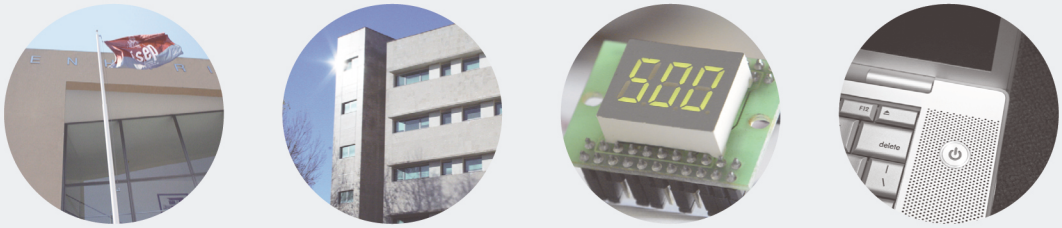
INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO
MESTRADO EM ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES



Modelação e Simulação de um Veleiro Terrestre com Vela Asa

VÍTOR DANIEL VELOSO TINOCO
outubro de 2020

POLITÉCNICO
DO PORTO

Instituto Superior de Engenharia do Porto
Departamento de Engenharia Electrotécnica
Rua Dr. António Bernardino de Almeida, 431, 4249-015 Porto

Modelling and Simulation of a Wing Sail Land Yacht

Mestrado em Engenharia Eletrotécnica e de Computadores - Sistemas Autónomos

Vítor Daniel Veloso Tinoco

Supervision:
Prof. Maria Benedita Campos Neves Malheiro
Prof. Manuel Fernando Santos Silva

Ano Letivo: 2019-2020

Resumo

A maioria dos veículos autónomos utilizam baterias elétricas ou combustível como fonte de propulsão e alimentação. Contudo, estas fontes energéticas são dispendiosas, poluentes e não sustentáveis. O vento constitui uma fonte propulsora natural, gratuita e sustentável, o que pode, em certas situações, substituir motores eléctricos e de explosão. O objetivo desta dissertação é desenhar uma vela de asa rígida, de rotação livre, e integrá-la num veículo terrestre autónomo com um sistema de controlo baseado em Robot Operating System (ROS). O estudo inicial deste trabalho abrangeu: (*i*) veleiros terrestres tripulados e não tripulados e as suas respetivas competições; (*ii*) perfis alares do National Advisory Committee for Aeronautics (NACA); e (*iii*) diferentes tipos de materiais e componentes eletrónicos usados em veleiros terrestres autónomos. De seguida, desenhou-se uma plataforma composta por um chassis de quatro rodas, uma vela asa com cauda (ambas com um perfil alar NACA63(3)-018), um sistema de controlo de cauda (que usa o ângulo de ataque atual e desejado como entradas), e um sistema de controlo da direção (que usa a posição atual e desejada como entradas). Na etapa seguinte, procedeu-se à simulação da plataforma desenhada no ambiente de simulação Gazebo. Para isso, foi criado um modelo do veleiro terrestre, usando o software de Computer-Aided Design (CAD) Fusion 360, e foi importado para o Gazebo. Posteriormente, foram adicionados vários *plugins* ao ambiente de simulação, nomeadamente *plugins* importados, alterados e desenvolvidos de raiz. O sistema de controlo foi desenvolvido em ambiente ROS, usando um controlador Proportional - Differential (PD) para a cauda da asa e um controlador Proportional (P) para a direção da plataforma. Finalmente, foram realizados testes de controlo da vela asa, para validar o controlo via cauda e a propulsão do veículo, e testes de controlo da direção, para validar o mecanismo de direção e a capacidade de seguimento de rotas. Os resultados demonstram o correto funcionamento dos sistemas de simulação e controlo.

Palavras Chave

Veleiro terrestre, velejar em terra, vela de asa rígida, vela de asa rígida com rotação livre.

Abstract

The majority of autonomous vehicles use electric batteries or fossil fuels for propulsion and power. However, these energy sources are costly, pollutant and unsustainable. The wind is a natural, free and sustainable propellant and energy source and can, in certain situations, replace electric and combustion engines. The objective of this dissertation is to design a free rotating rigid wing sail and integrate it in an autonomous land vehicle with a control system based on Robot Operating System (ROS). The initial study of this work covered: (*i*) manned and unmanned land yachts as well as the respective competitions; (*ii*) National Advisory Committee for Aeronautics (NACA) airfoils; and (*iii*) different types of materials and electronic components used in autonomous land yachts. Subsequently, a land yacht platform, consisting of a four-wheeled chassis, a wing sail with a flap tail (both with NACA63(3)-018 airfoils), a flap control system (that uses the current and desired angle of attack as inputs), and a steering control system (which uses the current and desired position as inputs), was designed. Then, the designed platform was simulated in the Gazebo simulation environment. For this, the land yacht model was created, using the Fusion 360 CAD software, and imported into Gazebo. Next, several plugins were added to the simulation environment, ranging from imported to changed and newly developed plugins. The control system was developed in ROS, using a Proportional - Differential (PD) controller for the wing flap and a Proportional (P) controller for the platform direction. Finally, wing control tests were carried out to validate the flap control and vehicle propulsion, and steering control tests to validate the steering mechanism and the ability to follow routes. The results demonstrate the correct functioning of the simulation and control systems.

Keywords

Land yacht, land sailing, rigid-wing sail, free rotating rigid-wing sail.

Contents

List of Figures

List of Tables

List of Equations

List of Listings

Glossary

ABS Acrylonitrile Butadiene Styrene. 13

AGV Autonomous Ground Vehicles. 1, 2

ASV Autonomous Surface Vehicles. 1

AUV Autonomous Underwater Vehicles. 1

CAD Computer-Aided Design. iii, 41

CAN Controller Area Network. 22

CPU Central Processing Unit. 26

D-GPS Differential GPS. 16

DART Dynamic Animation and Robotics Toolkit. 40

EPS European Project Semester. 7, 9

FISLY International Land and Sandyachting Federation. 5, 6

GLONASS Global Navigation Satellite System. 16

GNSS Global Navigation Satellite System. ii, vii, 15, 16, 18, 25, 29, 36, 40, 45

GPS Global Positioning System. 12, 13, 16, 24, 25, 40

I^2C Inter-Integrated Circuit. 22

I/O Input and Output. 22, 26

IMU Inertial Measurement Unit. ii, 12, 13, 15, 16, 25, 36, 45

ISEP Instituto Superior de Engenharia do Porto. xv, 7, 9

LiDAR Light Detection And Ranging. 40, 41

LiPo Lithium-Polymer. 30

MOAST Mobility Open Architecture Simulation and Tools. 41

MORSE Modular Open Robots Simulation Engine. iii, 39, 40

NACA National Advisory Committee for Aeronautics. iii, v, 7

NALSA North American Landsailing Association. 5, 6

NASA National Aeronautics and Space Administration. 3

ODE Open Dynamics Engine. 39, 40

OGRE Object-Oriented Graphics Rendering Engine. 40

OSRF Open Source Robotics Foundation. 40

P Proportional. iii, v, 46, 47, 49, 52, 63

PD Proportional - Differential. iii, v, 46, 48, 52, 56, 58, 60, 61, 63, 64

PWM Pulse Width Modulation. 35, 36

RAM Random-Access Memory. 26

RF Radio Frequency. 12, 13, 22

ROS Robot Operating System. iii–v, 2, 22, 26, 27, 39–41, 43–48, 51, 54, 56, 61, 65, 75–77

RPM Revolutions per Minute. 20, 21

SBAS Satellite-Based Augmentation System. 16, 18

SBC Single-Board Computer. 22, 26, 30

SDF Spatial Data File. 41, 42

SPI Serial Peripheral Interface. 22

TTL Transistor-Transistor Logic. 24

UART Universal Asynchronous Receiver-Transmitter. 22

UAV Unmanned Aerial Vehicles. 1

USARSim Unified System for Automation and Robot Simulation. iii, 39, 41

XML Extensible Markup Language. 41, 42

Acknowledgements

I would like to express my gratitude to Prof. Benedita Malheiro and Prof. Manuel Silva for introducing and giving me the opportunity to develop this thesis. Furthermore, I would like to thank these Professors for all the guidance and tutoring done throughout the development of this dissertation.

I would also like to thank all of my Professors that accompanied my journey as a student in Instituto Superior de Engenharia do Porto (ISEP).

Finally, I would like to thank my friends and family for all the support that was given to me during all of this time.

Chapter 1

Introduction

Most autonomous vehicles rely on electric batteries or fuel to achieve propulsion [9]. However, there are certain instances where an alternative, such as wind power, can be used to replace the conventional means of propulsion. This enhances these vehicle's utility in scenarios where it is awkward or even impossible to refuel or recharge batteries.

1.1 Problem

The majority of autonomous vehicles use electric batteries or fuel as a power source [9]. This is ideal for Unmanned Aerial Vehicles (UAV) and Autonomous Underwater Vehicles (AUV), for example, as these vehicles are required to operate in a space where it is not ideal to use cables to power them. However, the use of electric motors, and other power electronics, are expensive in terms of energy consumption [9]. In some autonomous vehicles, such as Autonomous Surface Vehicles (ASV) and Autonomous Ground Vehicles (AGV), it is possible to lower the energy consumption by replacing its motors with a wind powered sail [9][17]. The use of a sail opens the possibility of long term missions for AGV on harsh environments with constant high wind speeds, such as Antarctica [9][17], where it is not ideal, and some times impossible, to recharge the batteries of the vehicles on a daily basis. Although the usage of a sail on land vehicles is a good option in saving battery charge, the control and the aerodynamics of a conventional cloth sail are not efficient [9][17]. The control of a conventional sail requires actuators on several ropes, making this a hard and complex task. Furthermore, the conventional cloth sail does not distribute the wind pressure uniformly [9][17]. In order to conduct surveys, data logging, zone monitoring or other scientific projects in remote areas (where there is little to no human presence) with an AGV and since solar energy is, per se, insufficient to maintain these vehicles fully operational, an energy efficient vehicle that does not rely on electrical power or fuel for propulsion is required. This dissertation aims to improve the research for the development of a solution to this problem.

The use of a rigid-wing sail has been proposed in the literature [9][17] as an efficient alternative for conventional cloth sails. These rigid-wing sails work the same way as a wing from an aeroplane, with similar aerodynamic forces, and distribute the wind pressure uniformly. Furthermore, they only require one actuator for control purposes.

1.2 Motivation

The main motivation of this dissertation is to explore a more sustainable way of powering an AGV with the use of wind power. A wind powered AGV, or autonomous land yacht, can be highly recommended for missions in remote areas, specifically those conducted on vast flat surfaces, such as on Antarctica, allowing to significantly reduce the downtime and human maintenance usually required when using other power sources.

As for the author's personal motivation, the use of renewable energy and sustainability driven design are particularly important nowadays, given the current human carbon footprint. Consequently, the author wishes to contribute to the adoption of renewable energies, which, in this case, materialises in the design and simulation of a land yacht.

1.3 Objectives

This dissertation has the following main objectives:

- Modelling of a rigid-wing sail;
- Creation of an autonomous land vehicle;
- Integration of the sail on an autonomous land vehicle.

Moreover, the dissertation has the following objectives:

- Development of a motion control system;
- Development of a free rotating wing sail control system.

1.3.1 Requirements

As for requirements, this dissertation has the following:

- Development using the Robot Operating System (ROS);
- The vehicle model must be robust to avoid capsizing;
- The vehicle's dimensions must not exceed 1 m of length, 0.5 m of width and 1.5 m of height;
- The vehicle must not exceed a total mass of 20 kg;
- The vehicle must have an autonomy of at least two hours;
- The vehicle must be energy efficient;
- The vehicle must have a manual control that overrides the autonomous control.

1.4 Use Cases

For the use cases, the user can choose:

- Autonomous drive - Set a trajectory, or a path, and the vehicle will navigate autonomously, *i.e.*, will adjust automatically the angle of attack of the wing sail and the steering mechanism, according to the direction and velocity of the wind, to reach the destination;
- Manual drive - Control manually the steering and/or position the wing sail.

1.5 Functionalities

There are several functionalities for wind powered land vehicles and for rigid-wing sails as well.

1.5.1 Wind Powered Vehicles

The vehicles can serve various purposes, including racing & competition and scientific research. As for racing, there are various competitions worldwide for land yachts, mainly composed of three-wheeled manned sail vehicles. For scientific research, these vehicles can be used in polar regions where the vast majority of its area is flat and there is constant wind to propel the vehicle. Moreover, these vehicles can be used for planetary research in planets such as Venus or Mars. These planets have an atmosphere which means that hybrid wind and electrical powered vehicles are a possibility. A wind powered rover for Venus was proposed by the National Aeronautics and Space Administration (NASA) since the planet's thick atmosphere makes it impossible to collect solar energy [7].

1.5.2 Rigid-Wing Sails

As for the rigid-wing sail functionalities, modelling an efficient wing sail can be advantageous. For example, they can be used to increase power generation in wind turbines as the dimensions and format of their blades influence the aerodynamic performance needed to efficiently produce electrical power [18]. Furthermore, rigid-wing sails are mostly used in autonomous sailboats, making them suitable for long-term ocean sampling, monitoring and surveillance. The main reason is that wing sails provide a robust and simple control, comparatively to conventional cloth sails, and typically do not require human interaction to complete a given mission [19].

1.6 Structure of the Dissertation

This dissertation is composed of the following chapters:

Chapter 1 **Introduction**;

Chapter 2 **Land Sailing** reviews existing manned and unmanned land yachts, land yacht competitions, NACA airfoils, materials and electrical components used in land yachts;

Chapter 3 **Platform Design** describes a proposed platform, a wing sail model as well as a proposed control system for the wing flap and for the steering;

Chapter 4 **Platform Simulation** outlines the development of the proposed platform as well as the control systems in a simulated environment;

Chapter 5 **Tests, Results and Validation** reports the results and validation of a series of experiments performed for the wing sail control, the propulsion and the steering control;

Chapter 6 **Conclusion** presents the outcomes of the dissertation, the future work, and the conclusion to the dissertation.

Chapter 2

Land Sailing

Land sailing is the action of moving a ground vehicle through wind propulsion with the use of a sail [20]. These vehicles are called land yachts and function similarly as a sailboat except, instead of being steered using a rudder, the land sailors (pilots) steer the land yacht by turning the wheels, and have access to brakes [20]. In this Chapter, the state of the art on land yachts is presented. This includes a listing of competitions, manned and autonomous platforms, some autonomous water platform examples and a comparative analysis of the modules found in these vehicles.

2.1 Land Yacht Competitions

Land yacht competitors are present all over the planet with competitions happening every week [21]. At the time of the writing of this document, the main land yacht competitions are organised by the International Land and Sandyachting Federation (FISLY), the Blokart International Ltd and by the North American Landsailing Association (NALSA) [21].

2.1.1 The International Land and Sandyachting Federation

FISLY, founded in 1963, is a non-profit organisation from Belgium whose main purpose is to promote sand and land yachting in a safe environment [22]. By doing so, FISLY encourages: the exchange of research and development data of land yachts, the cooperation between international organisations with shared goals, the creation of new organisations on areas where they do not yet exist and other activities relevant to land yachting [22].

FISLY also promotes and organises various land and sand yacht competitions from all over the globe [22]. It is responsible for the organisation of the Euro Cup, between European countries, and the World Championship, at the international level. The Euro Cup competition happens annually while the World Championship every four years. On a World Championship year there is no Euro Cup competition [21][22].

2.1.2 Blokart World Championships

Blokart World Championship is a land yacht racing competition organised by Blokart International Ltd [21][2]. It happens every two years in a different country and each contestant must use a Blokart vehicle [2]. Blokart has a single build worldwide. This one design racing standard ensures fairness for competitors [2].

2.1.3 North America's Land Sailing Cup

The North America's Land Sailing Cup is the largest landsailing event in the United States [23]. It is organised by NALSA and takes place annually on the Ivanpah Dry Lake in California [21][23].

2.2 Land Yacht Classes

FISLY has divided the various land yacht types into different classes [22]. This is important so that different types of vehicles do not compete directly [22].

2.2.1 Classes

The land yacht classes are [22][24][25]:

- **Class 2** – This class contains the largest land yachts with masts up to 8 m. These vehicles can have a maximum width (fully rigged with a pilot in the cockpit) of 3.65 m and a minimum body length of 4.15 m. The sail area cannot exceed 11.3 m^2. As for the mast, it must contain a fluorescent orange strip, 40 mm wide and at least 2 m long, on its leading edge;
- **Class 3** – This class holds the fastest land yachts with a maximum sail area of 7.5 m^2 and a rotating aerofoil section wing mast. The mast is limited to 6.10 m with a fluorescent orange strip, 40 mm wide and at least 2 m long, on its leading edge. These vehicles can have a maximum width (fully rigged with a pilot in the cockpit) of 3.5 m, and a maximum distance between the front wheel and the middle of the rear wheels of 3.8 m. Its maximum weight is limited to 100 kg;
- **Class 5** – The vehicles in this class must have a minimum weight of 50 kg. The width of its chassis must not exceed 2 m, and the distance between front and rear wheels must not exceed 2.5 m. The maximum diameter of the wheels is 750 mm. As for the mast, its distance to the ground must not exceed 5.5 m. The maximum sail area for this class is 5.5 m^2;
- **Class Standard** – These vehicles have a sail area of 5.8 m^2. The chassis maximum width is 2.64 m and its maximum length is 4.06 m, and the mast has a maximum size of 5.45 m;
- **Class 7** – Any land yacht which is sailed by a pilot standing upright, the rig of which will not remain upright unless the pilot is holding it, is considered class 7;
- **Class 8** – The class 8 consists of land vehicles controlled by a kite, also called Parakarts. The pilot must be sitting or lying on the vehicle to steer it. The brake

system must be a wind brake done by the kite. The maximum width and length is 3 m and 3.5 m, respectively. The wheels have a maximum diameter of 0.6858 m and the vehicles total weight must not exceed 60 kg;

- **Class Mini-Yacht** – The mini-yacht is considered a "first yacht" for beginners. The vehicle must be small enough to fit inside a continuous loop of rope 5.60 m long, and have a free rotating sail.

2.3 NACA Airfoils

NACA airfoils are aircraft wing profiles developed by the National Advisory Committee for Aeronautics (NACA) [1]. This type of airfoil design can be divided into four to eight digit series. In the four-digit series the first digit refers to the camber percentage, the second digit refers to the distance, in percentage, from the leading edge of the wing, to the maximum camber and the last two digits refer to the maximum thickness of the chord [1]. For example, the NACA0018 airfoil has a maximum camber of 0 % and a maximum chord thickness of 18 %. In [1] there are several pre-made NACA airfoil models with the lift to drag ratio for a certain angle of attack; or the drag coefficient given a lift coefficient. Some NACA airfoil designs are presented in Figure 2.1.

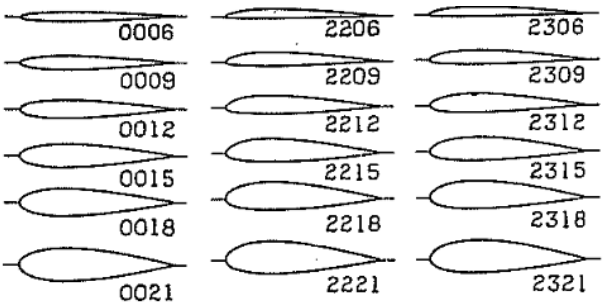


Figure 2.1: Some Examples of NACA Airfoils. Adapted from [1]

2.4 Manned Platforms

In the past, land yachts were used for transport or to have fun [4], but from 1950 they have been mostly used in racing. For this purpose, these vehicles are required to be manned. Some examples of manned land yachts are: the Blokart model, the Greenbird, the Pacific Magic, the Whike and the Sail Car by European Project Semester (EPS) at ISEP.

2.4.1 Blokart

The Blokart model is a generic land yacht model used in competitions [2]. The model is a three wheeled vehicle with a single sail. The chassis is composed either of stainless steel or Chromium Molybdenum steel. As for the sail, it is made from dacron fabric and there are 4 variants: the 5.5 m Extra LITE, the 4 m LITE, the 3 m Standard and the 2 m Storm. The Blokart vehicle model is presented in Figure 2.2.

Figure 2.2: Blokart Vehicle Model [2]

(a) Greenbird Land

(b) Greenbird Ice

Figure 2.3: Greenbird [3]

2.4.2 Greenbird

The Greenbird is a land yacht and an ice yacht [3]. This vehicle was designed with the objective to break both the land and ice world speed records set by wind driven vehicles. The land yacht uses a rigid-wing sail and achieved a land speed of 203 km/h with 40 km/h winds [3]. As for the ice variant, it achieved a speed of 135 km/h [3]. The land and ice versions of Greenbird are presented in Figures 2.3a and 2.3b respectively.

2.4.3 Pacific Magic

The Pacific Magic is a class 5 land yacht. It is a cheap and light weight vehicle that uses wheelbarrow wheels [4]. Its purpose was to stop its creators from falling into a future of very expensive class 5 land yachts [26].

Figure 2.4: Pacific Magic Class 5 Land Yacht [4]

2.4.4 Whike

Whike is a Dutch company that produces pedal and wind-powered vehicles. This company produces a lightweight tricycle with a sail, pedals and gears. The vehicle comes with two types of polyester film sails with $1\,m^2$ and $1.6\,m^2$ of sail area. Finally, the mast is made out of carbon fibre [5]. Figure 2.5 displays a Whike vehicle.

