F. Eliassen. Self-Organization as a Supporting Paradigm for Military UAV Relay Networks. In: *IEEE Communications Letters* 20.4 (Apr. 2016), 804–807. ISSN: 1089-7798. DOI: 10.1109/LCOMM.2016.2524405.

[27] M. A. Ma'sum, M. K. Arrofi, G. Jati, F. Arifin, M. N. Kurniawan, P. Mursanto, and W. Jatmiko. Simulation of intelligent Unmanned Aerial Vehicle (UAV) For military surveillance. In: *Proc. of International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. Sept. 2013, 161–166. DOI: 10.1109/ICACSIS.2013.6761569.

[28] J. Y. Chen. UAV-guided navigation for ground robot tele-operation in a military reconnaissance environment. In: *Ergonomics* 53.8 (2010). PMID: 20658388, 940–950. DOI: 10.1080/00140139.2010.500404. eprint: https://doi.org/10.1080/00140139.2010.500404. URL: https://doi.org/10.1080/00140139.2010.500404.

[29] M. Hassanalian and A. Abdelkefi. Classifications, applications, and design challenges of drones: A review. In: *Progress in Aerospace Sciences* 91 (2017), 99–131.

[30] U. D. of Transportation John A. Volpe National Transportation Systems Center. *Unmanned Aircraft System (UAS) Service Demand 2015 - 2035 Literature Review and Projections of Future Usage*. 2013. URL: https://fas.org/irp/program/collect/service.pdf (visited on 03/2019).

[31] O. Evangelatos and J. Rolim. An Airborne Wireless Sensor Network for Ambient Air Pollution Monitoring. In: *SENSORNETS* (2015), 231–239.

[32] W. Yi, K. Lo, T. Mak, K. Leung, Y. Leung, and M. Meng. A survey of wireless sensor network based air pollution monitoring systems. In: *Sensors* 15.12 (2015), 31392–31427.

[33] D. W. Casbeer, D. B. Kingston, R. W. Beard, and T. W. McLain. Cooperative forest fire surveillance using a team of small unmanned air vehicles. In: *International Journal of Systems Science* 37.6 (2006), 351–360.

[34] L. Merino, F. Caballero, J. R. Martinez-De-Dios, I. Maza, and A. Ollero. An unmanned aircraft system for automatic forest fire monitoring and measurement. In: *Journal of Intelligent & Robotic Systems* 65.1-4 (2012), 533–548.

[35] K. Nonami. Prospect and recent research & development for civil use autonomous unmanned aircraft as UAV and MAV. In: *Journal of system Design and Dynamics* 1.2 (2007), 120–128.

[36] A. Goodchild and J. Toy. Delivery by drone: An evaluation of unmanned aerial vehicle technology in reducing CO2 emissions in the delivery service industry. In: *Transportation Research Part D: Transport and Environment* 61 (2018), 58–67.

[37] J. Park, S. Kim, and K. Suh. A comparative analysis of the environmental benefits of drone-based delivery services in urban and rural areas. In: *Sustainability* 10.3 (2018), 888.

[38] H. Saha, S. Basu, S. Auddy, R. Dey, A. Nandy, D. Pal, N. Roy, S. Jasu, A. Saha, S. Chattopadhyay, et al. A low cost fully autonomous GPS (Global Positioning System) based quad copter for disaster management. In: *IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE. 2018, 654–660.

[39] F. Malandrino, C.-F. Chiasserini, C. Casetti, L. Chiaraviglio, and A. Senacheribbe. Planning UAV activities for efficient user coverage in disaster areas. In: *Ad Hoc Networks* 89 (2019), 177–185.

[40] S. A. R. Naqvi, S. A. Hassan, H. Pervaiz, and Q. Ni. Drone-aided communication as a key enabler for 5G and resilient public safety networks. In: *IEEE Communications Magazine* 56.1 (2018), 36–42.

[41] D. Kozyrev, A. Ometov, D. Moltchanov, V. Rykov, D. Efrosinin, T. Milovanova, S. Andreev, and Y. Koucheryavy. Mobility-centric analysis of communication offloading for heterogeneous internet of things devices. In: *Wireless Communications and Mobile Computing* 2018 (2018).

[42] A. Ometov, V. Petrov, S. Bezzateev, S. Andreev, Y. Koucheryavy, and M. Gerla. Challenges of Multi-Factor Authentication for Securing Advanced IoT (A-IoT) Applications. In: *arXiv preprint arXiv:1901.06977* (2019).