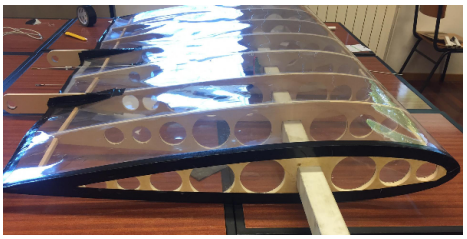
Figure 2.5: Whike Tricycle [5]

2.4.5 Sail Car by The European Project Semester (EPS) at ISEP

The Sail Car is a manned land yacht with a rigid-wing sail developed at ISEP by the team of the European Project Semester (EPS) [6]. Its prime objective is to be both

sustainable/ethic and efficient. The frame is made out of steel S235 and has a length and width of 2 m and 1.55 m, respectively. The wing sail has a NACA0015 airfoil and is composed by the rigid-wing sail and the tail. It is made out of plywood and has a height of 2.5 m, a length of 0.5 m and an area of 1.25 m^2. Although the vehicle is manned, the control of the wing sail is autonomous. The sail has a servo motor that turns the tail, since this component is much smaller and lighter than the rest of the wing, reducing the power needed to control it. As the tail adjusts to the wind, the rest of the sail adjusts automatically. The control system uses an Arduino Uno as microcontroller, a servo motor and a rotary encoder to measure the wing rotation [27]. The Sail Car frame and sail are shown in Figures 2.6a and 2.6b, respectively.

(a) Sail Car Frame



(b) Sail Car Wing Sail

Figure 2.6: Sail Car Frame & Wing Sail [6]

2.5 Autonomous Platforms

Although autonomous land yachts have not been as studied as manned land yachts, some research has been done [9]. This sections presents a number of autonomous land yacht prototypes developed by the scientific community.

2.5.1 Zephyr

Zephyr is a concept for a wind powered rover to be used in Venus [7] and is represented in Figure 2.7. Unlike the previous platforms, this vehicle is autonomous. While providing propulsive force, its large sail also serves for mounting a solar array. Moreover, the wheels are large to reduce the force to climb rocks, and the distance between wheels is wide to increase stability [7].

2.5.2 Mad Mads 1

Mad Mads 1 is a prototype, presented in [8], of an autonomous land yacht. It uses a conventional cloth sail, and it uses pedals and a rope attached to the sail as control

Figure 2.7: Zephyr Concept Model [7]

inputs. Its purpose is to study a control strategy for steering a land yacht in a tracking operation [8]. A picture of the Mad Mads 1 is shown in Figure 2.8

Figure 2.8: Mad Mads 1 [8]

2.5.3 Autonomous Controlled Four Wheeled Land Yacht

The Autonomous Controlled Four Wheeled Land Yacht is a prototype presented in [9]. This vehicle, shown in Figure 2.9, only uses wind power as propulsion. It is composed by: an aluminium framed hull, a rigid-wing sail, two rear and two front wheels, a steering mechanism, an axle, two wire ropes, a servomotor and a steering engine. The aluminium frame weighs 19 kg and its shape ensures low thrust from the wind hitting the hull. The wing sail is 1 m tall, 2 mm thick and weighs 2.15 kg with an area of 0.25 m^2. It is connected to the output shaft of the servo motor and can rotate from 0° to 360°. Moreover, the wing sail aerodynamic characteristics are based on the NACA0018 airfoil design.

The two frontal wheels are part of the steering mechanism, while the two back wheels

are connected by a long axle. This triangular chassis improves stability, and the four wheels grant better manoeuvrability. In [9], the author shows that the lift and drag coefficients become null for a symmetrical wing sail with an angle of attack of 0°, stopping the propulsion. This makes the symmetrical wing sail well suited for harsh environments.

The control system utilises a Global Positioning System (GPS) unit and an Inertial Measurement Unit (IMU) sensor to calculate the running distance, and a timer to accumulate the time used to calculate the average speed. Given a trajectory, the position error is calculated and sent to an A3C6410 non-linear fuzzy controller, and an STM32 microcontroller, via an nRF24L01 Radio Frequency (RF) chip, to control the steering mechanism. As for the wing sail rotation, the control system utilises an anemometer to measure the relative wind velocity and a dogvane to measure the wind direction, which is used to actuate a servo motor, turning the wing sail.

Figure 2.9: Autonomous Controlled Four Wheeled Land Yacht [9]

2.5.4 Multiple Wing Sail Land Yacht Robot

A multiple wing sail land yacht is presented in [10]. The mechanical structure is similar to the autonomous controlled four wheeled land yacht, presented in the previous subsection. Its body is made of a light aluminium frame to minimise the weight, and has a small volume, reducing the thrust generated by the wind. The wheel distribution and steering mechanism are also identical. The main difference between this and the previous models is that this one uses multiple wing sails. The wing sails aerodynamic characteristics are also based on the NACA0018 airfoil and can rotate from 0° to 360°.

Each wing sail has a height of 1 m, a thickness of 2 mm, an area of 0.25 m^2 and weighs 2.15 kg. According to the authors [10], the usage of multiple wing sails increases the lift force of the vehicle, as it has more effective sail area at the expense of having a more complex control system. With an angle of attack of 90° and 3 wing sails, the vehicle starts its movement with a wind velocity of 3 m/s. The multiple wing sail land yacht is presented in Figure 2.10.

The control system adjusts the wing sail rotation and the steering gear. For the wing sail control, the system uses an anemometer and a dogvane to obtain the relative wind velocity and wind direction, and actuates a servo motor to turn the wing sail. The steering control uses a navigation system that compares the current pose with the trajectory. The navigation system uses a GPS and a xsens MTi IMU. All the sensors referred are connected to an nRF24L01 RF chip that communicates with the S3C6410 controller chip and the STM32 microcontroller.

Figure 2.10: Multiple Wing Sail Land Yacht [10]

2.5.5 Wing Sail Land Yacht

In [11], Dong *et al.* present a wing sail land yacht. This vehicle mainly consists of two parts: a four wheeled body and a wing sail. The body is composed of a steel and plastic frame 35 cm long, 20 cm wide and 10 cm high. The wheels are made out of solid rubber slicks and the front wheels have a steering angle ranging from $-60°$ to $+60°$ with the use of a servo motor. The front wheel base is 18 cm and the rear wheelbase is 25 cm. The wing sail is a NACA0015 airfoil with 80 cm wingspan and 30 cm chord and is made of Styrofoam with smooth plastic film on the outside. The wing sail has an angle range of $0°$ to $270°$. The control system uses a CALYPSO wind sensor and a HI219 IMU in order to obtain the wind direction and the vehicles heading, respectively. For controlling the vehicle, a Raspberry PI is used as the on board computer. The presented land yacht is shown in Figure 2.11

2.5.6 Optimizing the Driving Forces Threshold

Mirzaei and Rad designed a land yacht prototype to reduce the minimum forces required to start the vehicle movement (driving forces) [12]. Although the authors do not present the control system, they describe some of its mechanical characteristics. It is composed of Acrylonitrile Butadiene Styrene (ABS) copolymer and a NACA0012 airfoil wing sail. The body has a triangular shape with two rear wheels and one front wheel. Its length is 109.1 cm and its width is 142 cm. The wing sail is composed of balsa wood coated with fibreglass strings, and has a wingspan of 100 cm with a 50 cm chord. The vehicle has a total mass of 8 kg. This vehicle can achieve an acceleration of $3.9\,\mathrm{m\,s^{-2}}$ with a relative

Figure 2.11: Dong *et al.* Land Yacht Prototype [11]

wind velocity and wind direction of 20.1 m/s and 0° respectively, and an angle of attack of 20°. The described land yacht prototype is presented in Figure 2.12.

Figure 2.12: Mirzaei and Rad Land Yacht Prototype [12]

2.5.7 Velocity of a Land Yacht with Rigid and Cloth Sails

Khayyat and Rad present in [28] a comparative study of the velocity achieved by a land yacht equipped with rigid-wing sail or a cloth sail. With the use of wind tunnel simulations and velocity prediction programs, the authors determined the efficiency of both types of sail, given their dynamic characteristics. They concluded that, for a land yacht, rigid-wing sails are more efficient than regular cloth sails.

2.6 Autonomous Water Platforms

As mentioned previously, there is scarce research done on autonomous land yachts. However, further inquiry shows that the research on autonomous rigid-wing sail boats, whose sails are identical to the rigid-wing sails on land yachts, is popular.

Several rigid-wing sail boats utilise a free-rotating rig system, such as the Maribot Vane [13] or the ASPire [29][30]. The free-rotating system, as the name suggests, allows the wing sail to rotate freely without an actuator. It uses a flap to define the angle of attack of the main sail. An actuator on the flap will require much less energy than one for the whole wing, meaning this is a more energy efficient way of controlling the vehicle [13][29][30]. A free-rotating rigid-wing sail is presented in Figure 2.13.

Figure 2.13: Maribot Vane [13]

2.7 Material and Electrical Component Survey

In this section, possible types of materials and electrical components, such as an Inertial Measurement Unit (IMU) or a Global Navigation Satellite System (GNSS) unit, that can be used to create an autonomous land yacht, are going to be characterised.

2.7.1 Frame Materials

The selection of the materials to build the vehicle frame is important, as it has a direct impact on the weight, resistance and price of the vehicle. In this subsection, stainless steel and aluminium are presented as valid building materials for the land yacht chassis.

- **Stainless Steel** – Stainless steel is an alloy with high percentages of iron, chromium and nickel with a density of $8000\,\mathrm{kg/m^3}$. It is resistant to corrosion, fire and heat, and maintains its strength at high temperatures. Furthermore, stainless steel has a long life cycle and a high strength to weight ration, which reduces the necessary material thickness [31][32].

- **Aluminium** – Aluminium is a light metal with a density of $2700\,\mathrm{kg/m^3}$ and strength to weight superior to steel [33]. Pure aluminium is a soft metal that,

when alloyed with other metals, such as manganese, silicon, copper, and magnesium, becomes a stronger material. Furthermore, this metal is cheap, corrosion resistant, can be recycled several times [33], and it is a very good building material as it can also be reshaped to meet the requirements [34].

2.7.2 Sensors

The development of a land yacht requires a selection of sensors to perceive the environment, such as IMU and GNSS devices that output sensor information on the pose and motion of the vehicle. Furthermore, the development of an autonomous land yacht requires logic units, to process the sensor information and command actuators, such as servo motors, to act on the environment. In this subsection, a selection of different sensors, actuators and computer boards, that can be used in a land yacht, are studied.

2.7.2.1 IMU

An IMU is a device that contains: an accelerometer, a gyroscope and/or a magnetometer [35]. It outputs a body's acceleration, angular velocity and orientation (if it contains a magnetometer) and is used in several navigation applications [36][35]. With the use of a GPS, it can accurately determine a vehicle's pose [36].

Currently, there are various high grade IMU modules designed specifically for robotic navigation, such as the Elipse 2 Micro Series from SBG Systems (1100 € to 3500 €) [37], the MTi 600 Series from XSENS (400 € to 1000 €) [38], the NavChip from Intersense (400 € to 1000 €) [39] or the 1750 KVH IMU (400 € to 800 €) [40]. There are, however, several consumer grade IMU chips on the market with affordable prices. These include the low power LSM6DSOX from ST (4 €) [41], the MPU-6000 from TDK (10 €) [42], or the BMF055 from Bosh Sensortec (10 €) [43]. The presented IMU characteristics are shown in Table 2.1.

2.7.2.2 GNSS Receiver

GNSS is the standard term for global navigation satellite systems. These worldwide coverage systems include: GPS, Galileo, Global Navigation Satellite System (GLONASS) and BeiDou [44][45]. A GNSS receiver relies on information received from several satellites to determine its position, velocity and time. The more satellite signals captured by the receiver, the better will be the positioning accuracy [44][45]. This position accuracy can be further augmented using a Satellite-Based Augmentation System (SBAS), which uses additional regional GNSS augmentation data, or using Differential GPS (D-GPS) that uses known fixed positions to correct the positioning error.

Some GNSS units are: the NEO-M8 by u-blox (30 € to 60 €) [46], the Ultimate GPS (board) by Adafruit (30 € to 40 €) [47], and the GM Series by Linx Technologies (board) (50 € to 70 €) [48]. The main features of these GNSS units are presented in Table 2.2.

2.7.2.3 Rotary Encoder

A rotary encoder is a sensor that converts angular motion, or position, information into analog or digital values [49]. This sensor can be incremental or absolute. The incremental encoder needs all previous readings in order to determine the angular motion

Table 2.1: Characteristics of the Presented Inertial Measurement Units

	Elipse 2 Micro	**MTi 600**	**NavChip**	**1750 KVH**	**LSM6DSOX**	**MPU-6000**	**BMF055**
Accel. Range ($\tilde{g}$)	± 16	± 10	± 16	± 10	± 16	±16	± 16
Accel. Bias ($m\tilde{g}$)	± 5	± 0.01	± 0.006	±7.5	± 20	±50	±80
Velocity Random Walk ($m/s/\sqrt{h}$)	0.034	0.035	0.05	0.007	0.06	0.24	0.09
Gyro. Range (°/s)	± 450	± 2000	±2000	± 490	±2000	±2000	±2000
Gyro. Bias (°/s)	± 0.2	±0.002	±0.001	±0.0006	±1	±20	±1
Angular Random Walk ($°/s/\sqrt{h}$)	0.15	0.42	0.3	0.012	0.23	0.3	0.84
Input Voltage (V)	4 - 15	4.5 - 24	3.25–5.5	9–36	1.71–3.6	2.375–3.46	2.4–3.6
Power Consumption (mW)	400	310–530	135	5000–8000	1000	12.5	50
W×L×H (mm^3)	26.8×18.8×9.5	31.5×28×13	12.5×24.5×5.4	70×61×73	25×3×0.83	4×4×0.9	1.1×5.2×3.8
Magnetometer	Yes	Yes	Yes	n/a	No	No	Yes
Barometer	No	Yes	No	No	No	No	No
Output Interface	RS-232, RS-422, USB, CAN	XBus, CAN	UART, SPI	RS-422	SPI, I^2C	SPI, I^2C	SPI
Weight (g)	10	8.9	3	700	n/a	n/a	n/a
Price (€)	1100–3500	400–1000	400–1000	4000–8000	3–5	6–10	6–10

n/a - not available

Table 2.2: Characteristics of the Presented GNSS Units

	NEO-M8	**Ultimate GPS**	**GM Series**
Position Accuracy (m)	2	1.8	2.5
Receiver Channels	72	66	99
Receiver Type	GPS GLONASS BeiDou Galileo	GPS	GPS GLONASS BeiDou Galileo QZSS
SBAS	WAAS EGNOS MSAS	WAAS EGNOS	WAAS
DGPS	Yes	Yes	Yes
Sensitivity (dBm)	-167	-165	-161
Protocols	NMEA UBX RTCM	NMEA	NMEA
Update Rate (Hz)	10	10	10
Input Voltage (V)	1.65 - 3.6	3.0 - 5.5	3 - 4.3
Power Consumption (mW)	63	100	53
W×L×H (mm^3)	12.2×16.0×2.4	25.5×35.0×6.5	28×28.0×7.0
Output Interface	UART USB SPI I^2C	UART USB	UART
Weight (g)	1.6	8.5	n/a
Price (€)	30 - 60	30 - 40	50 - 70

n/a - not available

or position. On the other hand, the absolute encoder only needs the current reading in order to determine the angular position [49]. In the case of a land yacht, an absolute encoder can be utilised to determine both the wing sail and the steering wheel orientation. Some examples of absolute rotary encoders are the AMT21 by CUIdevices (40 € to 50 €) [50] or the EAW - Absolute Contacting Encoder by BOURNS (4 € to 10 €) [51]. The characteristics of these encoders are presented in Table 2.3. A potentiometer can be used for the same purpose.

2.7.2.4 Wind Sensor

Given that a land yacht relies on the wind for propulsion, a sensor that measures the wind velocity and wind direction is necessary. The FST-200-201 is a 3-cup anemometer that measures wind speeds from less than 0.5 m/s to 50 m/s [52]. It is made of metal alloys, preventing oxidation, has an input voltage of 12 V to 36 V, and comes in two versions: the FST-200-201-I with an output signal of 0 mA to 20 mA and the FST-200-201-U with an output signal of 0 V to 12 V. Furthermore, this sensor uses an M12 connector and has a diameter of 131.5 mm and 127.5 mm of height. Its price range is 100 € to 200 € [52]. From the same company, there is the FST-200-202 wind direction sensor [53]. This sensor measures the direction of the wind with a resolution of 22.5° with a ± 3° accuracy. Furthermore, its readings range from 0° to 360° with a minimum wind velocity of 0.8 m/s and a maximum wind velocity of 70 m/s. The input voltage ranges from 12 V to 36 V with

Table 2.3: Characteristics of the Presented Rotary Encoders

	AMT21	**EAW**
Resolution (bit)	12/14	8
Accuracy (°)	0.2	2
Turn	Single / Multi	n/a
Shaft Diameter (mm)	9.0	6.3
Input Voltage (V)	3.8 - 5.5	10
Power Consumption (mW)	80	100
W×L×H (mm^3)	28.6×39.9×11.7	23.9×27.9×27.6
Output Interface	RS-485	Binary Value
Weight (g)	15.7	14
Price (€)	40 – 50	4 – 10

n/a - not available

an electrical resistance of 250 Ω. Finally, this sensor has 157.5 mm of height, 234.0 mm of diameter, and the same output signal (0 V to 12 V or 0 mA to 20 mA), connector (M12), and price range (100 € to 200 €) as the previous sensor [53].

Although the wind sensors referred sense wind velocity, and wind direction, individually, there are wind sensor modules which are composed by both a wind velocity sensor and a wind direction sensor. An example of this type of module is the Wind Monitor Model 05103 by YOUNG [54]. Starting with the wind velocity sensor, the Wind Monitor Model 05103 has a measurement range of 0 m/s to 100 m/s with a ± 0.3 m/s accuracy, and a 1.0 m/s minimum wind velocity threshold. The wind direction sensor of the module has a measurement range of 0° to 360° with an accuracy of ± 3.0° and a 1.1 m/s minimum wind velocity threshold. The module has an input voltage of 8 V to 24 V and has a power consumption of 5 mA at 12 V. It has a height of 370.0 mm, a diameter of 550.0 mm, and weighs 1.0 kg. This module has 2 versions: one has a voltage output of 0 V to 5 V and the other version has a current output of 4 mA to 20 mA. Finally, its price range is 1000 € to 1400 € [54]. Another example of a wind sensor is the WM30 module from Vaisala [55]. This module is composed of a wind velocity sensor and a wind direction sensor. The wind velocity measurement range is from 0.5 m/s to 50 m/s with an accuracy of ± 0.3 m/s. The wind direction sensor has a 0° to 355° range with an accuracy of ± 3°. The wind velocity sensor has a pulse output, and the wind direction sensor has an analog voltage output. It requires a supply voltage of 3 V to 15 V, and the module diameter is 360.0 mm with a height is 265.0 mm, with a weight of 360 g [55]. The main features of these wind sensors are presented in Table 2.4.

2.7.3 Actuators

With the sensor information and environment perception, the land yacht can use actuators, such as motors, to control the wing sail and the steering mechanism. For this purpose, a servo motor is ideal. The servo motor allows for the precise control of angular position [56]. A low weight servo motor with enough torque and low power consumption is ideal.

An example of a servo motor is the Hitec HS-805BB by Sparkfun (30 € to 40 €)

Table 2.4: Characteristics of the Presented Wind Sensors

	FST-200-201	**FST-200-202**	**WM Model 05103**	**WM30**
Wind Velocity				
Range (m/s)	0.5 - 50	-	0 - 100	0.5 - 50
Accuracy (m/s)	n/a	-	± 0.3	± 0.3
Wind Velocity Threshold (m/s)	< 0.5	-	1.0	< 0.5
Wind Direction				
Range (°)	-	0 - 360	0 - 360	0 - 355
Resolution (°)	-	22.5	n/a	n/a
Accuracy (°)	-	± 3.0	±3.0	±3.0
Wind Velocity Threshold (m/s)	-	0.8	1.1	n/a
Module				
Input Voltage (V)	12 - 36	12 - 36	8 - 24	3 - 15
Power Consumption (mW)	576 @12 V	576 @12 V	60 @12 V	n/a
D×H (mm^3)	127.5×131.5	157.5×234.0	370.0×550.0	265.0×360.0
Output Interface	Voltage or Current	Voltage or Current	Voltage or Current	Voltage
Weight (g)	n/a	n/a	1000	360
Price (€)	100 - 200	100 - 200	1000 - 1400	n/a

n/a - not available

[57]. This servo operates between 0° and 180°, has a voltage range of 4.8 V to 6 V, a torque of 19.5 kg cm and an angular velocity of 53 Revolutions per Minute (RPM), with no load at 4.8 V, and it has a power consumption of 3.36 W without load. This servo motor has a width of 30.0 mm, a length of 66.0 mm, a height of 84.0 mm and weighs 152 g [57]. Another example is the DSS-M15S (10 € to 15 €) [58]. This servo motor can rotate between 0° and 270°, and operates between 6.0 V and 7.0 V with a 13.5 kg cm and 15.0 kg cm of torque at 6.0 V and 7.0 V, respectively. With no load, this servo motor has a free running power consumption of 80/100 mA (6.0/7.0 V) and a stall current of 1.8/2.0 A (6.0/7.0 V), with an angular velocity of 56 RPM and 63 RPM at 6.0 and 7.0 V, respectively. Finally, the dimensions are 54.5 mm of length, 20.0 mm of width and 47.5 mm of height [58]. Although these servo motors have a high torque, they also have a high power consumption. An alternative is the SER0006 by DFRobot (3 € to 4 €) [59]. This motor has an operation range of 0° to 180° with a torque of 1.6 kg cm, and an input voltage of 4.8 V to 6 V with a stall power consumption of 2.4 W. It has an angular velocity of 83 RPM, a length of 22.0 mm, a width of 12.5 mm and a height of 29.5 mm with a weight of 9 g. Another low torque servo motor is the SER0020, also by DFRotobot (10 € to 12 €) [60]. This motor operates between 0° and 180° with an input voltage of 4.8 V to 6 V and a torque of 3.5 kg cm at 4.8 V, or 4.5 kg cm at 6.0 V. The free running power consumption is 96 mW at 4.8 V and 150 mW at 6.0 V, and the stall power consumption is 1.4 W at 4.8 V, and 2.1 W at 6.0 V. The angular velocity is 59 RPM and 83 RPM at 4.8 V and 6.0 V, respectively. Finally, it has a length of 40.2 mm, a width of 20.2 mm and a height of 43.2 mm, with a weight of 38 g. The servo motors main characteristics are presented in Table 2.5.