[43] S. Sekander, H. Tabassum, and E. Hossain. Multi-tier drone architecture for 5G/B5G cellular networks: Challenges, trends, and prospects. In: *IEEE Communications Magazine* 56.3 (2018), 96–103.

[44] B. Vergouw, H. Nagel, G. Bondt, and B. Custers. Drone technology: types, payloads, applications, frequency spectrum issues and future developments. In: *The Future of Drone Use*. Springer, 2016, 21–45.

[45] R. Kovalchukov, D. Moltchanov, A. Samuylov, A. Ometov, S. Andreev, Y. Koucheryavy, and K. Samouylov. Analyzing Effects of Directionality and Random Heights in Drone-based mmWave Communication. In: *IEEE Transactions on Vehicular Technology* 67.10 (2018), 10064–10069.

[46] R. Kovalchukov, D. Moltchanov, A. Samuylov, A. Ometov, S. Andreev, Y. Koucheryavy, and K. Samouylov. Evaluating SIR in 3D Millimeter-Wave Deployments: Direct Modeling and Feasible Approximations. In: *IEEE Transactions on Wireless Communications* 18.2 (Feb. 2019), 879–896. ISSN: 1536-1276. DOI: 10.1109/TWC.2018.2886188.

[47] FOXTECH. *GAIA 100 Carbon Fiber Monocoque Hexacopter Frame*. 2018. URL: https://www.foxtechfpv.com/product/hexacopter/gaia100-sy10.jpg (visited on 04/15/2019).

[48] Applied Aeronautics. *ALBATROSS UAV*. 2019. URL: http://store.appliedaeronautics.com/albatross-uav-ready-to-fly/# (visited on 04/15/2019).

[49] A. Pyattaev, D. Solomitckii, and A. Ometov. 3D Folded Loop UAV Antenna Design. In: *International Conference on Wired/Wireless Internet Communication*. Springer. 2018, 269–281.

[50] K. Bullington. Radio propagation for vehicular communications. In: *IEEE Transactions on Vehicular Technology* 26.4 (Nov. 1977), 295–308. ISSN: 0018-9545. DOI: 10.1109/T-VT.1977.23698.

[51] D. Ndzi, L. Kamarudin, E. Mohammad, A. Zakaria, R. Ahmad, M. Fareq, A. Shakaff, and M. Jafaar. Vegetation attenuation measurements and modeling in plantations for wireless sensor network planning. English. In: *Progress in Electromagnetics Research B* 36 (2012), 283–301. ISSN: 1937-6472. DOI: 10.2528/PIERB1109190.

[52] A. Ometov, D. Moltchanov, M. Komarov, S. V. Volvenko, and Y. Koucheryavy. Packet Level Performance Assessment of mmWave Backhauling Technology for 3GPP NR Systems. In: *IEEE Access* 7 (2019), 9860–9871.

[53] T. Tamir. On radio-wave propagation in forest environments. In: *IEEE Transactions on Antennas and Propagation* 15.6 (Nov. 1967), 806–817. ISSN: 0018-926X. DOI: 10.1109/TAP.1967.1139054.

[54] R. Umar, S. S. Sulan, A. W. Azlan, Z. A. Ibrahim, W. Mokhtar, and N. H. Sabri. Radio frequency interference: The study of rain effect on radio signal attenuation. In: *Malaysian Journal of Analytical Sciences* 19.5 (2015), 1093–1098.

[55] L. J. Ippolito. *Radiowave propagation in satellite communications*. Springer Science & Business Media, 2012.

[56] A. Pyattaev, K. Johnsson, S. Andreev, and Y. Koucheryavy. Communication challenges in high-density deployments of wearable wireless devices. In: *IEEE Wireless Communications* 22.1 (Feb. 2015), 12–18. ISSN: 1536-1284. DOI: 10.1109/MWC.2015.7054714.

[57] R. G. Vaughan and J. B. Andersen. Antenna diversity in mobile communications. In: *IEEE Transactions on Vehicular Technology* 36.4 (Nov. 1987), 149–172. ISSN: 0018-9545. DOI: 10.1109/T-VT.1987.24115.

[58] J. H. Winters, J. Salz, and R. D. Gitlin. The impact of antenna diversity on the capacity of wireless communication systems. In: *IEEE transactions on Communications* 42.234 (1994), 1740–1751.