Table 2.5: Characteristics of the Presented Servo Motors

	Hitec HS-805BB	**DSS-M15S**	**SER0006**	**SER0020**
Torque (kg cm)	19.5	13.5 @6.0 V/ 15.0 @7.0 V	1.6	3.5 @4.8 V/4.5 @6.0 V
Angular Velocity (RPM)	53 @4.8 V	56 @6.0 V/63 @7.0 V	83	59 @4.8 V/83 @6.0 V
Range (°)	0 - 180	0 - 270	0 - 180	0 - 180
Input Voltage (V)	4.8 - 6.0	6.0 - 7.0	4.8 - 6.0	4.8 - 6.0
Free Running Power Consumption (W)	3.36	0.48 @ 6.0 V/0.7 @ 7.0 V	n/a	0.096 @4.8 V/0.150 @6.0 V
Stall Power Consumption (W)	n/a	10.8 @6.0 V/14.0 @7.0 V	2.4	1.4 @4.8 V/2.1 @6.0 V
W×L×H (mm^3)	30.0×66.0×84.0	54.5×20.0×47.5	22.0×12.5×29.5	40.2×20.2×43.2
Weight (g)	152	n/a	9	38
Price (€)	30 - 40	10 - 15	3 - 4	10 - 12

n/a - not available

2.7.4 Logic Components

In order to log sensor data and control the system, it is necessary to include a microcontroller or a computer board. The microcontroller can be used to log the sensor data, and the computer board can be used to control the system.

2.7.4.1 Microcontroller

Microcontrollers used in a land yacht should have enough Input and Output (I/O) pins for all the sensors and actuators, low power consumption and enough peripherals (timers, interrupts and communication interfaces) for the implementation of all functionalities. The STM32 microcontrollers by ST have a 32 bit ARM based architecture [61]. These high performance devices have real-time capabilities and offer a high amount of connectivity options such as: Controller Area Network (CAN) Bus, Serial Peripheral Interface (SPI), Inter-Integrated Circuit (I^2C), Universal Asynchronous Receiver-Transmitter (UART), etc. These connectivity options make this device ideal for sensor data logging.

2.7.4.2 Single-board Computer

A Single-Board Computer (SBC), as the name implies, is a computer built on a single board which has all the necessary components to be functional [62]. These computers offer a low price and low power solution for several applications. The Raspberry Pi is an SBC that can run the Ubuntu Core operating system, which is a Linux distribution, necessary to run ROS (on 64-bit machines) [63][64]. Some examples of the Raspberry Pi SBC are: the Raspberry Pi 2 (30 € to 40 €) [65], the Raspberry Pi 3 (30 € to 40 €) [65], the Raspberry Pi A+ (25 € to 30 €) [65], the Raspberry Pi 3 Model A+ (30 € to 40 €) and B+ (30 € to 40 €) [65] and the Raspberry Pi 4 B (30 € to 65 €) [65]. These SBC characteristics are presented in Table 2.6.

2.7.5 Solar Panel/Battery

A solar panel can be used to provide the vehicle with energy to power its electronic components. It must be connected to a battery with enough power output to drive the system. Different kinds of batteries require different methods of charging [66]. Lead-acid and Lithium-ion batteries require a constant voltage source, while nickel based require a constant current source [66]. Given this information, with the use of a buck-boost converter, it is possible to maintain the input voltage of a lead-acid or a lithium-ion battery [67].

2.7.6 Remote Control and Telemetry

The land yacht requires a module for communication with a base station. The module will transmit information of the state and pose of the vehicle (telemetry) and will receive commands from the base station to control the land yacht (remote control).

An example of these types of modules is the Long Range RF link kit by Seeedstudio (15 € to 20 €) [68]. This device uses 433.92 MHz radio frequency with a 1.5 MHz bandwidth. It has a 2 km range with a sensitivity of -105 dBm and transmits data with a rate of 5 kbit/s. Furthermore, its voltage range is 3 V to 9 V with a current draw of 2.5 mA (22.5 mW) and it has a width of 35 mm, a length of 25 mm and a height of 7 mm. Another

Table 2.6: Characteristics of the Presented Single-Board Computers

Raspberry Pi	**2**	**3**	**A+**	**3 Model A+**	**3 Model B+**	**4 B**
Processor	ARM Cortex-A7 32-bit	ARM Cortex-A53 64-bit	ARM1176JZF-S 32-bit	ARM Cortex-A53 64-bit	ARM Cortex-A53 64-bit	Cortex-A72 (ARM v8) 64-bit
Number of Cores	4	4	1	4	4	4
Clock	900 MHz	1.2 GHz	700 MHz	1.4 GHz	1.4 GHz	1.5 GHz
Output Interface	UART,SPI,I^2C	UART,SPI,I^2C	UART,SPI,I^2C	UART,SPI,I^2C	UART,SPI,I^2C	UART,SPI,I^2C
GPIO	40	40	40	40	40	40
RAM	1 Gb	1 Gb	256 Mb	512 MHz	1 GB	1 GB , 2 GB, 4 GB
Ethernet	10/100M	10/100M	No	No	Gigabit - Over USB 2.0	Gigabit
Bluetooth	No	4.1 LE	No	4.2 BLE	4.2 BLE	5.0
Wi-Fi	No	802.11n	No	2.4GHz and 5GHz 802.11 b/g/n/ac	2.4GHz and 5GHz 802.11 b/g/n/ac	2.4GHz and 5GHz 802.11 b/g/n/ac
Storage	microSD	microSD	microSD	microSD	microSD	microSD
Power Rating (A)	0.80	1.34 @5 V	0.20	1.30 @5 V	3.00 @5 V	3.00 @5 V
W×L×H (mm^3)	56.5×85.6×17.0	56.5×85.6×17.0	56.5×65.0×11.0	56.0×65.0×10.0	56.5×85.6×17.0	56.5×85.6×11.0
Weight (g)	45.0	14	23	29	45	n/a
Price (€)	30 - 40	30 - 40	25 - 30	30 - 40	30 - 40	30 - 65

n/a - not available

example of a radio frequency module is the APC802 Radio Communication Module by DFRobot (35 € to 40 €) [69]. This device operates with a frenquency range of 418 MHz to 455 MHz, or 470 MHz to 510 MHz, with a bandwidth of 0.2 MHz. It has a 2.8 km range with a sensitivity of -117 dBm and transmits data with a rate of 9800 bit/s with RS232, RS485 or Transistor-Transistor Logic (TTL). It has an input voltage of 5 V and a current draw of 400 mA (2 W). Finally, it has a width of 50 mm, a length of 31 mm and a height of 7 mm.

2.8 Analysis and Comparison

In this section, an analysis and comparison of the land yachts previously presented in this chapter is made. Moreover, the materials and electrical components presented previously are also compared and analysed.

2.8.1 Manned Land Yachts

The manned land yachts presented in this chapter are composed of both cloth and rigid-wing sails. The Green Bird and the Sail Car are both composed of a rigid-wing sail. Although the Sail Car did not break any speed record, it is worth mentioning the free rotating wing sail, which is only actuated through its tail, reducing the energy necessary to turn the whole wing sail to a favourable angle of attack. Furthermore, the Sail Car controls the wing sail autonomously, *i.e.*, the tail actuator reacts to the anemometer readings.

2.8.2 Unmanned Land Yachts

The unmanned land yacht prototypes presented previously are composed of metal frames and wing sails with relatively small wingspans (800 mm to 1000 mm). These wing sails share the NACA four-digit airfoil design and appear to have similar dimensions. Table 2.7 shows a summary of the main mechanical characteristics of the previous land yachts. These systems rely on the wind, GPS and inertial sensors (apart from Dong *et al.*, which does not use GPS) for propulsion and navigation. Moreover, the wing sails and the steering mechanisms are all controlled by servo motors. Table 2.8 shows a summary of the control characteristics of the unmanned land yachts studied.

Table 2.7: Main Mechanical Characteristics of the Presented Unmanned Land Yachts

Land Yacht	**Frame**			**Wing sail**			
	Material	**Weight** (kg)	**L×W×H** (mm^3)	**Airfoil Profile**	**H×L** (mm^2)	**Weight** (kg)	**Area** (m^2)
Xie et al. (2015) [9]	Aluminium	19	n/a	NACA0018	1000 × 250	2.15	0.25
Chen et al. (2018) [10]	Aluminium	light	n/a	3×NACA0018	1000 × 250	2.15	0.25
Dong et al. (2019) [11]	Steel & plastic	n/a	350×200×100	NACA0015	800 × 300	n/a	0.24

n/a - not available

Table 2.8: Control Characteristics of the Presented Unmanned Land Yachts

Land Yacht	Sensors			Controller	Actuators	
	Wind	GPS	IMU		Wing Sail	Steering
Xie et al. (2015) [9]	Yes n/s	Yes n/s	Yes n/s	STM32 & A3C6410	Servo Motor	Servo Motor
Chen et al. (2018) [10]	Yes n/s	Yes n/s	xsens MTi	STM32 & S3C6410	Servo Motor	Servo Motor
Dong et al. (2019)	CALYPSO	n/a	HI219	Raspberry PI	Servo Motor	Servo Motor

n/a - not available
n/s - not specified

2.8.3 Materials and Electrical Components

In this subsection, the different materials and electrical components that were studied previously in this chapter are compared and analysed.

2.8.3.1 Frame Materials

Both aluminium and stainless steel are excellent options for the development of a land yacht. However, the choice of the material is situational. If the purpose is to develop a low weight and cheap land yacht, aluminium is the best candidate. However, if the purpose is to develop a heavier, more robust vehicle, stainless steel is more suitable.

2.8.3.2 IMU

The IMU presented in Table 2.1 are good candidates for a land yacht; however, most of them are expensive. If the price is not to be considered, the NavChip would be the best bet, given its low velocity random walk, low angular random walk and its power consumption. Moreover, a GNSS module can be connected directly to this device. If a cheaper alternative is to be chosen, the BMF055 is the best choice. This IMU has low velocity random walk, low angular random walk and has an integrated magnetometer, necessary for updating the vehicles orientation.

2.8.3.3 GNSS Unit

For the GNSS module, the devices presented in Table 2.2 are all similar in terms of technical specifications and price; however, they differ slightly. The Ultimate GPS has the best accuracy but, on the other hand, it has the highest power consumption and the lowest number of channels. The GM Series has the highest number of channels and lowest power consumption, but it has the lowest sensitivity the higher price. Finally, the NEO-M8 is "in the middle" regarding the specifications and price. It also has the higher number of output interfaces and the lowest input voltage.

2.8.3.4 Rotary Encoder

The rotary encoders presented in Table 2.3, known as absolute rotary encoders, determine the absolute value of the angular position without needing the previous readings, as mentioned before. In order to determine the angular position of a wing sail, it is necessary

to have good resolution and accuracy, because in some airfoils the difference of having an angle of attack of 9° and 10° is significant. Given this, both rotary encoders presented in Table 2.3 are ideal for a rigid-wing sail land yacht. If the price is not a factor to consider, the AMT21 would be the ideal rotary encoder, but the EAW is a perfectly fine encoder for the task, with a much lower cost.

2.8.3.5 Wind Sensor

While knowing the relative direction of the wind is a must to control the wing sail, obtaining the relative wind velocity is a plus. The wind sensor modules in Table 2.4 measure the wind velocity, the wind direction or both. Consequently, the choice of the module will dictate if one or two wind sensors will be needed. For this project, a module that can sense both the wind velocity and direction is ideal, such as the WM Model 05103 or the WM30. Both of these modules have an optimal measurement range and accuracy.

2.8.3.6 Actuators

The actuator will highly depend on the dimensions and weight of the vehicle and its modules. A land yacht with a wing sail with a higher wingspan or a different type of material will have a higher weight and, thus, require a higher torque actuator. Also, a vehicle with a higher weight will require a higher torque motor to control the steering mechanism. A selection of higher and lower torque servo motors is presented in Table 2.5. The higher torque motors have a higher power consumption and weight than the lower torque ones. Given this, in order to be as energy efficient as possible, the servo motor with the lowest torque necessary must be chosen.

2.8.3.7 Microcontroller

In the case the SBC does not have enough interfaces to communicate with the sensors, a microcontroller can be utilised to log and send the data using an interface ideal for the SBC. The STM32 is an ideal device given its various peripherals and communication interfaces, as mentioned before. It should have enough I/O pins for every sensor and actuator.

2.8.3.8 Single-board Computer

To run ROS it is necessary to have a Linux distribution in a 64 bit machine. According to Table 2.6, the Raspberry 2 and the Raspberry A+ can be excluded. The remaining SBC are all ideal with similar prices, but the Raspberry 4B has a slightly better Central Processing Unit (CPU) and various different selections of Random-Access Memory (RAM).

2.8.3.9 Solar Panel/Battery

The solar panel and battery are dependent on the power budget of the system. A system with a higher power budget will require a heavier, more expensive battery and solar panel than a system with a lower power budget.

2.9 Conclusion

This Chapter served as a means to study existent competitions regarding land yachts, existing manned and unmanned land yachts, and the different components needed to develop an autonomous ROS based wing sail propelled land yacht. Finalising the Chapter, an analysis and comparison of the main themes presented was made.

In conclusion, an energy efficient wing sail land yacht must be light and stable, and the selected components must consume as little energy as possible. The use of a solar panel for battery charging is also ideal as it is a renewable and free source of clean energy.

Chapter 3

Platform Design

After the review of the land yachts and their components, this chapter proposes an autonomous land yacht solution, including a conceptual high level architecture, to the problem stated in the Introduction.

3.1 Proposal

Given the need to design an energy efficient wind propelled vehicle, as stated in Chapter 1, a wing sail autonomous land yacht, with a design similar to the Wing Sail Land Yacht by Dong *et al.* [11], is proposed. In addition it incorporates a flap tail on the wing sail, making it a free rotating wing sail. With less weight on the wing sail, the torque necessary to turn the flap will also be reduced, consuming less power.

In order for the vehicle to be low weight and robust, a stainless steel frame is proposed for the chassis. The frame format reduces the drag generated on the chassis and it has a parallelepipedic shape with a width of 360 mm, a length of 700 mm and a thickness of 5 mm. Furthermore, it has two side frames to attach modules, if necessary. Given its dimensions and the density of stainless steel (7700 kg/m^3), the chassis has a mass of 9.43 kg. It is expected for the vehicle to have a maximum payload of 10 kg, with the sail included. The land yacht has four solid rubber wheels for manoeuvrability and the rear wheels have a higher baseline than the front wheels, making it triangular for increased stability. The front wheels have a steering mechanism that is controlled autonomously or by a human operator. Finally, the wing sail is a free rotating NACA00xx airfoil (the 00 means the airfoil is symmetrical and the xx is the percentage of the chord thickness [1]) with a flap attached to it. The flap uses the same airfoil as the wing sail and, as mentioned before, the flap will be used to set the angle of attack of the wing sail.

For the control system, the proposed land yacht uses a SER0020 [60] servo motor to control the steering mechanism. This servo motor has enough torque (4.5 kg cm) to turn a steering mechanism on a vehicle with a total mass of 19.43 kg (chassis + payload). For the flap control, a SER0006 [59] servo motor is proposed since the flap control will require less torque than the necessary to turn the vehicle. For motion sensing, the vehicle utilises a BMF055 [43] inertial sensor and a NEO-M8 [46] GNSS unit. In order to know the wing sail and steering angular position, an EAW [51] rotary encoder is used. The

relative wind velocity and direction is sensed using an FST-200-201 [52] and an FST-200-202 [53], respectively. Finally, a Raspberry Pi 4B [65] SBC is used to control the system. The concept of the proposed land yacht is shown in Figure 3.1 and a block diagram of the system architecture is show in Figure 3.2, where the red, purple and blue areas correspond to the actuators, power supply and sensor data, respectively.

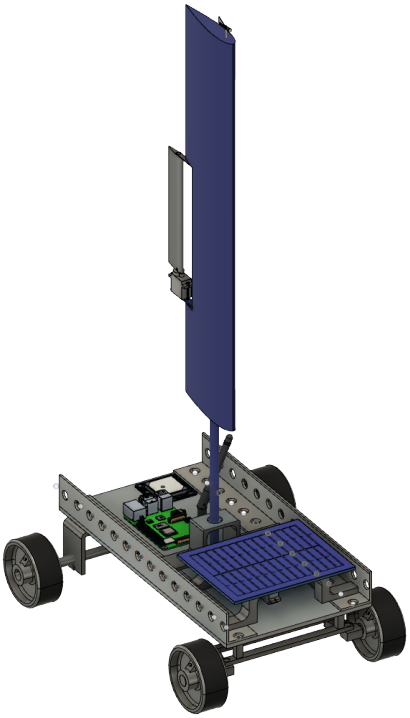


Figure 3.1: Proposed Land Yacht Conceptual Model

For the telemetry and remote control, the vehicle uses an APC802 Radio Communication Module, which operates on a frequency range acceptable for amateur use in the European Table of Frequency Allocations [70].

Given the information presented in Chapter 2, the power budget of the proposed system is presented in Table 3.1, totalling 22.865 W. For the power supply, given on Table 3.1, a 12 V battery that can discharge at least 1.91 A ($12.00\,\mathrm{V} \times 1.91\,\mathrm{A} = 22.92\,\mathrm{W}$) is needed. A Lithium-Polymer (LiPo) battery with a capacity of 4400 mA h and a 10 C (or higher) discharge rate is proposed. A buck-boost converter is used to make the connection between a solar panel and the battery, making the input voltage of the battery constant. If the platform is required to have additional sensors or equipment an area of $0.1\,\mathrm{m}^2$ is available under the solar panel and an area of $0.05\,\mathrm{m}^2$ is available on the backside of the vehicle. Given the previous power budget and the battery, the vehicle has an autonomy of 83 min, if every component is functioning at maximum power consumption and assuming there is no power generated by the solar panel. A battery with higher capacity can be used as long as it supplies 22.865 W.

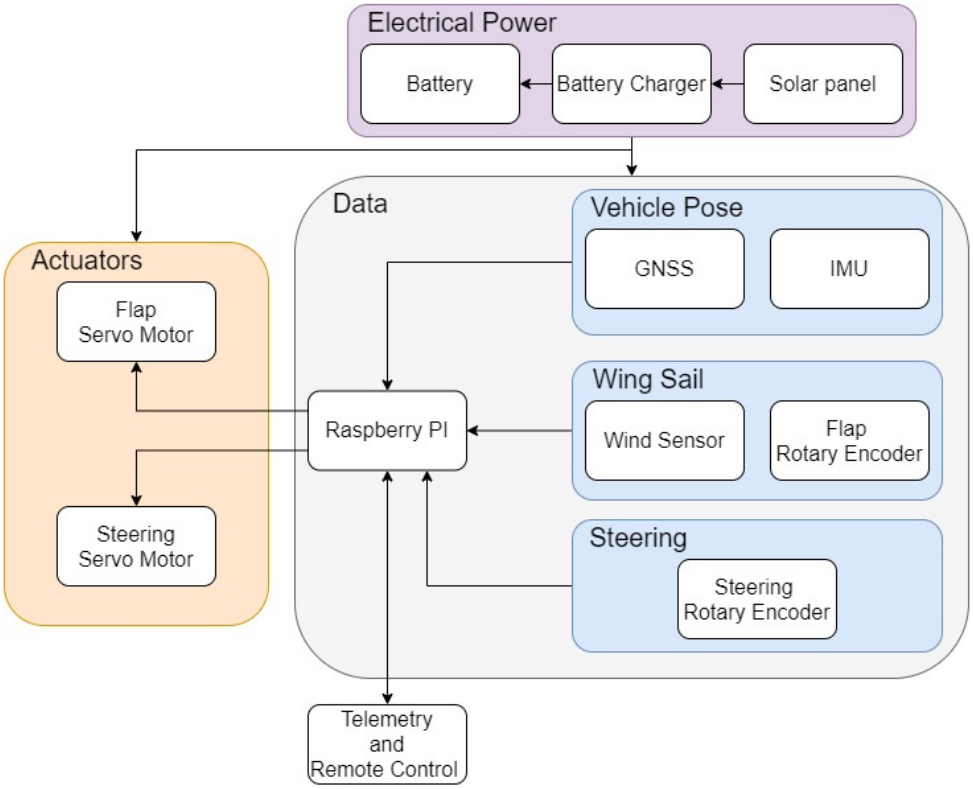


Figure 3.2: System Diagram

3.2 Wing Sail Characteristics

The wing sail, with an angle between the wing sail chord line and the apparent wind direction (angle of attack α), generates perpendicular and parallel forces relative to the apparent wind, known as lift (L) and drag (D), respectively, and their scale depend on the corresponding lift (C_L) and drag (C_D) coefficients [71]. The lift and the drag forces are presented in Equations 3.1 and 3.2, respectively, where ρ is the air density, V is the apparent wind speed and A is the area of the wing sail. The lift and drag forces generate

$$L = \frac{1}{2}\rho V^2 A C_L \tag{3.1}$$

$$D = \frac{1}{2}\rho V^2 A C_D \tag{3.2}$$

a resultant force, as show in Figure 3.3 [14]. The wing sail must be able to take the most advantage of the wind, regardless of its direction, therefore, the airfoil must be symmetrical. The Reynolds Number (Re) represents the ratio of kinematic or inertial

Table 3.1: Power Budget of the System

Component	**Power Consumption** (W)
SER0020	2.100
SER0006	2.400
BMF055	0.050
NEO-M8	0.063
FST-200-201	0.576
FST-200-202	0.576
EAW	0.100
APC802	2.000
Raspberry Pi 4B	15.000
Total	**22.865**

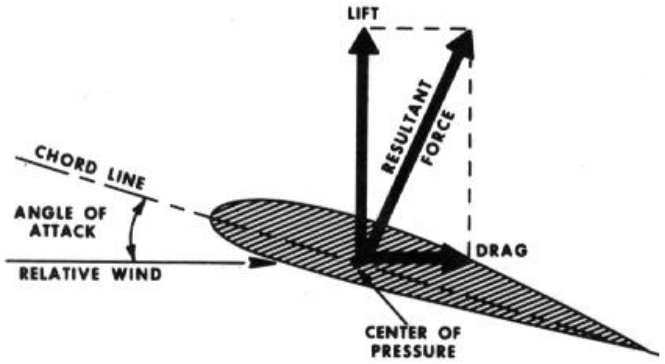


Figure 3.3: Forces Applied to an Airfoil [14]

forces to the viscous forces in a fluid, as shown in Equation 3.3, where ρ is the density of the medium, V the velocity of the flow, L the characteristic length, and μ the viscosity of the medium [14]. In [14], the authors analyse an Re range of $200000 < Re < 500000$. The

$$Re = \rho V L/\mu \tag{3.3}$$

same Re range is used in this dissertation to select an airfoil. Given this information, the NACA0012, NACA0018, EPPLER 473, E169 and NACA63(3)-018 airfoils are considered. Figure 3.4 presents the lift and drag coefficients of each of the considered airfoils relatively to the angle of attack.

Analysing Figure 3.4, any of these airfoils is a valid candidate to the land yacht, as they all have a high lift to drag ratio (C_L/C_D) at a certain angle of attack (α). Nevertheless, the NACA63(3)-018 airfoil was selected because it has a high lift to drag ratio of 73 at an angle of attack of 6.75° and, furthermore, at an angle of attack of 10° the drag coefficient is still low (0.02). A 2D model of the NACA63(3)-018 airfoil is presented in Figure 3.5.

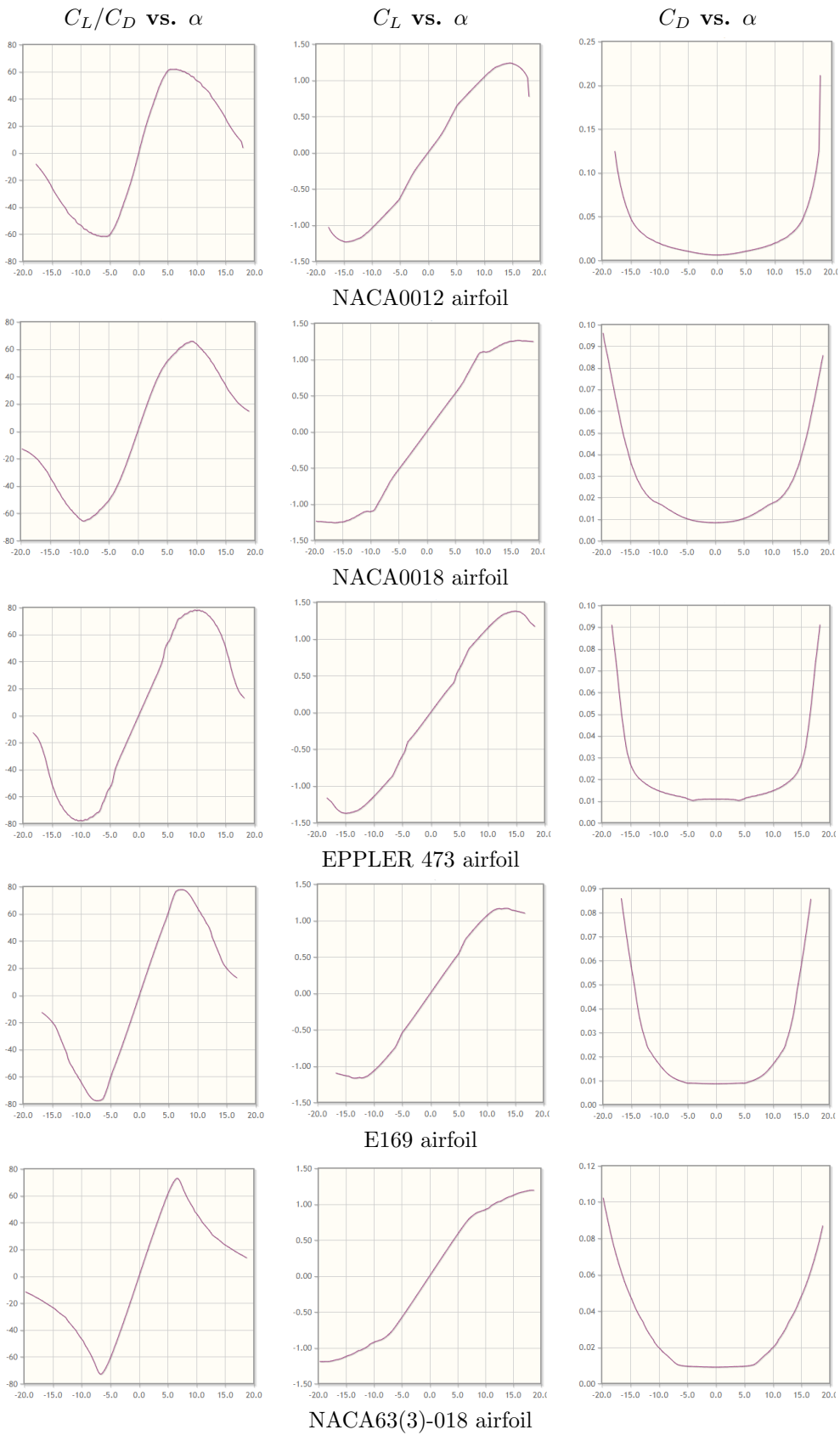


Figure 3.4: NACA0012; NACA0018; EPPLER 473; E169; NACA63(3)-018 Airfoils

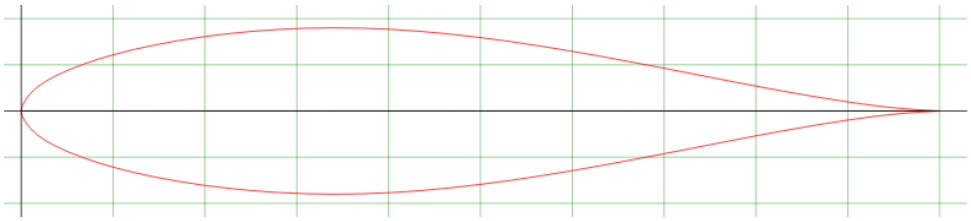


Figure 3.5: NACA63(3)-018 Airfoil 2D Profile

As mentioned previously, the total mass of the land yacht is 19.43 kg and, in order for the vehicle to start its movement, the initial acceleration, a, must not be null ($|a| > 0$). The force F required to put a vehicle in motion, given its mass m and acceleration a, is calculated using Newton's second law (Equation 3.4). The force vectors in this case are

$$F = ma \tag{3.4}$$

presented in Figure 3.6, where T is the thrust force generated by the lift force, L and D are the lift and drag forces, respectively, R is the resultant force from the lift and the drag forces, α is the angle of attack and β is the angle between the lift vector and the vehicles heading. Consequently, the force necessary to put the land yacht in motion is given by Equation 3.5, where L is the lift force, β is the angle between the lift vector and the vehicle heading, m is the mass of the vehicle, a is the acceleration and F_f is the friction between the wheels and the ground. Replacing the lift factor in Equation 3.5 with

$$L\cos\beta = ma + F_f \tag{3.5}$$

the lift expression in Equation 3.1 and given that the friction force can be expressed by μmg, where μ is the friction coefficient, m is the vehicle mass and g is the gravitational acceleration, Equation 3.6 is obtained.

$$\frac{1}{2}\rho V^2 A C_L \cos\beta = ma + \mu mg \tag{3.6}$$

The lift vector is perpendicular to the apparent wind direction and the T vector is parallel to the vehicle x axis, meaning β can be expressed by $\beta = |\alpha_{wind} + 90|$, where α_{wind} is the apparent wind direction.

With $\beta \leq 45°$, a mass of 19.43 kg, an apparent wind velocity of 5 m/s, a starting acceleration of 0.15 m/s^2, a lift coefficient of 0.75, assuming the air density is 1.225 kg/m^3 and a friction coefficient of 0.002, the wing area of approximately 0.4 m^2 is obtained from

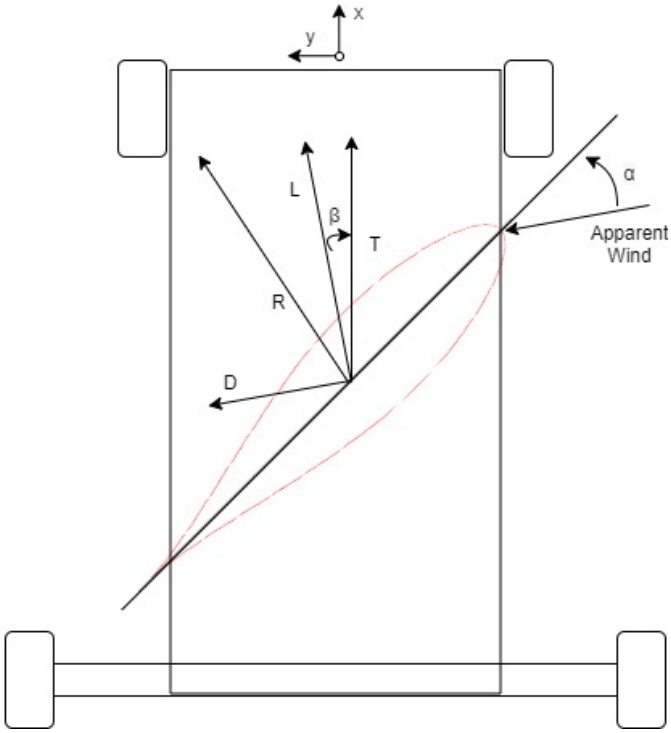


Figure 3.6: Land Yacht Forces

Equation 3.6. This results in a NACA63(3)-018 airfoil with a chord of 400 mm and a wingspan of 1000 mm.

With these parameters, the wing sail can generate an effective lift force with apparent wind directions of $45° \leq \alpha_{wind} \leq 135°$ and $-45° \geq \alpha_{wind} \geq -135°$.

3.3 Control System

The control system is responsible for placing the wing sail in the correct angular position, using the flap, and to actuate the steering mechanism in order for the vehicle to maintain a trajectory.

3.3.1 Wing Sail

To control the wing sail angular position, the system requires as inputs the wind direction, the current wing sail angular position and the desired angle of attack. First, the current angle of attack is calculated from the wind direction and the angular direction of the wing sail. The current angle of attack is then subtracted from the desired angle of attack and the difference is provided as an input to the controller. Finally, the controller determines the optimal angular position of the wing flap and sends the required Pulse Width Modulation (PWM) signal to the flap servo motor, as shown in Figure 3.7.

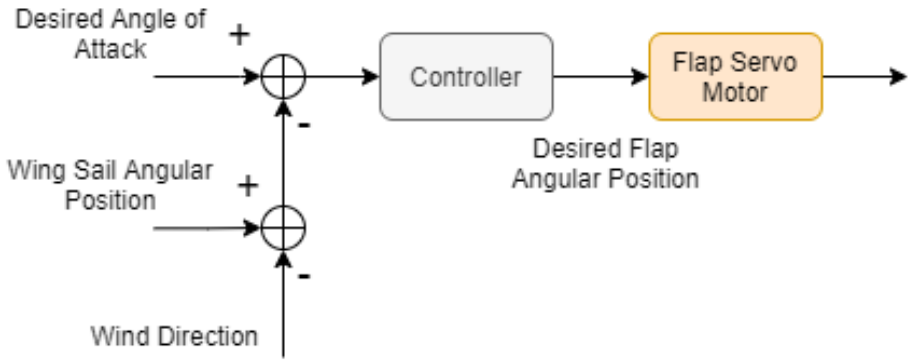


Figure 3.7: Wing Sail Control System

3.3.2 Steering Control

To follow a path, the vehicle must control its heading. The system determines its current position and orientation based on the inputs from the GNSS and IMU devices. The steering control system subtracts the current and the desired poses to calculate the desired steering angle. Finally, the steering control system generates the corresponding PWM signal to the steering servo motor. Figure 3.8 shows the diagram of the proposed steering control system.

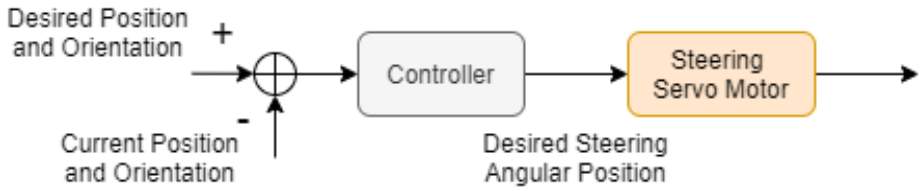


Figure 3.8: Steering Control System

3.4 Conclusion

In this chapter, a conceptual model of the vehicle, a wing sail and a control diagram were presented.

It is noteworthy that the proposed land yacht solution is a support platform for data acquisition and monitoring, meaning that it can be equipped with diverse sensors and equipment. Since the controller is a Raspberry PI, several types of sensors can be used. These can range from simple image capturing sensors, to any other kind of sensors. The four wheeled chassis allows manoeuvrability and the rear wheel baseline is larger than the front wheel baseline for increased stability and, thus, prevent capsizing. The stainless steel chassis is robust, and its shape reduces the overall drag force. The maximum system power consumption was calculated and an appropriate battery selected. When missions require it, the battery can be substituted by one with higher capacity. The NACA63(3)-018 airfoil was chosen for the wing sail. Other possible candidates were also presented

and should not be discarded, especially the EPPLER 473 airfoil, as the latter has a higher range of angles with high lift/drag ratio, comparatively to the other presented airfoils. The chosen wing sail dimensions were chosen given a number of factors such as the vehicle maximum payload, initial acceleration, apparent wind velocity, etc. Finally, the wing sail and steering mechanism control system was considered. The wing sail control system takes into account the current angle of attack and the desired angle of attack, adjusting the flap accordingly. The steering mechanism control system takes into account the land yacht current position and orientation and actuates on the steering servo motor.

Chapter 4

Platform Simulation

In this chapter, a number of ROS based simulation platforms are described, as well as the simulation of the land yacht proposed in Chapter 3.

4.1 ROS Compatible Simulation Platforms

As mentioned in Chapter 1, a requirement of this dissertation is to adopt the ROS framework for the control of the land yacht. However, due to the COVID-19 pandemic, the dissertation will focus on the simulation rather than on the building, assembly and test of the designed platform. To simulate the proposed land yacht behaviour it is necessary to find not only a ROS-compatible simulation tool, but one which supports a fluid dynamics physics engine, that enables the simulation of wind propelled vehicles. The platforms that are being considered are: CoppeliaSim [16][72], Gazebo [73], Modular Open Robots Simulation Engine (MORSE) [16][16][74], Webots [75][16] and Unified System for Automation and Robot Simulation (USARSim) [16][76].

4.1.1 CoppeliaSim

CoppeliaSim is a 3D general purpose robotic simulator developed by Coppelia Robotics [16][72]. This simulator is based on a distributed control architecture, meaning that each module can be controlled using ROS nodes. CoppeliaSim supports Windows, Linux and Mac operating systems and its programs can be written in C/C++, Python, Lua, Matlab, Urbi, Octave or Java [16][72].

The CoppeliaSim simulator has two physics engines: the Open Dynamics Engine (ODE) [72], which simulates rigid body dynamics and collisions [77], and the Bullet Engine [72], which simulates soft and rigid body dynamics and collision detection [78].

Although this simulator is compatible with ROS, it does not have a fluid dynamics physics engine and can not directly simulate wind mechanics. In order to simulate the wind, its behaviour must be fabricated by adding auxiliary forces to the simulation, making it possible to simulate a wind powered vehicle, yet also making it more time consuming and more susceptible to error.

4.1.2 Gazebo

Gazebo is an open source 3D robotic simulator created in the University of Southern California and later integrated in the ROS framework [16][73]. It is currently maintained by the Open Source Robotics Foundation (OSRF) and supports Linux and Mac platforms [16][73].

This simulator uses ODE as the default physics engine; however, it can make use of different physics engines such as Bullet, Simbody and Dynamic Animation and Robotics Toolkit (DART). Its main programming language is C++ and Gazebo has support code for sensor simulation, with the option to add noise to the simulated readings, and actuator control. It also provides a high degree of realism in its simulations by adding friction, mass and other attributes to the simulated objects and by using the Object-Oriented Graphics Rendering Engine (OGRE) to provide 3D objects and 3D environments [16][73].

Gazebo has access to various plugins that allow the developer to have control of almost any aspect of the simulation. These code plugins are compiled shared libraries ready to be included in simulation. There are six types of Gazebo plugins: World, Model, Sensor, System, Visual and Graphical User Interface [73]. Amongst these plugins, there is an aerodynamics plugin, called `liftdragplugin`, which simulates the Lift and Drag forces on an airfoil [15]. This plugin makes the Gazebo simulator an ideal tool for the simulation of the land yacht presented in this dissertation.

4.1.3 MORSE

MORSE is a generic robotics simulator created by the Laboratory for Analysis and Architecture of Systems in France [16][74]. This simulator focuses on realistic 3D simulations of small to large environments, both indoor and outdoor, with several robots being simulated at the same time.

The MORSE simulator is ROS compatible, based in the Python programming language and can be entirely controllable from the command line. Furthermore, this simulator can simulate a set of standard sensors, such as cameras, GNSS receivers and odometry. MORSE can also simulate control systems, such as motor control and waypoint controllers. The rendering is based on the Blender Game Engine and uses the Bullet physics engine. Finally, this simulator does not simulate the wind or aerodynamic forces [16][74].

4.1.4 Webots

Webots is an open source robot simulator, created by Cyberbotics for professional use, that is used to simulate, model and program robots in 3D environments [16][75]. It is ROS compatible, uses different programming languages, such as C/C++, Python and Matlab, and supports Linux, Windows and Mac platforms.

This simulator uses ODE as its physics engine and OGRE for the 3D rendering. It can also simulate various sensors used in robots, such as Light Detection And Ranging (LiDAR), light sensors, proximity sensors and GPS. Finally, Webots has a physics plugin that can be used to simulate different forces, including aerodynamic forces [16][75].

4.1.5 USARSim

USARSim is a 3D simulator based on the Unreal Engine and was developed to simulate multiple robots in search and rescue environments, such as the RoboCup rescue competitions [16][76].

It is ROS compatible through the use of the ROS Integration plugin [79] and uses C++ as its main programming language. It supports Windows, Linux and Mac platforms, and uses the Mobility Open Architecture Simulation and Tools (MOAST) framework, which provides a modular control system. Furthermore, USARSim supports various types of sensors such as sound sensors, LiDAR, encoders and cameras. However, this simulator does not directly simulate aerodynamic forces or wind dynamics [16][76].

4.1.6 Analysis and Comparison

The presented 3D simulators are all ROS compatible; however, only Gazebo and Webots allow simulating the land yacht, presented in this dissertation, as these can directly simulate aerodynamic forces. The characteristics of the previous simulators are presented in Table 4.1. Gazebo was chosen because it satisfies all requirements (including a plugin to simulate aerodynamic forces) and due to the author's previous familiarity with the simulator.

Table 4.1: Characteristics of the Presented Simulators (Adapted from [16])

	CoppeliaSim	**Gazebo**	**MORSE**	**Webots**	**USARSim**
Main Language	C++	C++	Python	C++	C++
Operating System	Mac Linux	Mac Linux	Mac Linux BSD	Mac Linux	Linux
Physics Engine	ODE Bullet Vortex Newton	ODE Bullet Dart	Bullet	ODE	Unreal
3D Rendering Engine	Internal External	OGRE	Blender Game	OGRE	Karma
ROS Compatibility	Yes	Yes	Yes	Yes	Yes
Wind/Aero. Forces	No	Yes	No	Yes	No

4.2 Simulation Development

The simulation was developed using the Gazebo simulator. For this, the version 7 of Gazebo and Gazebo-ROS were installed. In the development of this dissertation, the Gazebo simulation was always launched using ROS. In order to create the simulation, the vehicle and wing sail model were created using the Fusion 360 Computer-Aided Design (CAD) software. The 3D meshes created in Fusion 360 were integrated with Gazebo, using Spatial Data File (SDF) files and Extensible Markup Language (XML). Apart from importing the 3D meshes to Gazebo, the SDF files connect joints, enable joint

properties and are used to import Gazebo plugins into the environment, for example, the aerodynamic forces plugin.

4.2.1 Vehicle 3D Model

As mentioned previously, the vehicle 3D model was created in Fusion 360 and imported to Gazebo using the corresponding SDF and XML files. The vehicle model is composed of various components (links) and connections (joints). The SDF file where all the vehicle components and joint properties are imported is called the model file. An example of importing a 3D mesh into Gazebo is presented in Listing 4.1, that contains the set of parameters: the link name, the link pose relative to the base link (in this case the base link is the chassis), the inertial properties (such as mass and the inertial matrix), the link collision properties (in this case the collision zone is the same as the model format) and the link visual format. Listing 4.2 illustrates how to add a joint between two links, where the parameters used are the joint type (since it is connecting a wheel, then the joint type is revolute), the joint name, the pose relatively to the child link, the child and parent links (in this case it is the wheel and the chassis, respectively), the rotation axis (in this case the joint rotates in the y axis), and the joint friction and damping. This model file is loaded in another SDF file called the world file. The world file loads all the models and environmental characteristics, such as the ambient light and ground plane. Furthermore, the world file also loads world related plugins. On the other hand, the model file, apart from loading the vehicle components, loads model related plugins, such as the aerodynamic or the sensor plugins. The land yacht model loaded in Gazebo is presented in Figure 4.1. This model is composed of a chassis, four wheels, a steering mechanism composed of 4 links, a wing sail and a flap (11 links and 12 joints).