[59] C. B. Dietrich, K. Dietze, J. R. Nealy, and W. L. Stutzman. Spatial, polarization, and pattern diversity for wireless handheld terminals. In: *IEEE transactions on antennas and propagation* 49.9 (2001), 1271–1281.

[60] J. H. Winters, J. Salz, and R. D. Gitlin. The impact of antenna diversity on the capacity of wireless communication systems. In: *IEEE Transactions on Communications* 42.234 (Feb. 1994), 1740–1751. ISSN: 0090-6778. DOI: 10.1109/TCOMM.1994.582882.

[61] A. A. Gharapurkar, A. F. Jahromi, R. B. Bhat, and W. Xie. Semi-active control of aircraft landing gear system using H-infinity control approach. In: *2013 International Conference on Connected Vehicles and Expo (ICCVE)*. Dec. 2013, 679–686. DOI: 10.1109/ICCVE.2013.6799877.

[62] R. Clarke. AUTONOMOUS MULTI-ROTOR AERIAL VEHICLE WITH LANDING AND CHARGING SYSTEM. US Patent 2017/0139409 A1.

[63] H. Chae, J. Park, H. Song, Y. Kim, and H. Jeong. The IoT based automate landing system of a drone for the round-the-clock surveillance solution. In: *Proc. of IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. July 2015, 1575–1580. DOI: 10.1109/AIM.2015.7222767.

[64] R. Barták, A. Hraško, and D. Obdržálek. A controller for autonomous landing of AR.Drone. In: *The 26th Chinese Control and Decision Conference (2014 CCDC)*. May 2014, 329–334. DOI: 10.1109/CCDC.2014.6852167.

[65] M. Skoczylas. Vision analysis system for autonomous landing of micro drone. In: *acta mechanica et automatica* 8.4 (2014), 199–203.

[66] J. M. A. Michael A. WattsGeorge Raymond Root. UAV recovery system. US Patent 7.219,856 B2.

[67] A. Ometov, S. Bezzateev, V. Davydov, A. Shchesniak, P. Masek, E. S. Lohan, and Y. Koucheryavy. Positioning Information Privacy in Intelligent Transportation Systems: An Overview and Future Perspective. In: *Sensors* 19.7 (2019), 1603.

[68] I. A. Getting. Perspective/navigation-the global positioning system. In: *IEEE spectrum* 30.12 (1993), 36–38.

[69] E. S. Lohan, A. Alén-Savikko, L. Chen, K. Järvinen, H. Leppäkoski, H. Kuusniemi, and P. Korpisaari. 5G Positioning: Security and Privacy Aspects. In: *A Comprehensive Guide to 5G Security* (2018), 281–320.

[70] J. Elgot. *UK may never recover £1.2bn invested in EU Galileo satellite system*. Ed. by www.theguardian.com. [Online]. Nov. 2018. (Visited on 04/30/2019).

[71] European Global Navigation Satellite System Agency. *Who's involved in Galileo?* Ed. by gsc-europa.eu. [Online]. 2019. URL: https://www.gsc-europa.eu/helpdesk/faqs (visited on 04/30/2019).

[72] P. Misra and P. Enge. Global Positioning System: signals, measurements and performance second edition. In: *Massachusetts: Ganga-Jamuna Press* (2006).

[73] H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of wireless indoor positioning techniques and systems. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 37.6 (2007), 1067–1080.

[74] E. S. Lohan, J. Talvitie, P. Figueiredo e Silva, H. Nurminen, S. Ali-Löytty, and R. Piché. Received signal strength models for WLAN and BLE-based indoor positioning in multi-floor buildings. In: *Proc. of International Conference on Location and GNSS (ICL-GNSS)*. June 2015, 1–6. DOI: 10.1109/ICL-GNSS.2015.7217154.

[75] E. S. Lohan, K. Koski, J. Talvitie, and L. Ukkonen. WLAN and RFID Propagation channels for hybrid indoor positioning. In: *International Conference on Localization and GNSS 2014 (ICL-GNSS 2014)*. June 2014, 1–6.

[76] A. Basiri, E. S. Lohan, T. Moore, A. Winstanley, P. Peltola, C. Hill, P. Amirian, and P. F. e Silva. Indoor location based services challenges, requirements and usability of current solutions. In: *Computer Science Review* 24 (2017), 1–12.