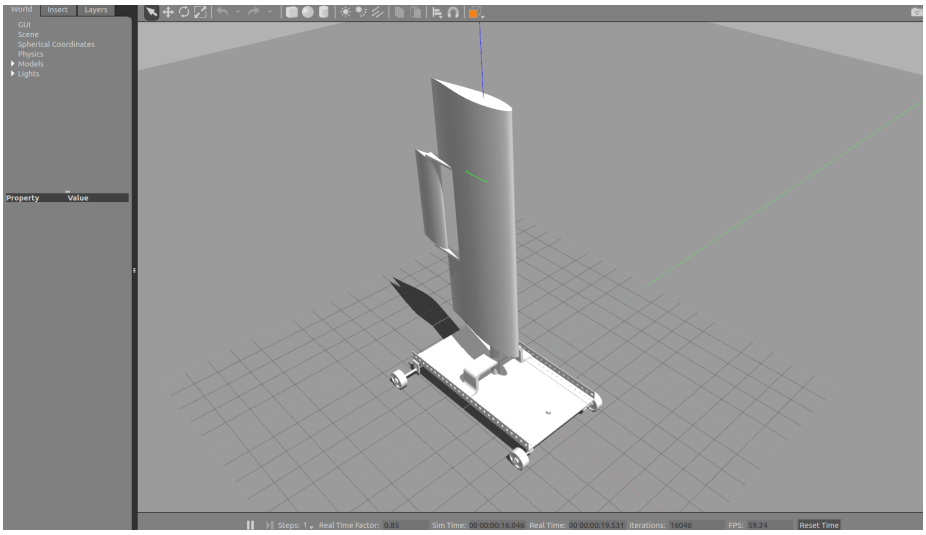


Figure 4.1: Gazebo Land Yacht 3D Model

Listing 4.1: Model SDF Code Example

```
<link name='steering_middle'>
    <pose>3.373 −0.12 0.6 0 0 0</pose>
    <inertial>
        <mass>0.195315</mass>
        <inertia>
            <ixx>1.614e−3</ixx>
            <ixy>4.98132e−7</ixy>
            <ixz>1.72258e−7</ixz>
            <iyy>7.383226e−6</iyy>
            <iyz>2.843631e−6</iyz>
            <izz>1.612e−3</izz>
        </inertia>
    </inertial>
    <collision name='collision'>
        <geometry>
            <mesh>
                <uri>model://my_robot/Cad_Files/SteeringMiddleGazebo.stl</uri>
                <scale>0.01 0.01 0.01</scale>
            </mesh>
        </geometry>
    </collision>
    <visual name='visual'>
        <geometry>
            <mesh>
                <uri>model://my_robot/Cad_Files/SteeringMiddleGazebo.stl</uri>
                <scale>0.01 0.01 0.01</scale>
            </mesh>
        </geometry>
    </visual>
</link>
```

Listing 4.2: Joint SDF Code Example

```
<joint type="revolute" name="left_back_wheel_hinge">
    <pose>0 0 0 0 0 0</pose>
    <child>left_back_wheel</child>
    <parent>chassis</parent>
    <axis>
        <xyz>0 1 0</xyz>
        <dynamics>
            <friction>0.0625</friction>
            <damping>0.4</damping>
        </dynamics>
    </axis>
</joint>
```

4.2.2 ROS Integration

To launch Gazebo using ROS, the package `gazebo_ros_pkgs` [80] was used, as it enables the interface between ROS and Gazebo, allowing ROS to launch Gazebo and the creation of ROS nodes inside Gazebo plugins. A ROS package, named `land_yacht`, was created

specifically for the development of this dissertation. This package uses a launch file to load an empty world from `gazebo_ros_pkgs`, together with the vehicle model created, and a node responsible for the control of the wing flap and the steering. This empty world serves merely as an interface between `gazebo_ros_pkgs` and `land_yacht`. For this to be possible, the world file where the vehicle is loaded must be in a worlds folder inside the package. A block diagram of the Gazebo-ROS interface and the Gazebo elements interface is presented in Figure 4.2.

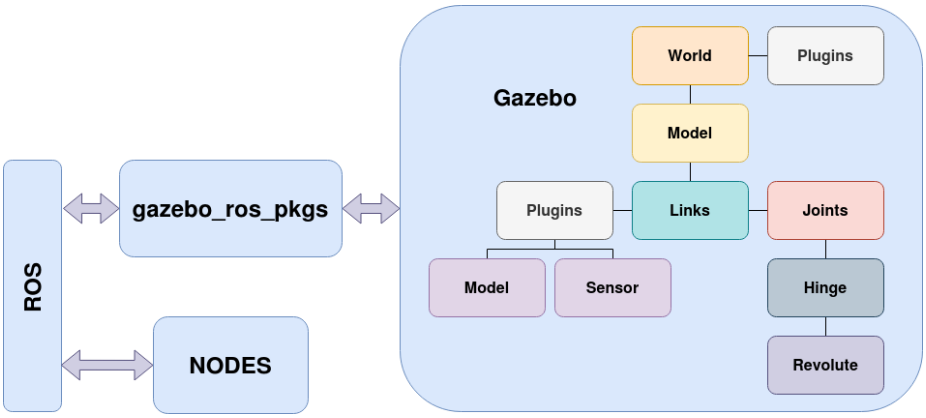


Figure 4.2: Gazebo-ROS Interface Block Diagram

4.2.3 Gazebo Plugins

The Gazebo plugins used for the simulation were the aerodynamic forces plugin `liftdragplugin`, the `hector_gazebo-plugin` for the sensors [81] and other plugins specifically created for this project. Furthermore, the `liftdragplugin` was modified in order to interface with ROS.

The `liftdragplugin` simulates the aerodynamic forces and also the airfoils pitching moment and, thus, the torque generated by the wind on the airfoil. In short, the `liftdragplugin`, apart from generating lift and drag forces, also aligns the wing sail with the direction of the wind. However, this plugin was created to simulate these forces on aeroplanes or fixed wing drones on perfect weather conditions, i.e., without wind. In other words, this plugin assumes the world wind velocity is the vehicle's world linear velocity, which, in this case, is partially correct. The plugin was modified to also use the world's wind velocity and add it to the vehicle's world linear velocity. For example, if the vehicle is facing east and moving with a linear velocity of 1 m/s and the wind is coming from east with a linear velocity of 2 m/s, then the two linear velocities are added, generating a total apparent wind velocity of 3 m/s. Without modifying the plugin, the apparent wind velocity would remain 1 m/s. The plugin was also modified to subscribe to a ROS topic that broadcasts the world wind velocity in the x and y axis. A diagram that helps understand the way this plugin works is presented in Figure 4.3. In order to use this plugin, the wing sail characteristics must be described using the format shown in Figure 4.4, where airfoil's curves are approximated into lines and the obtained slopes are used as parameters in the plugin importation.

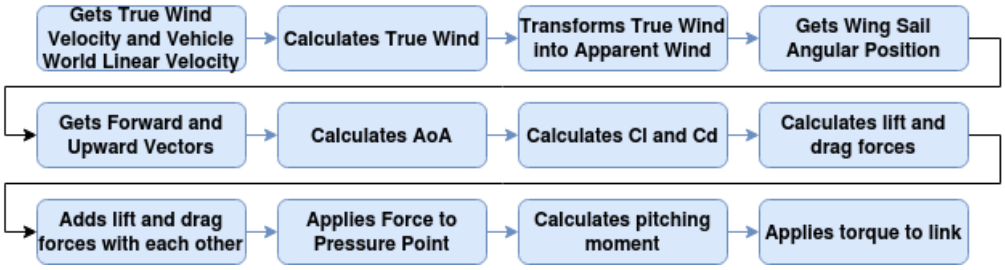


Figure 4.3: Liftdragplugin Diagram

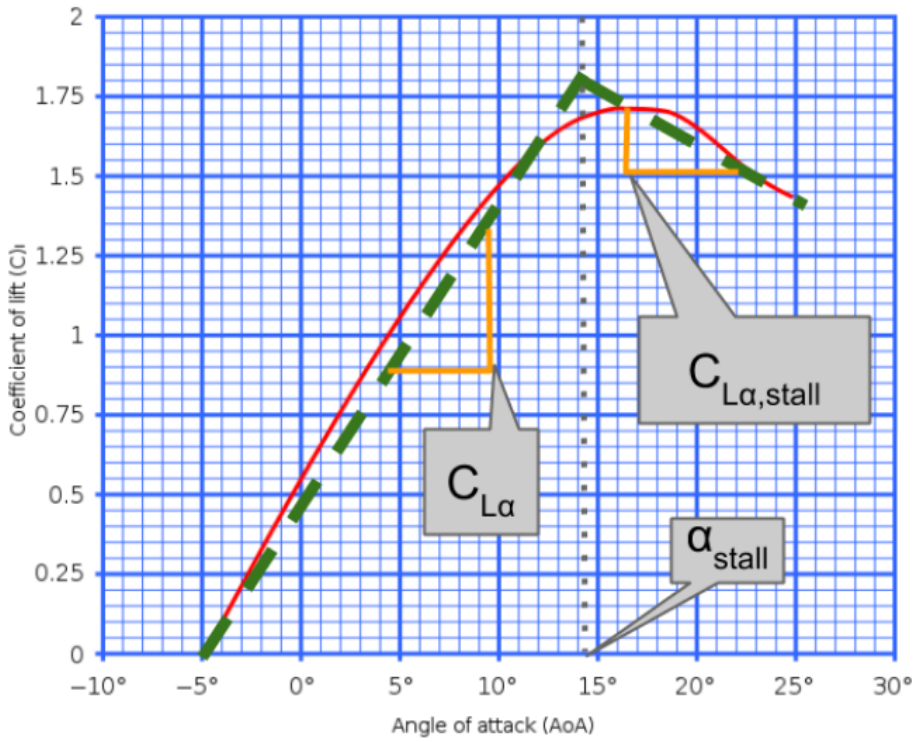


Figure 4.4: Liftdragplugin Wing Characteristic Curve [15]

An alternative to the `liftdragplugin` is the `usv_gazebo_wind_plugin` from `usv_gazebo_plugins` [82], which is used to simulate the wind on the vehicle. It takes the world's wind velocity and calculates the force and torque exerted on a link. However, this plugin is unsuitable for the simulation because it generates forces on an airfoil as if it were a regular object, meaning it does not generate lift and drag forces.

The `hector_gazebo_plugin` simulates various sensors, such as GNSS receivers, IMU and magnetometers. This plugin did not require any modifications as it works properly and publishes the sensor data on ROS topics.

The plugins created for the development of this dissertation were: (i) the `land_yacht_plugin` that interfaces ROS with the wing sail flap and steering mechanism joints;

(*ii*) the `land_yacht_appar_wind` plugin that simulates the wind sensor by using the velocity and direction of the vehicle and of the wind. Plugins can be created directly on the ROS workspace or externally. In this case, they were created externally. In order to do this, a folder with a header file, a source code file, a `cmakelists` file and a corresponding build folder were created. The code was written in the source and the header files. The `cmakelists` file contains the compiling and linking instructions, such as what packages to use and their directories (in the case of the modified plugins, the ROS packages had to be specified). After this, a `cmake` command needs to be executed on the build folder, as shown in Listing 4.3 and, finally, a `make` command to build the plugin. The `make` command will create a `.so` file that needs to be moved to the ROS workspace, as shown in Listing 4.4, where the *make* and copy commands are applied. The `land_yacht_plugin` is composed of a ROS node that subscribes to the flap angle

Listing 4.3: Plugin Creation Cmake Example

```
/land_yacht_plugin/build$ cmake../
```

Listing 4.4: Plugin Creation Make Example

```
/land_yacht_plugin/build$ make && cp libland_yacht_plugin.so /home/catkin_ws/devel/lib/
```

and steering angle topics. When the plugin receives a callback from these topics it sets the joint target position to the broadcasted angular position. Apart from this, this plugin also publishes the vehicle ground truth for control and debugging.

The `land_yacht_appar_wind` plugin subscribes the wind topic and transforms the velocity and direction of the true wind into the apparent wind velocity and direction, using the vehicle world rotation and velocity, and uses a ROS node to publish the apparent wind velocity and direction.

Another set of plugins that could have been used for the development of this project are the `sitl_gazebo` plugins [83], which includes a wind and aerodynamics plugin. The plugins depend on the mavros package [84], making the simulation development more complex.

The plugins can be loaded in the model or the world file. An example of how to load a plugin, in this case the `liftdragplugin`, is presented in Listing 4.5. The parameters obtained from the NACA63(3)-018 airfoil characteristics were converted according to the example presented in Figure 4.4.

4.3 Control Development

A control system for both the flap and the steering was implemented in a ROS node called `land_yacht_control`, in the `land_yacht` ROS package. Furthermore, a manual control that overrides the autonomous control was also created. The flap control uses a Proportional - Differential (PD) controller and the steering control uses a simple Proportional (P) controller. The PD controller has a control loop that uses the current value as

Listing 4.5: Plugin SDF Code Example

```
<plugin name="sail_plugin" filename="libTese_LiftDragPlugin.so">
  <a0>0.0</a0>
  <cla>5.72957795131</cla>
  <cda>−0.57295779513</cda>
  <cma>−0.05</cma>
  <alpha_stall>0.17453292519</alpha_stall>
  <cla_stall>−2.86478897565</cla_stall>
  <cda_stall>2.29183118052</cda_stall>
  <cma_stall>0.2 </cma_stall>
  <cp>0.05 0 1.5</cp>
  <area>0.4</area>
  <fluid_density>1.2041</fluid_density>
  <forward>−1 0 0</forward>
  <upward>0 1 0</upward>
  <link_name>wing_sail</link_name>
  <radial_symmetry>true</radial_symmetry>
</plugin>
```

feedback and the difference between this value and a set value as the error. The proportional part is obtained by multiplying the error with a proportional gain constant. The differential part is calculated by subtracting the current and previous error and dividing the result by the elapsed time and then multiplying it by the differential gain constant. Both parts are added, producing a single value. The P controller is identical and includes solely the proportional component [85].

4.3.1 Manual Control

The `land_yacht_control` ROS node subscribes to the `teleop_key` node to read the keyboard inputs. If a specific key is pressed, then the autonomous control is turned off and the vehicle enters into manual mode. In manual mode, the keyboard inputs are used to turn the flap or the front wheels. Furthermore, pressing different keys will control manually only the steering or only the flap. In short, it is possible to control autonomously or manually both actuators as well as control one actuator manually and the other autonomously.

As a result, the system is composed of four states: full autonomous control, full manual control, manual flap control and manual steering control. All of these states are independent, meaning the operator can change to any state independently of what state the system is currently on. A flowchart of how the current state affects the control is presented in Figure 4.5.

4.3.2 Flap Control

The flap control ensures that the wing flap has an angular position such that the force generated by the wind induces torque on the wing sail. This torque shifts the wing sail direction from the wind direction to a certain angle of attack. The flap control system, described by the block diagram presented in Figure 4.6, has a desired angle of attack as an input. This input defaults at 10° and can be changed by the operator by

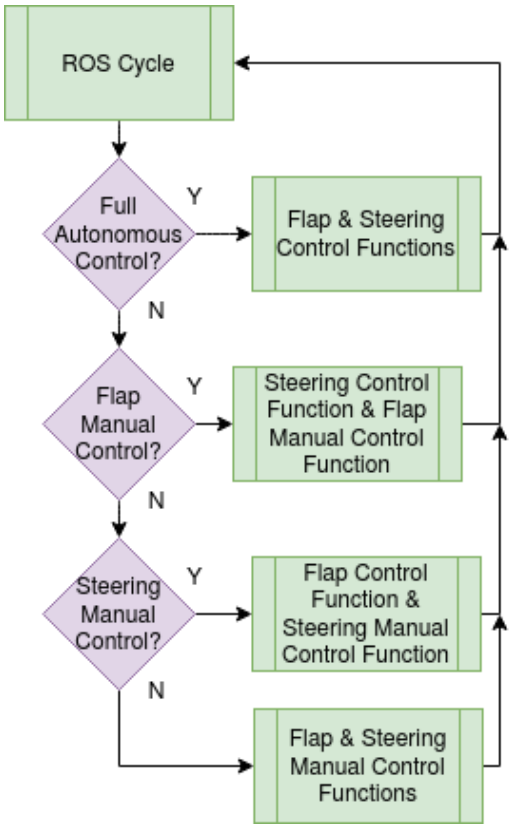


Figure 4.5: Control Function Selection Depending on the Current State

publishing it on a ROS topic. Every time there is a change of the wind direction and of the sail angular position, the control system recalculates the new flap angle. With these callbacks, the control system can calculate the current angle of attack and the error between it and the desired angle of attack, e. The system also calculates the differential error by calculating the difference between the current and the previous error, dividing the result by the time difference, de/dt. If the difference between the consecutive errors, prior to the difference of time multiplication, exceeds 20°, then the differential error is assumed to be 0 to avoid further control issues, as a higher magnitude differential error will result in substantial higher variations in the flap angle. Subsequently, the error e and the differential error de/dt are multiplied by the proportional and differential gains, respectively, and are added to create the controller output. This output value is summed to the last calculated flap angle to guarantee that the flap will stabilise at an angle where the angle of attack of the wing sail is as close as possible to the desired angle of attack. For example, if a certain flap angle produces insufficient torque at the wing sail, the controller will sum the controller output to the last flap angle, making it turn more. This slight change in the flap angle will be enough to generate enough force on the flap and, consequently, enough torque on the wing sail to achieve the desired angle of attack. If the calculated flap angle exceeds the actuator range, then it is set to a pre-set limit (in this case ±15°). The PD gains were obtained through trial and error. The flowchart that describes this control system is shown in Figure 4.7.

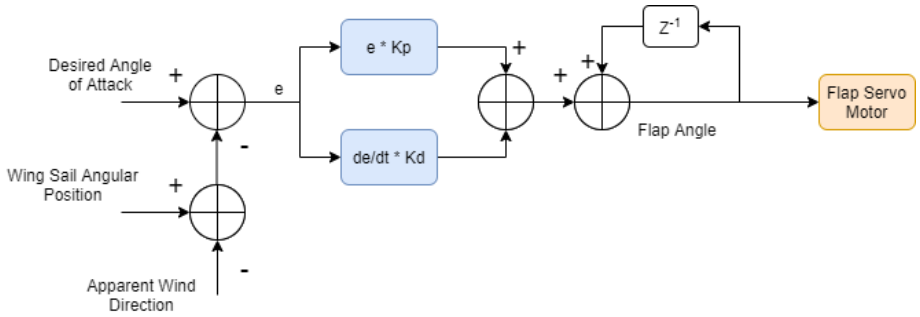


Figure 4.6: Flap Control Block Diagram

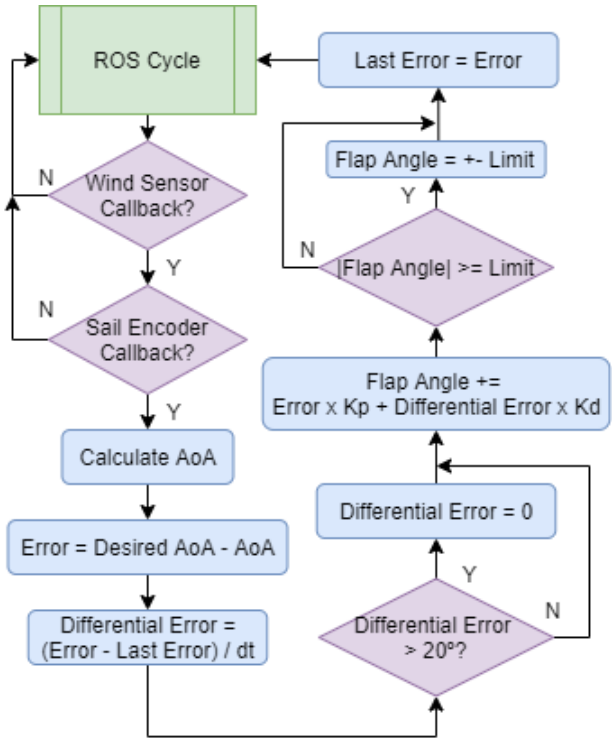


Figure 4.7: Flap Control Flowchart

4.3.3 Steering Control

Given a path, the steering control systems ensures that the front wheels turn towards the next waypoint. This system operates every time new vehicle's pose data is received. First, the system calculates the error to the next waypoint. If the vehicle is close, given a margin, to the waypoint, it starts moving to the next waypoint. However, if the vehicle is still distant to the waypoint, the desired vehicle direction (*Desired* θ) is calculated using the previous error and, subsequently, the error between the current vehicle direction and the desired vehicle direction (θ *Error*). After calculating this error, the steering angle is obtained by multiplying the error by the P gain, obtained through trial and error. If

the resulting angle exceeds the actuator range, then it is set to the pre-set limit. The kinematic model of the land yacht is presented in Figure 4.8, where θ and ϕ are the vehicle heading and steering orientation, respectively, and a flowchart describing the steering control is presented in Figure 4.9.