[77] J. Yan, C. Tiberius, G. Bellusci, and G. Janssen. Feasibility of Gauss-Newton method for indoor positioning. In: *Proc. of IEEE/ION Position, Location and Navigation Symposium*. May 2008, 660–670. DOI: 10.1109/PLANS.2008.4569986.

[78] H. O. Hartley. The Modified Gauss-Newton Method for the Fitting of Non-Linear Regression Functions by Least Squares. In: *Technometrics* 3.2 (1961), 269–280. DOI: 10.1080/00401706.1961.10489945.

[79] D. Marquardt. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. In: *Journal of the Society for Industrial and Applied Mathematics* 11.2 (1963), 431–441. DOI: 10.1137/0111030.

[80] S. Gratton, A. Lawless, and N. Nichols. Approximate Gauss–Newton Methods for Nonlinear Least Squares Problems. In: *SIAM Journal on Optimization* 18.1 (2007), 106–132. DOI: 10.1137/050624935. eprint: https://doi.org/10.1137/050624935. URL: https://doi.org/10.1137/050624935.

[81] G. K. Smyth. Nonlinear Regression. In: *Encyclopedia of Environmetrics*. American Cancer Society, 2006. DOI: 10.1002/9780470057339.van017.

[82] R. Wang. *Gauss-Newton algorithm for nonlinear models*. 2015. URL: http://fourier.eng.hmc.edu/e176/lectures/NM/node36.html (visited on 03/16/2019).

[83] S. C. E. Ron S. Dembo and T. Steihaug. Inexact Newton Methods. In: *SIAM Journal on Numerical Analysis* (1982).

[84] J. Pokorny, A. Ometov, P. Pascual, C. Baquero, P. Masek, A. Pyattaev, A. Garcia, C. Castillo, S. Andreev, J. Hosek, et al. Concept Design and Performance Evaluation of UAV-based Backhaul Link With Antenna Steering. In: *Journal of Communications and Networks* 20.5 (2018), 473–483.

[85] Python Software Foundation. *Python*. 2019. URL: https://www.python.org/about/ (visited on 04/10/2019).

[86] SciPy.org. *numpy.linspace*. 2019. URL: https://docs.scipy.org/doc/numpy/reference/generated/numpy.linspace.html (visited on 04/15/2019).

[87] WinterSim. *WINTERSIM – AN OPEN SIMULATION PLATFORM FOR SIMULATION STUDIES OF WIRELESS SYSTEMS*. 2014. URL: http://winter-group.net/downloads/ (visited on 04/25/2019).

[88] nsnam. *ns-3 Netowrk Simulator*. 2019. URL: https://www.nsnam.org/ (visited on 04/25/2019).

[89] IEEE. *Wi-Fi*. 2016. URL: https://standards.ieee.org/standard/802_11-2016.html (visited on 04/25/2019).

[90] 3GPP. *LTE specification*. 2017. URL: https://www.3gpp.org/release-14 (visited on 04/25/2019).

[91] 3GPP. *5G specification*. 2020. URL: https://www.3gpp.org/release-16 (visited on 04/25/2019).

A DEVELOPED CODE

In this chapter the author presents the developed code to implement the simulation tool used in this master thesis.