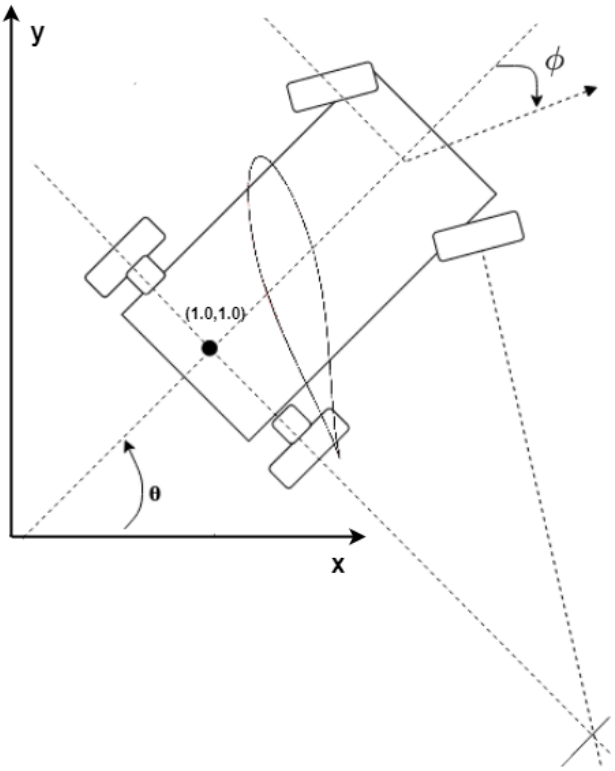


Figure 4.8: Land Yacht Kinematic Model

4.3.4 Maximum Velocity and Desired Angle of Attack Control

The land yacht needs not only to control the tail flap and the steering angle, but also to ensure stability when faced with strong winds. If the vehicle exceeds 12 m/s, the desired angle of attack will be set to 0°. When the flap achieves the desired angle of attack of 0°, there will be no force generated by the wind on the wing sail, making the vehicle decelerate. Once it reaches a velocity of 8 m/s, the desired angle of attack will return to its previous value and the vehicle will consequently reaccelerate. A flowchart describing the maximum velocity control is shown in Figure 4.10. Furthermore, if the apparent wind is between 180° and 360° (or −180° and 0°), the sign of the angle of attack changes. For example, if the desired angle of attack is 8°, and the apparent wind is −140°, the desired angle of attack will change to −8°. If this change in sign is not made, then the force generated on the vehicle will be negative and it will decelerate.

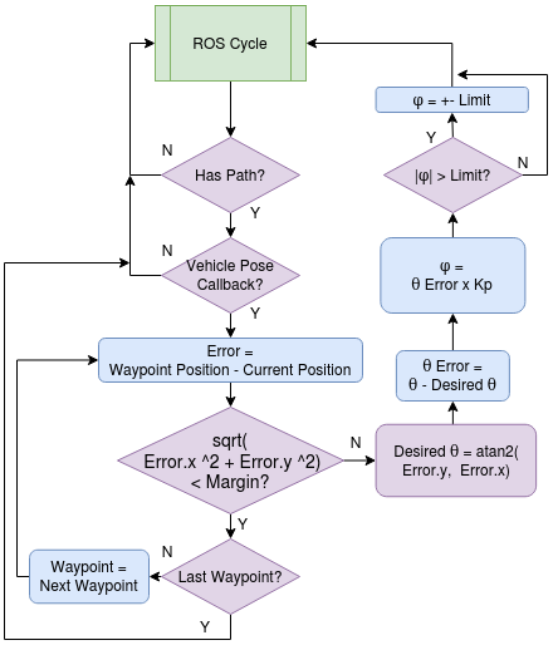


Figure 4.9: Steering Control Flowchart

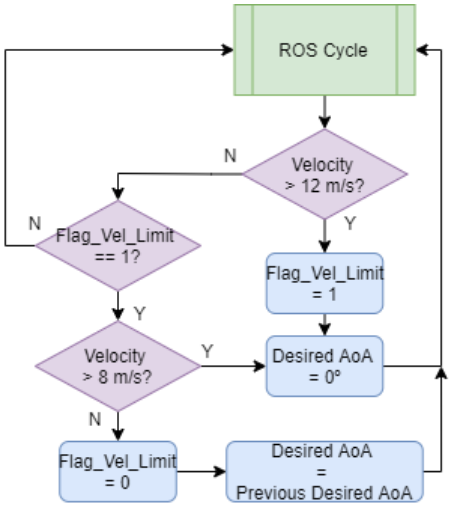


Figure 4.10: Maximum Velocity Control Flowchart

4.4 Conclusion

In this chapter, an analysis of different simulators was made, which culminated in the selection of the Gazebo environment. Subsequently, the simulation process was described. This description covered the building and importing of the vehicle 3D model to the Gazebo environment. The importing was done for each vehicle link, and connecting each link through joints. The ROS integration with Gazebo and the Gazebo plugins,

including the plugins developed specifically for this project, were also presented. Next, the system control, consisting of the flap control, the steering control and manual control, was described. The flap control uses a PD controller to turn the flap to an angle where it generates sufficient force to induce torque to the wing sail, making it turn. The steering control uses a P controller to turn the wheels towards the next waypoint. The manual control uses the keyboard to change the flap angle and the steering. Furthermore, a maximum velocity controller was implemented by setting the wing sail angle of attack to 0°, when the maximum velocity is reached, and setting it back to the previous value, when a minimum threshold velocity is reached.

With the development made in this chapter, the vehicle is ready to be simulated and tested.

Chapter 5

Tests, Results and Validation

In this chapter, the tests performed to validate and characterise the designed land yacht are presented, along with the results and respective analysis.

5.1 Experiment Deployment

Figure 5.1 presents the full set of nodes and topics involved in this experimental setup. The Gazebo environment (`gazebo` node) subscribes the wind, steering and flap topics and publishes the platform's ground truth as well as the sail's rotation and apparent wind topics. The control (`land_yacht_control` node) subscribes the ground truth, the sail rotation, the apparent wind from Gazebo and the manual key topics; it also publishes the flap and steering control topics. The `true_wind_generator` node is responsible for generating and publishing the true wind. The manual `teleop_key_pub` control node publishes the pressed key commands.

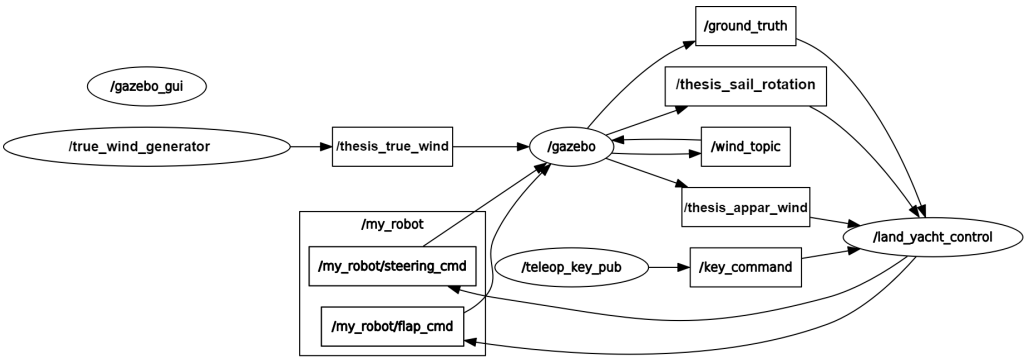


Figure 5.1: Experimental Setup

5.2 Wing Sail Control

The following tests were made to validate the wing flap control and the vehicle propulsion:

- Wing sail wind alignment test validates if the wing sail automatically aligns to the wind direction when flap and wing sail are lined up.
- Wing sail response to the flap angle test verifies if the wing sail turns to the opposite side of the angle between the flap and the wing sail.
- Wing sail rotation test validates if the wing sail assumes the angle of attack set by the operator.
- Platform motion tests validate if the vehicle is propelled according to the wind conditions and wing sail angle of attack.
- Polar diagram test characterises the behaviour of the platform with different wind conditions.

5.2.1 Wing Sail Alignment to the Wind

5.2.1.1 Setup

To test the wing sail alignment to the wind, the vehicle was positioned turned to east (heading of 90°). Then, a wind coming from north (0°) was applied to the simulation by subscribing the `thesis_true_wind` ROS topic, which publishes the current true wind velocity and direction. This generated an apparent wind of 90° relatively to the vehicle. Wind velocities of 8 kn, 9 kn and 10 kn were used in this test. This setup is presented in Figure 5.2a, where the blue arrow represents the wind direction and the dashed red line represents the wing sail orientation.

5.2.1.2 Results

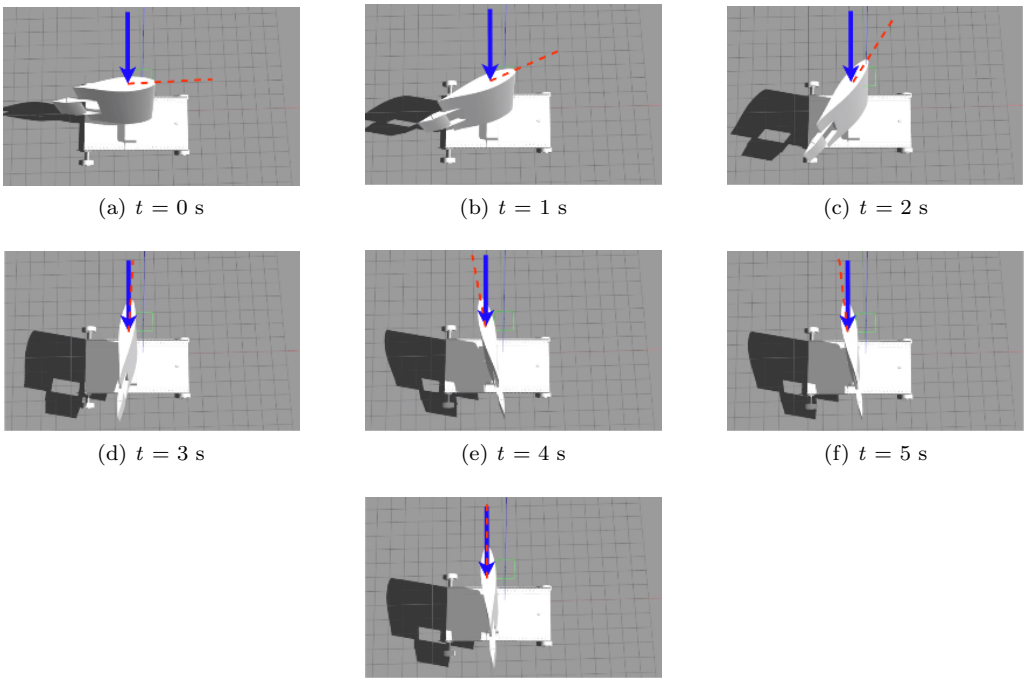
The wing sail started changing direction towards the wind direction. After reaching the wind direction, the wing sail overshot slightly due to inertia and, finally, aligned to the wind, as shown in Figure 5.2. Figure 5.3 plots the time response of the wing sail alignment with a 0° angle of attack for wind velocities of 8 kn, 9 kn and 10 kn in blue, red and green, respectively.

These results show that, when the wing and tail are lined up, the wing sail aligns automatically to the wind due to the pitching moment of the aerodynamics plugin. Figure 5.3 shows that, although the wing sail takes identical time to stabilise with different wind velocities, the higher the velocity, the higher the overshoot.

5.2.2 Wing Sail Response to the Flap Angle

5.2.2.1 Setup

To test the wing sail response to the flap angle, both wing and wind directions were set east. The wind velocity was set to 8 kn and kept constant throughout the test. The vehicle flap was turned to the left and to the right through the manual control, using the `teleop_key` ROS package and the Q and E keyboard keys.



(a) $t = 0$ s (b) $t = 1$ s (c) $t = 2$ s

(d) $t = 3$ s (e) $t = 4$ s (f) $t = 5$ s

(g) $t = 6$ s

Figure 5.2: Wing Sail Alignment to the Wind Over Time (10 kn)

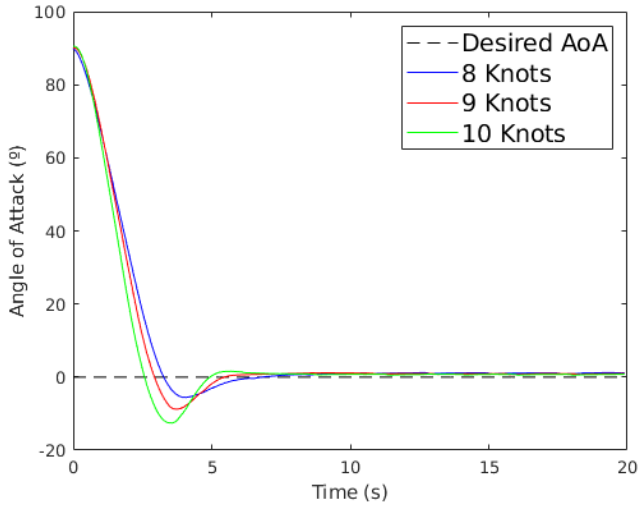
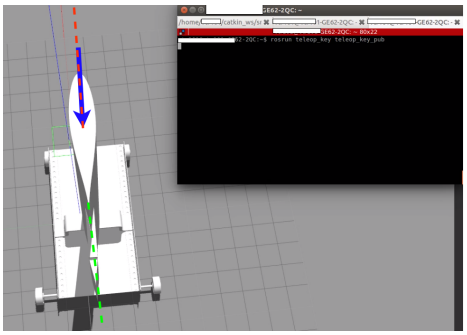


Figure 5.3: Wing Sail Angle of Attack with Wind Direction of 0° (No Control)

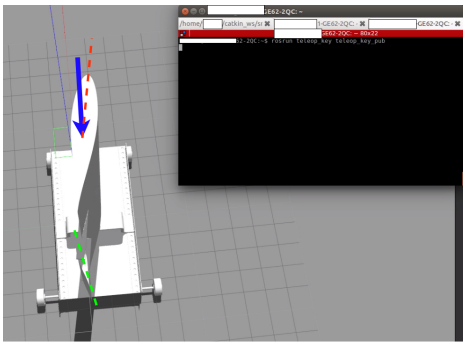
5.2.2.2 Results

Without actuating on the wing flap, the wing sail did not move, since it was already aligned with the wind. When the wing sail flap was turned to the right, the aerodynamic forces induced a torque on the wing sail, making it turn to the left. The wing sail turned

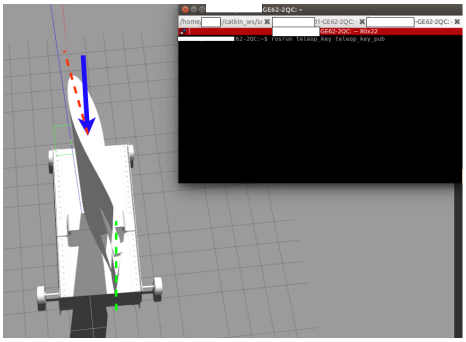
to the right when the flap was moved to the left. The response of the wing sail to the flap angle change is shown in Figure 5.4, where the blue arrow represents the wind direction, the red dashed line represents the wing sail direction and the green dashed line represents the flap direction. The wing sail responded correctly to the flap angle change.



(a) Flap Lined With the Wing Sail



(b) Flap Turned Right



(c) Flap Turned Left

Figure 5.4: Wing Sail Response to Flap Angle Changes (8 kn)

5.2.3 Setting of the Wing Sail to Different Angles of Attack

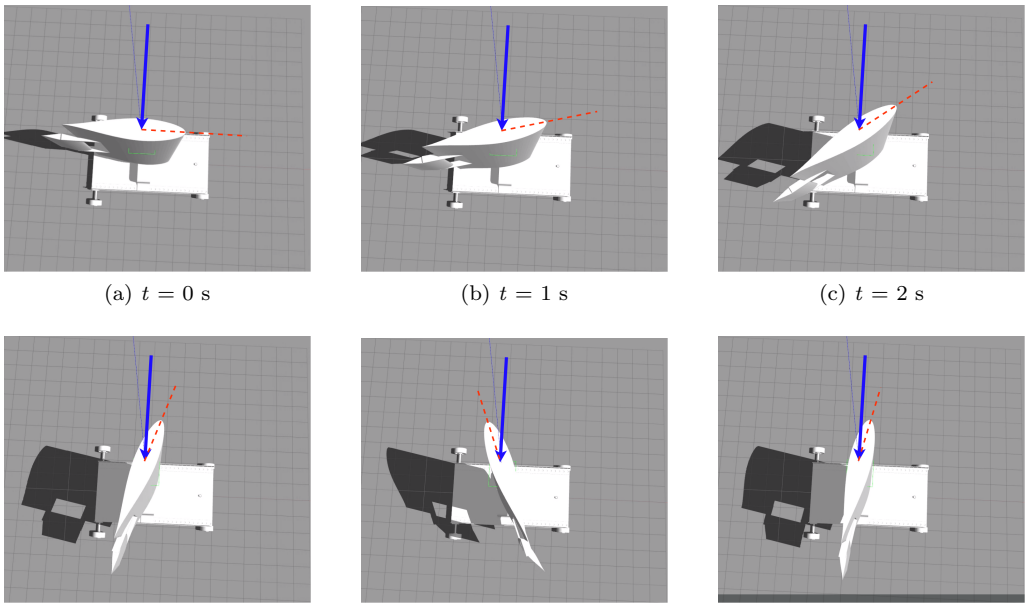
5.2.3.1 Setup

To perform this test the vehicle was headed east and the wind was defined by the `thesis_true_wind` ROS topic. First, the wind direction was set to 90° and the flap was controlled with two sets of PD gains: $K_p = 0.05$ and $K_d = 0.01$; $K_p = 0.05$ and $K_d = 0.07$. Finally, the test was repeated with an apparent wind direction of 45° and PD gains $K_p = 0.05$ and $K_d = 0.07$. Both tests were performed with wind velocities of 8 kn, 9 kn and 10 kn using a target angle of attack of 10°.

5.2.3.2 Results

Once a target angle of attack is set, the wing sail starts heading to the direction of the wind. As the error between the current and the desired angle of attack decreases, the control system starts stabilising the wing sail to the target angle of attack. Figure 5.5 displays the wing sail motion for a target angle of attack of 10°. Appendix B presents

additional plots with different wind velocities and angles of attack. The time the wing



(a) $t = 0$ s (b) $t = 1$ s (c) $t = 2$ s

(d) $t = 3$ s (e) $t = 4$ s (f) $t = 5$ s

Figure 5.5: Wing Sail Alignment With the Wind Over Time (10 kn, AoA = 10°)

takes to reach the desired angle of attack is identical to the time it takes to line up to the wind without flap control. However, it oscillates for a few more seconds as the control system determines the suitable flap angle for the desired wing sail angle of attack, as shown in Figures 5.6, 5.7 and 5.8.

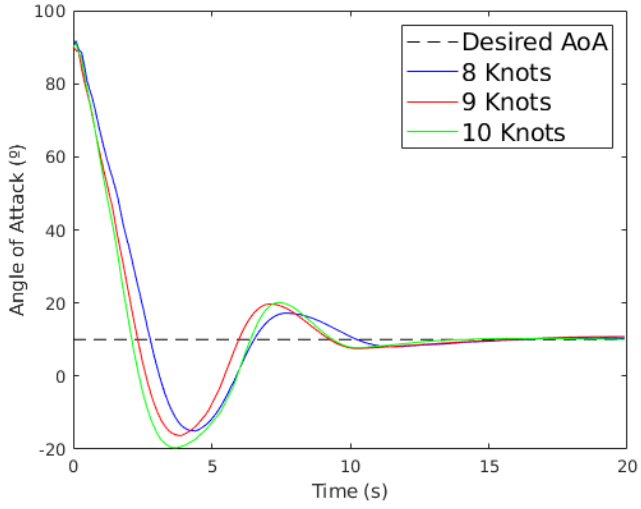


Figure 5.6: Wing Sail Angle of Attack with Wind Direction of 0° ($K_p = 0.05$, $K_d = 0.01$, AoA = 10°)

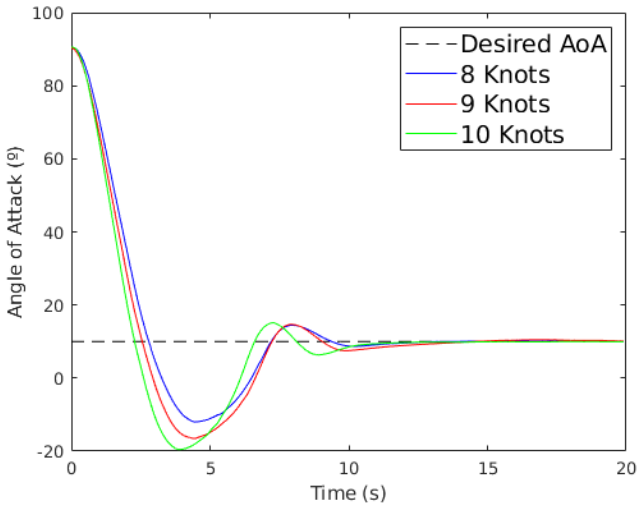


Figure 5.7: Wing Sail Angle of Attack with Wind Direction of 0° ($K_p = 0.05$, $K_d = 0.07$, AoA = 10°)

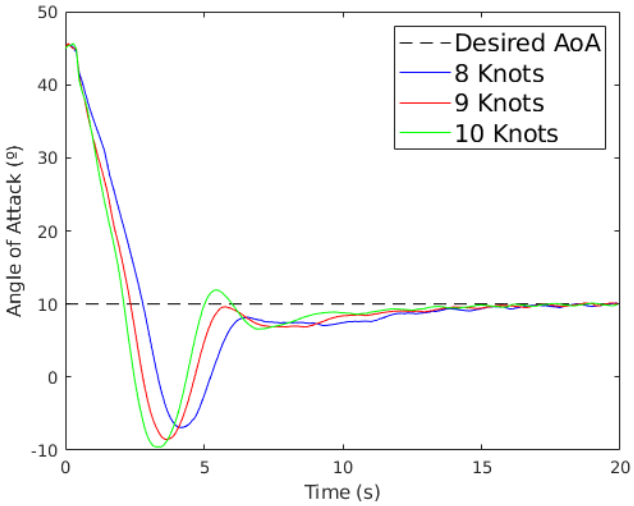


Figure 5.8: Wing Sail Angle of Attack with Wind Direction of 45° ($K_p = 0.05$, $K_d = 0.07$, AoA = 10°)

5.2.4 Vehicle Motion

5.2.4.1 Setup

The vehicle was positioned facing east, the wing sail desired angle of attack was set to 10°, the wind was set with a direction of 0° and velocities of 8 kn, 9 kn, 10 kn, 15 kn and 18 kn, while the flap controller PD gains were set to $K_p = 0.05$ and $K_d = 0.07$.

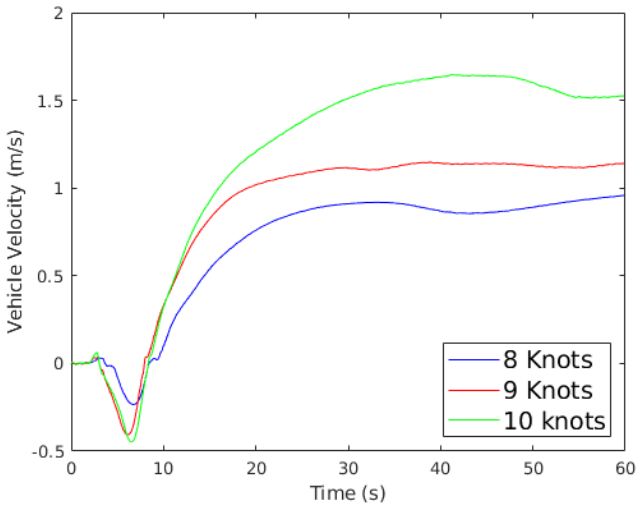


Figure 5.9: Vehicle Velocity Over Time ($K_p = 0.05$, $K_d = 0.07$, AoA = 10°)

5.2.4.2 Results

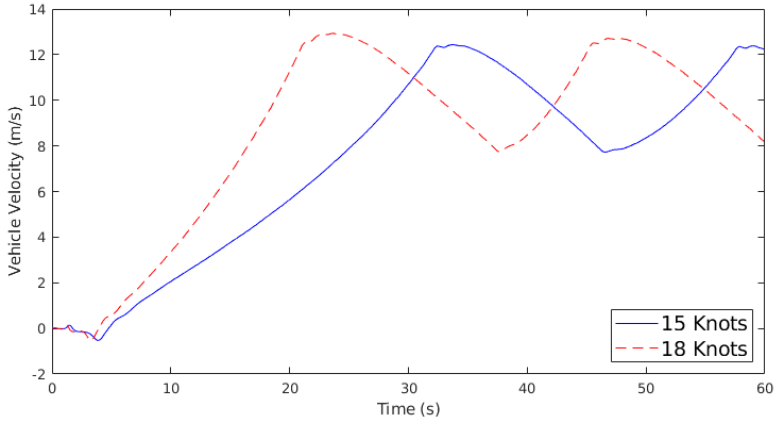
With wind velocities of 8 kn, 9 kn and 10 kn, the vehicle moved slightly backwards during the stabilisation of the wing sail. This is caused by the wing sail overshoot, which presents a negative angle of attack for a short period of time. Once the wing sail stabilised at the desired angle of attack, the vehicle started moving forward and accelerated until it reached a maximum velocity, as shown in Figure 5.9. With wind velocities of 15 kn and 18 kn, the vehicle accelerated until a maximum velocity of 12 m/s. When this upper threshold velocity was reached, the wing sail desired angle of attack changed to 0°, decelerating the vehicle for protection. When the vehicle velocity decreased to 8 m/s, the desired angle of attack was again set to 10°, accelerating the vehicle as shown in Figure 5.10. Figure 5.10a plots the vehicle velocity and Figure 5.10b plots the wing sail angle of attack over time. The angle of attack during the first seconds is not displayed in these plots since it was already shown in Figures 5.6, 5.7 and 5.8.

Although there was only an increase of 5 kn in the wind velocity relatively to the first test, the velocity curves presented in Figure 5.10a suggest the velocities are increasing exponentially and should be discussed as it can be a problem in the simulation. The tests showed that a minimum wind velocity of 8 kn is required to displace the vehicle. Furthermore, with wind velocities from 8 kn to 10 kn, the vehicle displays stable velocities of 0.9 m/s to 1.6 m/s and does not require the maximum velocity control. The vehicle becomes unstable when it reaches velocities above 15 m/s and the maximum velocity control was required to keep it stable.

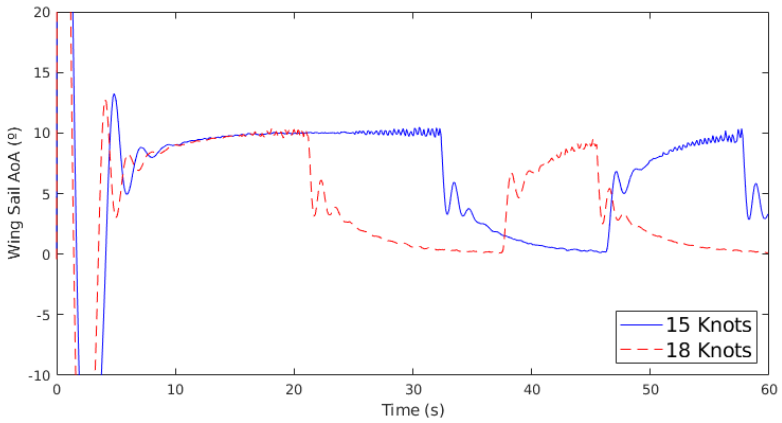
5.2.5 Polar Diagram

5.2.5.1 Setup

The polar diagram characterises the optimal performance of a wind propelled platform as a function of the wind velocity and apparent wind angle. In this test, the autonomous



(a) Vehicle Velocity Over Time



(b) Wing Sail Angle of Attack Over Time

Figure 5.10: Vehicle Velocity and Wing Sail AoA Over Time ($K_p = 0.05$, $K_d = 0.07$, AoA = 10°)

flap control was responsible for the optimisation of the lift force applied to the wing sail. The flap controller PD gains were set to $K_p = 0.05$ and $K_d = 0.07$. Three different wind velocities were simulated (8 kn, 9 kn and 10 kn) together with apparent wind direction increments of 5° from 0° to ±180°.

5.2.5.2 Results

The maximum velocity was null with low apparent wind directions. The velocity started increasing at approximately ±45° until peaked at ±90°. After peaking, the maximum velocity started decreasing until it reached null values near ±180°, as shown in the polar diagram presented in Figure 5.11. The no go zone is visible between −45° and 45°, and between −135° and 135°.

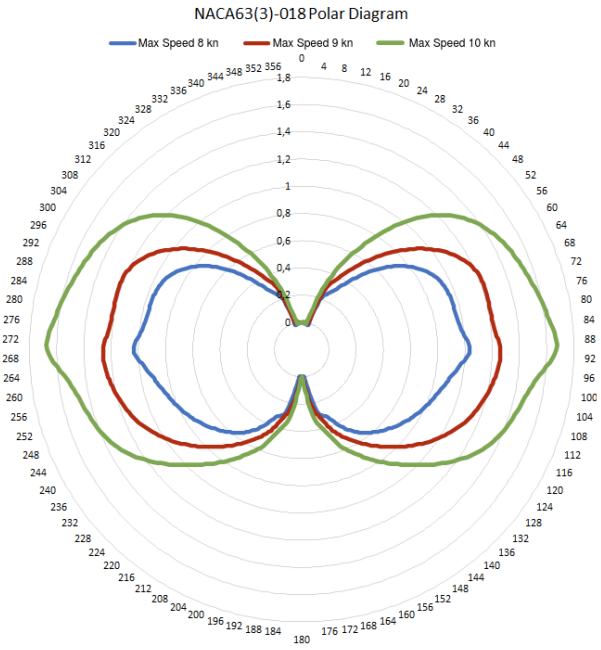


Figure 5.11: Land Yacht Polar Diagram

5.3 Steering Control

After validating the flap control and the vehicle propulsion, the steering control was subjected to the:

- Steering test to verify if the vehicle changes the direction properly by applying 90° turns.
- Path following test to confirm if the vehicle follows a course composed of a set of waypoints.

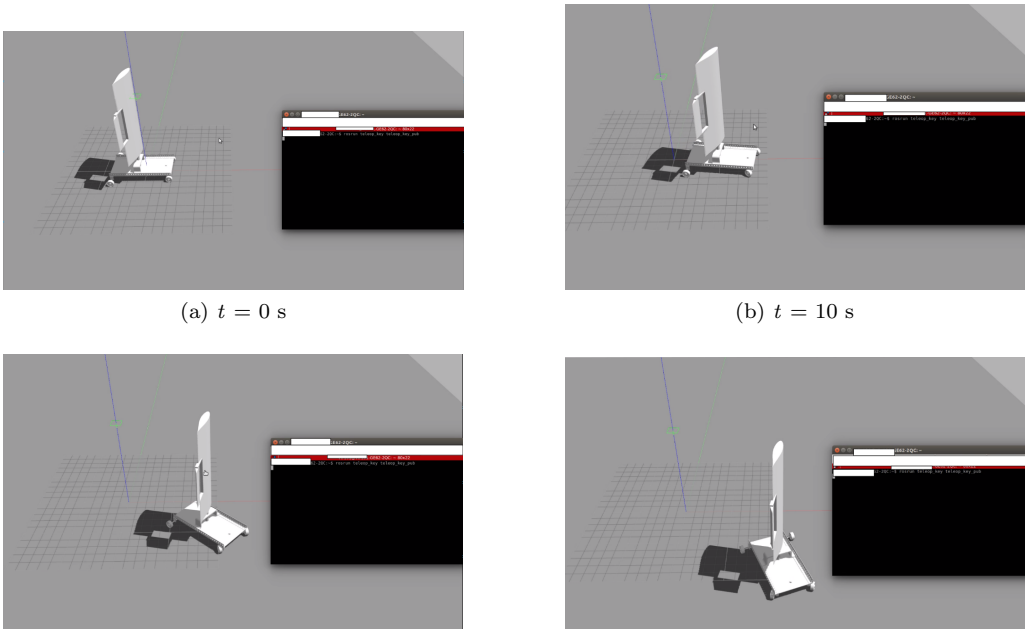
5.3.1 Vehicle Steering

5.3.1.1 Setup

The vehicle was positioned facing east and the wind direction set to 0°. The applied wind velocities were 10 kn, 14 kn and 16 kn, while the desired angle of attack was set to 10° and the flap controller PD gains were set to $K_p = 0.05$ and $K_d = 0.07$. This test was done with manual control with the help of the `teleop_key` ROS package. The vehicle accelerates for a period of 10 s before the steering mechanism dictates a 90° turn, changing the apparent wind from 90° to 0° and placing the vehicle in the no go zone, as shown in the polar diagram presented in Figure 5.11.

5.3.1.2 Results

Figure 5.12 illustrates the right turning, with a wind velocity of 10 kn and an initial apparent wind direction of 90°. Since the apparent wind direction remains unchanged



(a) $t = 0$ s (b) $t = 10$ s

(c) $t = 20$ s (d) $t = 30$ s

Figure 5.12: Land Yacht Turning Right

for all velocities, the vehicle stops after turning 90° left or right. Figure 5.13 displays in blue, red and green the results with wind velocities of 10 kn, 14 kn and 16 kn, respectively. Furthermore, the dashed lines correspond to the right turns, whereas the solid lines correspond to the left turns. This plot shows that the vehicle does not turn to the left

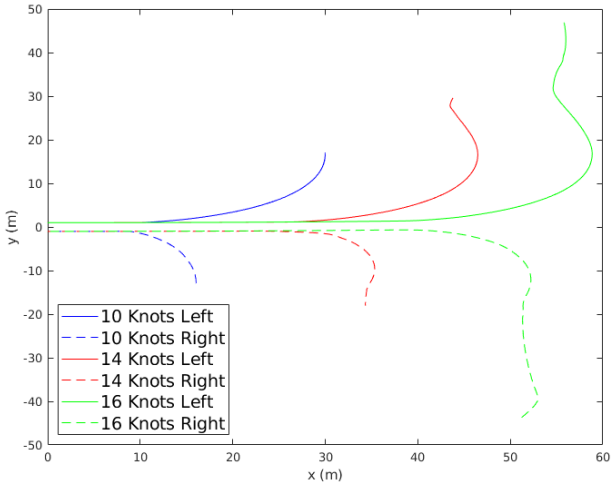


Figure 5.13: Land Yacht Turning Left and Right Graph

side the same way it turns to the right. This may result from an error in the steering

mechanism 3D mesh or in the Gazebo joints. The vehicle did not turn left and right perfectly due to the human manual steering.

5.3.2 Path Following

5.3.2.1 Setup

Initially, the vehicle was positioned facing east at the position (0.0,0.0) and the wind was set to a velocity of 12 kn and direction of 0°. The flap controller PD gains were set to $K_p = 0.05$ and $K_d = 0.07$, and the steering controller P gain was set to $K_p = 1$. This test included two paths: an open slalom and a closed circuit.

5.3.2.2 Results

The vehicle reached successfully all waypoints in both tests. In the first test, presented in Figure 5.14, even though the vehicle reached the waypoints, the distance to the last one was larger. This is because of the sub optimal steering to the left side. In the second test, presented in Figure 5.15, the vehicle passed beyond the last waypoint, continuing for a few more meters. This indicates the need to refine the control which decides when to set the angle of attack to 0°, taking into account the current wind velocity.

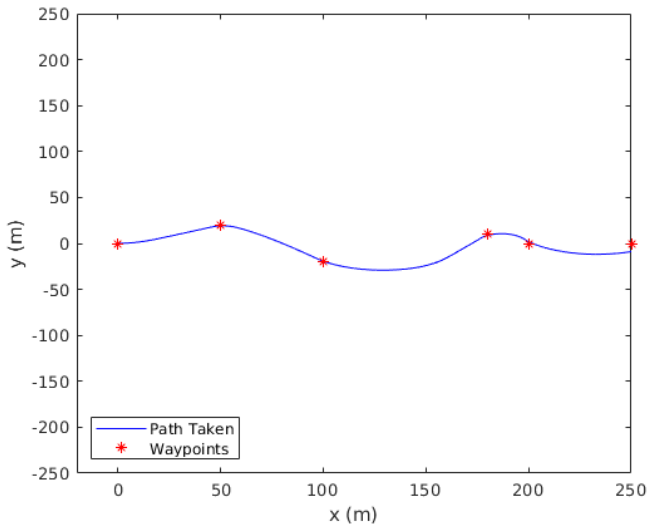


Figure 5.14: Land Yacht First Path

5.4 Conclusion

This chapter summarises the tests made to validate the model and control of the designed land yacht control.

Regarding the model, it appears to have a problem in the steering mechanism affecting left turns when compared to right turns and requires wind velocities of at least

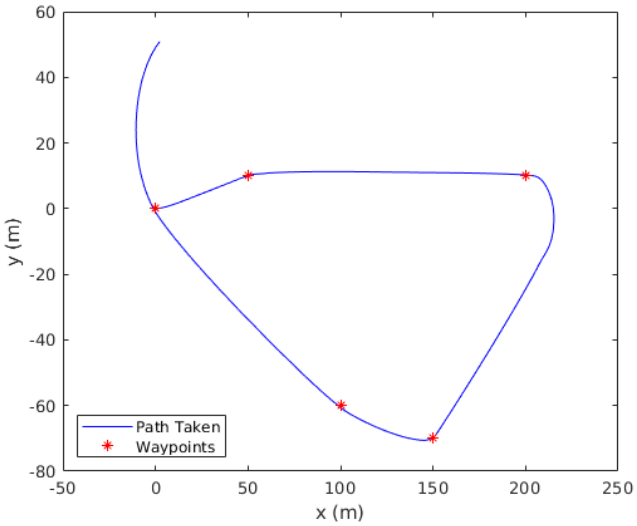


Figure 5.15: Land Yacht Second Path

8 kn to move. This may be caused by incorrect values in the Gazebo joints. However, experiments with different joint friction and damping values were unable to fix this issue.

Considering the control, while the wing sail flap control works properly, it is possible that there are better PD value gains, leading to a smoother wing sail stabilisation. The wing sail stabilisation showed a negative angle of attack for a short period of time (approximately 2.0 s), causing the vehicle to slightly move backwards (less than 1.5 m). The platform achieves stable velocities for wind from 8 kn to 10 kn, but, for wind velocities greater than 12 kn, the platform accelerates beyond its stability limit of 12 m/s. When this occurs, the wing sail angle of attack has to be set to 0° to decelerate the land yacht. Finally, the motion sensor plugins were not tested as all the tests were done using ground truth.

Chapter 6

Conclusions

6.1 Outcomes

This dissertation addresses the design and simulation of a small autonomous land yacht. This wind propelled platform constitutes a solution to the problem of power autonomy by using a rigid wing sail instead of an electrical or fuel powered motor. The proposed platform is a four wheeled land yacht with steering control and a free rotating wing sail, controlled by a tail flap. The model of the wing sail (based on a symmetrical NACA63(3)-018 profile) and chassis were designed in Fusion 360 and imported to Gazebo. The platform has a width of 360 mm and a length of 700 mm with a maximum payload of 19 kg, whereas the wing sail has an area of $0.4\,m^2$ with a chord of 400 mm and a wingspan of 1000 mm. The platform control comprises the flap controller, which actuates the wing sail based on the current and desired angles of attack, and the steering, that controls the direction based on the current and the desired platform pose. The experimental deployment was performed using ROS and the Gazebo simulator together with two open source plugins (aerodynamics and sensors) and two custom made plugins (apparent wind and platform control). The platform operates with stable velocities, of 0.9 m/s to 1.6 m/s for wind velocities of 8 kn to 10 kn, and requires a maximum velocity control for wind velocities above 12 kn. The world wind generation as well as the manual and autonomous platform control were implemented as ROS nodes.

6.2 Difficulties

During the development of this dissertation several difficulties were encountered. Given the COVID-19 pandemic and social distancing, there was few guidance regarding the Gazebo simulator. Even though there was some familiarity with the simulator, this development required further in depth study and several trial and error attempts, given the few documentation available on advanced Gazebo programming.

6.3 Future Work

The future work suggestions include:

- Optimisation of the navigation performance by ensuring the wheels turn equally to both sides.
- Power autonomy through the addition of a solar panel and a battery in order to simulate component power consumption.
- Exploring different world scenarios, such as windy snowy planes or tidal zones, where a land yacht would prevail over a conventional land rover, to validate its utility.
- Development of a non linear estimator, such as an Extended Kalman Filter or Particle Filter, to estimate the vehicle state namely its pose and velocity using sensor data.
- Build, develop and assess the real life version of the designed land yacht.

6.4 Conclusion

A solution which meets the objectives stated in Chapter 1 was successfully designed and simulated. However, the tests showed that the steering mechanism requires further work as the left turning is sub optimal when compared to the right turning. The simulation also requires further work as the tests suggested that the vehicle continued to gain velocity exponentially for wind velocities above 15 kn. The flap control was able to rotate the wing sail to the specified angle of attack within seconds at various wind velocities, making this solution optimal for a wing sail equipped platform, such as a land yacht or a sail yacht.

Although the original objective of the dissertation had to be changed to a simulated environment, this dissertation helped to understand the behaviour of an airfoil, which is identical for land, water and aerial wind powered platforms equipped with rigid wings.

The development of this dissertation contributed with the creation of a land yacht vehicle model, with a simulated environment for wind powered land vehicles and with the creation and enhancement of Gazebo plugins that can be used, not only in land yachts, but also in wind powered aquatic vehicle simulations.

Furthermore, due to social distancing and house confinement during the pandemic, this dissertation helped to improve soft skills, such as organisation and time management, which proved to be crucial.