A.1 Main Thread

The following code is the main thread of the simulation environment. The initial parameters are defined here. This code is generated to create the statistics of the system, as well as to call the landing method for every new generated drone position.

```
import matplotlib.pyplot as plt

from mpl_toolkits.mplot3d import Axes3D
import numpy as np
from scipy.spatial.distance import euclidean

from newton_method import *

from lib import free_space_path_loss, friis_range
from sympy import *
from trajectory import *

from lib.vectors import norm

np.seterr(divide='ignore', invalid='ignore')

ANCHOR_SPACING = [0.1,0.5,1,2]
EPS = [0.3]
ALPHA = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7]
FREQ = 2.4e9
BOX = 10
N = 100

ORIGIN_POS = np.zeros(3)

INITIAL_GUESS = [[0,0,0],[5,5,5],[10,10,10]]
```

```
def calc_matrix_err_dist(vector_err, vector_dist, max_to_origin):
    new_matrix = np.zeros([max_to_origin, N])

    for j in range(new_matrix.shape[0]):
        for i in range(new_matrix.shape[1]):
            if j-1 < vector_dist[i] < j:
            new_matrix[j, i] = vector_err[i]

    out_matrix = np.matrix(new_matrix)
    mean = out_matrix.mean(1)
    max = out_matrix.max(1)

    return out_matrix, mean, max

count = 0
iteration = 0

def calc_center(anchors):
    if len(anchors) == 4:
        center = [(anchors[0][0] + anchors[2][0]) / 2,
            (anchors[0][1] + anchors[2][1]) / 2, 0]  # 4anch
    elif len(anchors) == 3:
        center = [(anchors[0][0]+anchors[0][1]+anchors[0][2])/3 ,
            (anchors[1][0]+anchors[1][1]+anchors[1][2])/3, 0] #3anch
    elif len(anchors) == 2:
        center = [(anchors[0][0]+anchors[1][0])/2 ,
            (anchors[0][1]+anchors[1][1])/2, 0] #2anchors

    return center

def friis_distances(pos, anchors):
    dists = np.array([euclidean(pos, p) for p in anchors])
    path_losses = free_space_path_loss(dists, FREQ)
    path_losses += np.random.normal(loc=10**0.1)
    return friis_range(path_losses, FREQ)


stats_by_beans = np.zeros([20, N])

stat_errs = np.zeros([len(ANCHOR_SPACING), N])
orign_dist_matrix = np.zeros([len(ANCHOR_SPACING), N])
```

```
stat_pos = np.zeros([len(ANCHOR_SPACING), N, 3])
stat_pos_est = np.zeros([len(ANCHOR_SPACING), N, 3])


for k, spacing in enumerate(ANCHOR_SPACING):

    plt.figure()

    ANCHOR_POS = [np.array([
        [0, -1, 0],
        [1,  0, 0],
        [0, 1, 0],
        [-1, 0, 0],]) * spacing]

    for j, anchors in enumerate(ANCHOR_POS):
        outs = []
        means = []
        maxs = []
        POS = np.vstack([np.random.uniform(-BOX, BOX, [2, N]),
                np.random.uniform(0.1, BOX, N)]).T

        errs = np.zeros(N)
        dist_to_origin = np.zeros(N)
        pos_est = np.zeros([N, 3])

        count = 0
        iteration = 0
        for i, drone_pos in enumerate(POS):
            iteration += 1
            recovered_dists = friis_distances(drone_pos,
                    anchors)
            conv, drone_guess = guess_position(initial_guess,
                    anchors, recovered_dists, False)

            if conv < 999:
                cc.append(conv)
                count += 1

            errs[i] = euclidean(drone_guess, drone_pos)

            anchor_center = calc_center(anchors)
```

```
            dist_to_origin[i] = euclidean(drone_pos,
                    anchor_center)

            pos_est[i] = drone_guess

            land(anchors, anchor_center, drone_guess, True)

        dd = pos_est - POS

        stat_errs[j] = errs
        orign_dist_matrix[j] = dist_to_origin

        out, mean, max = calc_matrix_err_dist(stat_errs[j],
        orign_dist_matrix[j],
        max_to_origin=10)
        # plt.figure()
        # plt.plot(out, '*')
        # plt.bar(j, np.mean(mean))
        # plt.bar(j, np.max(max))
        # plt.bar(j, count/iteration*100)

        plt.plot(mean, label='{}␣-␣Iteration␣{}%'.format(initial_guess,
                int(np.round(np.mean(cc),0)))
        plt.plot(max, label='{}␣-␣Iteration␣{}%'.format(initial_guess,
                int(np.round(np.mean(cc),0)))

        means.append(mean)
        maxs.append(max)
        print("Has␣converged␣in␣iteration␣{}".format(iteration))

    print(initial_guess, np.mean(means), spacing)
    print(initial_guess, np.max(maxs), spacing)

x = np.arange(len(ANCHOR_POS))
plt.xlabel("Distance␣to␣the␣origin␣[m]")
plt.xticks(np.arange(out.shape[0]))

plt.ylabel("Positioning␣[m]")

plt.legend()
plt.grid()
plt.title("Mean␣error␣accuracy␣anchor␣dependency␣-␣Spacing␣=␣{}"
```

```
            .format(float("{0:.2f}".format(spacing))))

plt.legend()
plt.show(block=True)
```

A.2 Positioning Calculation

The following code it is how the author has implemented the positioning algorithm based on a modified Gauss-Newton method. This code is called every time the drone moves to a new coordinate.

```
import numpy as np
from lib.vectors import norm
import matplotlib.pyplot as plt
from lib import free_space_path_loss, friis_range
from scipy.spatial.distance import euclidean


def guess_position(initial_guess, anchor_positions, dist_meas,
                printing):
    eps = 0.3
    gamma = 0.4
    guess_pos = initial_guess.copy()
    num_iterations = 1000

    for a in range(num_iterations):
        nudge = np.zeros(3)

        for i in range(anchor_positions.shape[0]):
            dv = anchor_positions[i] - guess_pos
            dist = norm(dv)
            err = dist - dist_meas[i]
            if abs(err) <= float(eps):
                continue
            else:
                dv = dv / dist
                nudge += dv * err * gamma

        if not nudge.all():
            break
        guess_pos += nudge

        if guess_pos[2] < 0:
```

```
                    guess_pos[2] = 1

    return a, guess_pos
```

A.3 Unmanned Areial Vehicle Landing

This last piece of code defines the trajectory algorithm. While moving the drone to the desired landing point, the previous code is called to calculate the new positioning of the drone. After this code is run, a figure with the comparison between the designed landing trajectory and the estimated coordinates is represented.

```
import math as m
import numpy as np
from mpl_toolkits.mplot3d import Axes3D
from scipy.spatial.distance import euclidean
from sympy import *
import matplotlib.pyplot as plt

from newton_method import *

from lib import free_space_path_loss, friis_range


def friis_distances(pos, anchors):
    dists = np.array([euclidean(pos, p) for p in anchors])
    path_losses = free_space_path_loss(dists, 2.4e9)
    path_losses += np.random.normal(loc=10**0.1)
    return friis_range(path_losses, 2.4e9)

def calc_a(anchor, drone):
    return (0 - drone[2])/((anchor[0]-drone[0])**2)

def calc_parab(a, drone):
    x = Symbol('x')
    return a*((x-drone[0])**2)+drone[2]

def parabola_values(x_pos, y):
    x = Symbol('x')
    y = y.apart(x)
    print(y)
    lam_x = lambdify(x, y, modules=['numpy'])
    return lam_x(x_pos)
```

```python
def check_z_pos(drone):
    if drone[2] < 1:
        print('\033[91m'+'The␣drone␣is␣flying␣too␣low'+'\033[0m')
        drone[2] = 1.5

def land(anchors, anchor_center, drone, printing=False):
    check_z_pos(drone)

    x = Symbol('x')

    a = calc_a(anchor_center, drone)

    z = calc_parab(a, drone)
    lam_x = lambdify(x, z, modules=['numpy'])
    x_pos = np.linspace(drone[0], anchor_center[0],
    abs(euclidean(anchor_center, drone))*20)
    z_pos = lam_x(x_pos)
    y_pos = np.linspace(drone[1], anchor_center[1], len(x_pos))

    if printing:
        fig = plt.figure()
        ax = fig.add_subplot(111, projection='3d')

    p = 0

    while drone[0] != x_pos[-1] :

        dist_meas = friis_distances(drone, anchors)
        c, drone_e = guess_position(drone, anchors,
                        dist_meas, False)

        drone[0] = x_pos[p]
        drone[1] = y_pos[p]
        drone[2] = z_pos[p]

        if printing:
                ax.scatter(drone[0], drone[1], drone[2],
                        alpha=0.4,  marker='^', c='b')
                ax.scatter(drone_e[0], drone_e[1], drone_e[2],
                        alpha=0.8, marker='h', c='r')
            p += 1
```

```
    p2 = 0
    while drone[2] > anchor_center[2] and drone[2] != 0:
    dist_meas = friis_distances(drone, anchors)
    c, drone_e = guess_position(drone, anchors, dist_meas,
                        False)

    if (drone[2] - 0.2) > 0:
        drone[2] -= 0.2
    else:
        drone[2] = 0

    if printing:
        ax.scatter(drone[0], drone[1], drone[2],
            marker='^', c='b')
        ax.scatter(drone_e[0], drone_e[1], drone_e[2],
            alpha=0.4, marker='h', c='r')
    p2 += 1

if printing:
    plt.xlabel("x␣axis␣[m]")
    plt.ylabel("y␣axis␣[m]")
    plt.title("UAV␣trajectory␣-␣Spacing␣1␣m")
    plt.grid()
    plt.legend(loc=4)

plt.show()
```