Bibliography

[1] E. N. Jacobs, K. E. Ward, and R. M. Pinkerton, "The characteristics of 78 related airfoil section from tests in the variable-density wind tunnel," Tech. Rep. 460, National Advisory Committee for Aeronautics, 1933. [citado na p. v, 7, 29]

[2] Blokart, "Blokart - fast, fun, compact." `https://blokart.com`, n/a. Accessed: 2019-12-27. [citado na p. v, 6, 7, 8]

[3] Ecotricity, "About the greenbird." `http://www.greenbird.co.uk/about-the-greenbird`, n/a. Accessed: 2020-01-06. [citado na p. v, 8]

[4] Ecofriend, "Best land yachts designed to offer a green ride." `https://ecofriend.com/land-yachts-designed-offer-green-ride.html`, n/a. Accessed: 2020-01-19. [citado na p. v, 7, 9]

[5] Whike, "About the whike." `http://whike.com/en/pagina/25/about-the-whike`, n/a. Accessed: 2020-01-25. [citado na p. v, 9]

[6] N.-L. Bacau, L. Kádár, K. Juhandi, A. Zhu, C. Beer, and M. Orlov, "Sail Car." `http://www.eps2019-wiki1.dee.isep.ipp.pt/doku.php?id=report`, 2019. Accessed: 2020-01-25. [citado na p. v, 9, 10]

[7] G. A. Landis, S. R. Oleson, and D. Grantier, "Zephyr: A landsailing rover for venus," in *Proceedings of the 65th International Astronautical Congress*, pp. 1–14, International Astronautical Federation, 2014. [citado na p. v, 3, 10, 11]

[8] J. Jouffroy, "A control strategy for steering an autonomous surface sailing vehicle in a tacking maneuver," in *2009 IEEE International Conference on Systems, Man and Cybernetics*, pp. 2391–2396, IEEE, 2009. [citado na p. v, 10, 11]

[9] S. Xie, J. Chen, H. Li, J. Luo, H. Pu, and Y. Peng, "The research on wing sail of a land-yacht robot," *Advances in Mechanical Engineering*, vol. 7, no. 12, p. 1687814015623144, 2015. [citado na p. v, 1, 10, 11, 12, 24, 25]

[10] J. Chen, Z. Ye, R. Yang, G. Cai, J. Li, and H. Li, "Design and control of multiple wing-sail land yacht robot," in *2018 IEEE International Conference on Mechatronics and Automation (ICMA)*, pp. 1800–1805, Aug 2018. [citado na p. v, 12, 13, 24, 25]

[11] Y. Dong, X. Ding, Z. Li, L. Zhang, H. Liu, N. Ding, Z. Sun, and H. Qian, "Wing sail land-yacht modeling and system verification," in *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1350–1355, Dec 2019. [citado na p. v, 13, 14, 24, 29]

[12] P. A. Mirzaei and M. Rad, "Toward design and fabrication of wind-driven vehicles: Procedure to optimize the threshold of driving forces," *Applied Mathematical Modelling*, vol. 37, no. 1, pp. 50–61, 2013. [citado na p. v, 13, 14]

[13] C. Tretow, "Design of a free-rotating wing sail for an autonomous sailboat," 2017. [citado na p. v, 15]

[14] M. F. Silva, B. Malheiro, P. Guedes, and P. Ferreira, "Airfoil selection and wingsail design for an autonomous sailboat," in *Robot 2019: Fourth Iberian Robotics Conference* (M. F. Silva, J. Luís Lima, L. P. Reis, A. Sanfeliu, and D. Tardioli, eds.), (Cham), pp. 305–316, Springer International Publishing, 2020. [citado na p. v, 31, 32]

[15] Open Source Robotics Foundation, "Aerodynamics." `http://gazebosim.org/tutorials?tut=aerodynamics&cat=physics`, 2014. Accessed: 2020-06-29. [citado na p. v, 40, 45]

[16] F. M. Noori, D. Portugal, R. P. Rocha, and M. S. Couceiro, "On 3d simulators for multi-robot systems in ros: Morse or gazebo?," in *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, pp. 19–24, IEEE, 2017. [citado na p. vii, 39, 40, 41]

[17] S. Xie, K. Feng, Y. Peng, J. Luo, J. Chen, and J. Gu, "Design and analysis of an autonomous controlled four wheeled land yacht," *2014 IEEE International Conference on Information and Automation, ICIA 2014*, pp. 773–778, 10 2014. [citado na p. 1]

[18] S. Schmitz, *Aerodynamic of Wind Turbines - A Physical Basis for Analysis & Design.* John Wiley & Sons, 2020. [citado na p. 3]

[19] H. Erckens, G.-A. Busser, C. Pradalier, and R. Siegwart, "Avalon," *IEEE Robotics Automation Magazine*, vol. 17, no. 1, pp. 45–54, 2010. [citado na p. 3]

[20] G. Reina and M. Foglia, "Modelling and handling dynamics of a wind-driven vehicle," *Vehicle System Dynamics*, vol. 57, no. 5, pp. 697–720, 2019. [citado na p. 5]

[21] Wikiwand, "Land sailing." `https://www.wikiwand.com/en/Land_sailing`, n/a. Accessed: 2019-12-27. [citado na p. 5, 6]

[22] International Land and Sandyachting Federation, "Welcome to the homepages of fisly." `https://www.fisly.org/`, n/a. Accessed: 2019-12-25. [citado na p. 5, 6]

[23] North American Land Sailing Association, "North american land sailing association." `http://www.nalsa.org`, n/a. Accessed: 2019-12-27. [citado na p. 6]

[24] International Land and Sandyachting Federation, "Rules for the european and world championships." `https://www.fisly.org/rules/isrr_app.pdf`, n/a. Accessed: 2019-12-25. [citado na p. 6]

[25] British Federation of Sand and Land Yacht Clubs, "British landsailing." `http://www.britishlandsailing.org.uk/land-yachts.html`, 2018. Accessed: 2020-01-08. [citado na p. 6]

[26] Sea Breeze, "Best land yachts designed to offer a green ride." `https://www.seabreeze.com.au/Articles/Land+Sailing/Original-Pacific-Magic-Plans-by-Paul-Day_1514690.aspx`, n/a. Accessed: 2020-01-19. [citado na p. 9]

[27] A. Zhu, C. Beer, K. Juhandi, M. Orlov, N. Bacau, L. Kádár, A. J. Duarte, B. Malheiro, J. Justo, M. F. Silva, M. C. Ribeiro, P. D. Ferreira, and P. Guedes, "Sail Car—An EPS@ISEP 2019 Project," in *2020 IEEE Global Engineering Education Conference (EDUCON)*, pp. 487–492, 2020. [citado na p. 10]

[28] K. M and M. Rad, "Comparison final velocity for land yacht with a rigid wing and cloth sail," *Lecture Notes in Engineering and Computer Science*, vol. 22, 07 2008. [citado na p. 14]

[29] J. Less'ard-Springett, A. Friebe, and M. Le Gallic, "Voter based control system for collision avoidance and sailboat navigation," in *Robotic Sailing 2017*, pp. 57–68, Springer, 2018. [citado na p. 15]

[30] A. Friebe, M. Olsson, M. Le Gallic, J. L. Springett, K. Dahl, and M. Waller, "A marine research asv utilizing wind and solar power," in *OCEANS 2017-Aberdeen*, pp. 1–7, IEEE, 2017. [citado na p. 15]

[31] Uginox, "The benefits of stainless steel." `https://www.uginox.com/en/all-about-stainless-steel/`, n/a. Accessed: 2020-02-19. [citado na p. 15]

[32] M. Doshi, "Advantages of stainless steel." `https://www.quora.com/What-are-the-advantages-and-disadvantages-of-stainless-steel-2`, 2018. Accessed: 2020-02-19. [citado na p. 15]

[33] Aalco - Ferrous and Non-Ferrous Metals Stockist, "Aluminium - specifications, properties, classifications and classes." `https://www.azom.com/article.aspx?ArticleID=2863`, 2005. Accessed: 2020-02-19. [citado na p. 15, 16]

[34] Norsk Hydro ASA, "Why use aliminium?." `https://www.hydro.com/pt-PT/sobre-o-aluminio/porque-usar-aluminio/`, n/a. Accessed: 2020-02-19. [citado na p. 16]

[35] O. J. Woodman, "An introduction to inertial navigation," tech. rep., University of Cambridge, Computer Laboratory, 2007. [citado na p. 16]

[36] Sparton NavEx, "What is an imu?." `https://www.spartonnavex.com/imu/`, 2015. Accessed: 2020-02-19. [citado na p. 16]

[37] Sbg-systems, "Ellipse micro." `https://www.sbg-systems.com/products/inertial-measurement-unit-imu-sensor/`, n/a. Accessed: 2020-02-19. [citado na p. 16]

[38] Xsens Technologies B.V., "Mti 600 series." `https://www.xsens.com/products/mti-600-series`, n/a. Accessed: 2020-02-19. [citado na p. 16]

[39] InterSense, LLC, "Navchip family." https://www.intersense.com/navchip, n/a. Accessed: 2020-02-19. [citado na p. 16]

[40] KVH Industries, Inc, "Highly accurate inertial measurement units (imus)." https://www.kvh.com/fog-and-inertial-systems/inertial-measurement-units, n/a. Accessed: 2020-02-19. [citado na p. 16]

[41] ST, "Lsm6dsox." https://www.st.com/content/st_com/en/products/mems-and-sensors/inemo-inertial-modules/lsm6dsox.html, n/a. Accessed: 2020-02-19. [citado na p. 16]

[42] TDK Corporation, "Imu (inertial measurement unit)." https://product.tdk.com/info/en/products/sensor/mortion-inertial/imu/index.html, n/a. Accessed: 2020-02-19. [citado na p. 16]

[43] Bosch-sensortec, "Smart sensor bmf055." https://www.bosch-sensortec.com/products/smart-sensors/bmf055.html, n/a. Accessed: 2020-02-19. [citado na p. 16, 29]

[44] European Global Navigation Satellite Systems Agency, "What is gnss?." https://www.gsa.europa.eu/european-gnss/what-gnss, 2017. Accessed: 2020-02-19. [citado na p. 16]

[45] M. Venezia, "What is the difference between gnss and gps?." https://www.semiconductorstore.com/blog/2015/What-is-the-Difference-Between-GNSS-and-GPS/1550/, 2015. Accessed: 2020-02-19. [citado na p. 16]

[46] U-blox, "Neo-m8 series." https://www.u-blox.com/en/product/neo-m8-series, n/a. Accessed: 2020-02-19. [citado na p. 16, 29]

[47] Adafruit Industries, "Adafruit ultimate gps." https://www.adafruit.com/product/746, n/a. Accessed: 2020-02-20. [citado na p. 16]

[48] Linx Technologies, "Gm series gnss receiver module." https://linxtechnologies.com/wp/product/gm-series-gnss-receiver-module/, n/a. Accessed: 2020-02-20. [citado na p. 16]

[49] Society of Robots, "Gm series gnss receiver module." http://www.societyofrobots.com/sensors_encoder.shtml, 2014. Accessed: 2020-02-15. [citado na p. 16, 18]

[50] CUI Devices, "Amt21 series." https://www.cuidevices.com/product/motion/rotary-encoders/absolute/modular/amt21-series, n/a. Accessed: 2020-02-19. [citado na p. 18]

[51] Bourns, Inc., "Eaw - absolute contacting encoder (ace)." https://www.bourns.com/pdfs/ace.pdf, n/a. Accessed: 2020-02-17. [citado na p. 18, 29]

[52] Industrial-needs, "Wind speed sensor pce-fst-200-201." https://www.industrial-needs.com/technical-data/wind-speed-sensor-pce-fst-200-201.htm, n/a. Accessed: 2020-01-20. [citado na p. 18, 30]

[53] Pce-instruments, "Air flow meter pce-fst-200-202-i." https://www.pce-instruments.com/english/?id=5e7b11f269435088&_baseurl=%2Fenglish&action=ShowItem&_list=qr.art&_listpos=2&_artnr=60344, n/a. Accessed: 2020-02-20. [citado na p. 18, 19, 30]

[54] R. M. Young Company, "Mechanical wind sensors." http://www.youngusa.com/products/7/5.html, n/a. Accessed: 2020-02-20. [citado na p. 19]

[55] Vaisala, "Wind sensor wm30." https://www.vaisala.com/en/products/instruments-sensors-and-other-measurement-devices/weather-stations-and-sensors/wm30, n/a. Accessed: 2020-01-20. [citado na p. 19]

[56] Princeton University, "Hobby servo fundamentals." https://www.princeton.edu/~mae412/TEXT/NTRAK2002/292-302.pdf, n/a. Accessed: 2020-02-22. [citado na p. 19]

[57] Sparkfun Electronics, "Servo - hitec hs-805bb." https://www.sparkfun.com/products/11881, n/a. Accessed: 2020-02-26. [citado na p. 20]

[58] Dfrobot, "Df metal geared 15kg standard servo 270° (dss-m15s)." https://www.dfrobot.com/product-1177.html, n/a. Accessed: 2020-02-26. [citado na p. 20]

[59] Dfrobot, "9g 180° micro servo." https://www.dfrobot.com/product-255.html, n/a. Accessed: 2020-02-26. [citado na p. 20, 29]

[60] Dfrobot, "Dss-p05 standard servo (5kg)." https://www.dfrobot.com/product-236.html, n/a. Accessed: 2020-02-26. [citado na p. 20, 29]

[61] ST, "Stm32 32-bit arm cortex mcus." https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html#overview, n/a. Accessed: 2020-02-17. [citado na p. 22]

[62] V. Beal, "Sbc - single-board computer." https://www.webopedia.com/TERM/S/sbc_single_board_computer.html, n/a. Accessed: 2020-02-17. [citado na p. 22]

[63] Ubuntu, "Sbc - single-board computer." https://ubuntu.com/download/raspberry-pi-2-3-core, n/a. Accessed: 2020-02-17. [citado na p. 22]

[64] Open Source Robotics Foundation, "Ros distributions." http://wiki.ros.org/Distributions, n/a. Accessed: 2020-02-17. [citado na p. 22]

[65] Seralo, "Raspberrypi models comparison." https://www.socialcompare.com/en/comparison/raspberrypi-models-comparison, 2019. Accessed: 2020-02-25. [citado na p. 22, 30]

[66] Battery University, "Bu-706: Summary of do's and don'ts." https://batteryuniversity.com/learn/article/do_and_dont_battery_table, 2018. Accessed: 2020-02-23. [citado na p. 22]

[67] H. Moghbelli, K. Ellithy, Z. Eslami, R. Vartanian, D. Wannous, A. Ghamrawy, O. Basha, A. Fayad, M. Qaraqe, and S. Nicola, "Investigation of solar energy applications with design and implementation of photovoltaic traffic light signal system for qatar," *Renewable Energy and Power Quality Journal*, vol. 1, 04 2009. [citado na p. 22]

[68] Seeed Studio Inc., "2 km long range RF link kits with encoder and decoder." https://www.seeedstudio.com/2KM-Long-Range-RF-link-kits-with-encoder-and-decoder-p-321.htmll, n/a. Accessed: 2020-02-19. [citado na p. 22]

[69] Dfrobot, "Apc802 radio communication module (2800 meters)." https://www.dfrobot.com/product-375.html, n/a. Accessed: 2020-02-19. [citado na p. 24]

[70] Electronic Communications Committee, "The european table of frequency allocations and applications in the frequency range 8.3 kHz to 3000 GHz." https://www.ecodocdb.dk/download/2ca5fcbd-4090/ERCREP025.pdf, 2019. Accessed: 2020-02-19. [citado na p. 30]

[71] Avstop, "Forces on an airfoil." http://avstop.com/ac/flighttrainghandbook/forcesonanairfoil.html, n/a. Accessed: 2020-03-20. [citado na p. 31]

[72] Coppelia Robotics, "Coppelia robotics." https://www.coppeliarobotics.com, n/a. Accessed: 2020-05-19. [citado na p. 39]

[73] Open Source Robotics Foundation, "Gazebo. robot simulation made easy.." http://gazebosim.org, n/a. Accessed: 2020-05-19. [citado na p. 39, 40]

[74] ONERA, "What is morse?." https://www.openrobots.org/morse/doc/stable/what_is_morse.html, n/a. Accessed: 2020-05-19. [citado na p. 39, 40]

[75] Cyberbotics Ltd., "Webots: robot simulator." https://cyberbotics.com, n/a. Accessed: 2020-05-19. [citado na p. 39, 40]

[76] S. Carpin, M. Lewis, J. Wang, S. Balakirsky, and C. Scrapper, "Usarsim: a robot simulator for research and education," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 1400–1405, 2007. [citado na p. 39, 41]

[77] R. Smith, "Open dynamics engine." http://www.ode.org/index.html, n/a. Accessed: 2020-05-19. [citado na p. 39]

[78] Pybullet, "Bullet real-time physics simulation." https://pybullet.org/wordpress/, n/a. Accessed: 2020-05-19. [citado na p. 39]

[79] P. Mania and M. Beetz, "A framework for self-training perceptual agents in simulated photorealistic environments," in *International Conference on Robotics and Automation (ICRA)*, (Montreal, Canada), 2019. [citado na p. 41]

[80] N. K. John Hsu and D. Coleman, "gazebo_ros_pkgs." http://wiki.ros.org/gazebo_ros_pkgs, 2018. Accessed: 2020-06-29. [citado na p. 43]

[81] S. Kohlbrecher and J. Meyer, "hector_gazebo_plugins." http://wiki.ros.org/hector_gazebo_plugins, 2016. Accessed: 2020-06-29. [citado na p. 44]

[82] B. Bingham, C. Aguero, M. McCarrin, J. Klamo, J. Malia, K. Allen, T. Lum, M. Rawson, and R. Waqar, "Toward maritime robotic simulation in gazebo," in *Proceedings of MTS/IEEE OCEANS Conference*, (Seattle, WA), October 2019. [citado na p. 45]

[83] L. Meier, "Gazebo for mavlink sitl and hitl." `https://github.com/PX4/sitl_gazebo`, 2016. Accessed: 2020-06-27. [citado na p. 46]

[84] V. Ermakov, "Mavros – mavlink extendable communication node for ros with proxy for ground control station.." `http://wiki.ros.org/mavros`, 2018. Accessed: 2020-06-27. [citado na p. 46]

[85] A. Visioli, *Practical PID control*. Springer Science & Business Media, 2006. [citado na p. 47]

Appendix A

ROS and Gazebo Environment Setup

This appendix serves as a guide or tutorial on how to setup the ROS and Gazebo environments. It shows every step taken to setup said environments in order to complete this dissertation.

A.1 ROS

In order to install ROS one should follow the steps presented on its official website (`https:www.ros.org`). For this dissertation, the Kinetic version of ROS is considered. The installation steps are presented at `http://wiki.ros.org/kinetic/Installation/Ubuntu`.

It is highly recommended to use the command line program called Terminator, as this program can open several command lines at the same time.

1. Press `Ctrl + Alt + T` to open a command line terminal.
2. Use the following command to add the ROS packages to your computer:

   ```
   sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'
   ```

3. Use the following command to access the key server:

   ```
   sudo apt-key adv -keyserver 'hkp://keyserver.ubuntu.com:80' -recv-key C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
   ```

4. Use the command to update your system: `sudo apt-get update`
5. Use the following command to install the full ROS desktop version (ROS, rqt, rviz, robot-generic libraries, 2D/3D simulators, navigation and 2D/3D perception):

   ```
   sudo apt-get install ros-kinetic-desktop-full
   ```

6. Use the following command to install various tools and dependencies needed to compile and run ROS packages:

   ```
   sudo apt install python-rosdep python-rosinstall python-rosinstall-_generator python-wstool build-essential
   ```

7. In order to easily install system dependencies, run the following commands: `sudo apt install python-rosdep` -> `sudo rosdep init` -> `rosdep update`

8. Run the command `source /opt/ros/kinetic/setup.bash` to have access to the ROS commands. This needs to be used every time a new terminal is open. However, this can be automated as it will be shown.

9. Create the ROS workspace with `mkdir -p /catkin_ws/src` -> `cd catkin_ws` -> `catkin_make`

10. To automate step 8, edit the shell configuration file with `gedit /.bashrc` and add a new line containing `source yourlocation/catkin_ws/devel/setup.bash`, as shown in Figure A.1, and save and close the text file. The set up of the ROS environment is finished.

```
# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi
#source /opt/ros/kinetic/setup.bash
source /opt/ros/kinetic/setup.bash
source /home/vtr101/catkin_ws/devel/setup.bash
#source /opt/ros/indigo/setup.bash
#source /opt/ros/indigo/setup.bash
```

Figure A.1

A.2 Gazebo

With ROS installed, follow the steps presented in `http://gazebosim.org/tutorials?cat=guided_b&tut=guided_b1` to install Gazebo.

1. Download Gazebo from `http://get.gazebosim.org/` and save it in the `Downloads` folder.

2. Press `Ctrl + Alt + T` to open the command line.

3. Type in the command line the command `chmod +x /Downloads/gazebo.sh` to turn the downloaded file into an executable file.

4. Run the command `gnome-terminal -working-directory=" " -e "./Download_s/gazebo.sh"` to start the Gazebo installation.

5. A new window will appear to insert the system password.

6. Once the window disappears, Gazebo should be installed on the system.

Before installing the aerodynamics plugin, the development package must be installed. This can be done using the command `sudo apt install libgazebo8-dev`.

To write the plugin, the following steps need to be taken:

1. Start by creating a new directory: `mkdir /liftdragplugin` -> `cd /liftdrag_plugin`
2. Create a code source and header files for the plugin: `gedit pluginexample.cc` -> `gedit pluginexample.hh` and write the plugin code.
3. Create a CMakeLists file: `gedit CMakeLists.txt` and type in the required directiories for the plugin.
4. Create a build folder: `mkdir build` -> `cd build`
5. Compile the plugin: `cmake../` -> `make`

The plugin is now created as a .so file inside the build folder. To use it in the ROS workspace, copy and paste this .so file to the /devel/lib/ directory inside the ROS workspace.

Appendix B

Results

B.1 Autonomous Positioning of the Wing Sail to Different Angles of Attack

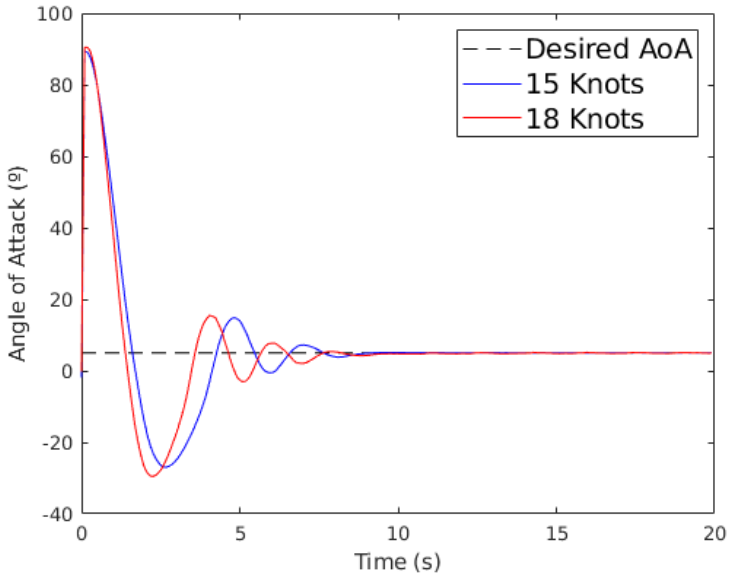


Figure B.1: Wing Sail Angle of Attack with Wind Direction of 0° ($K_p = 0.05$, $K_d = 0.07$, AoA = 5°)

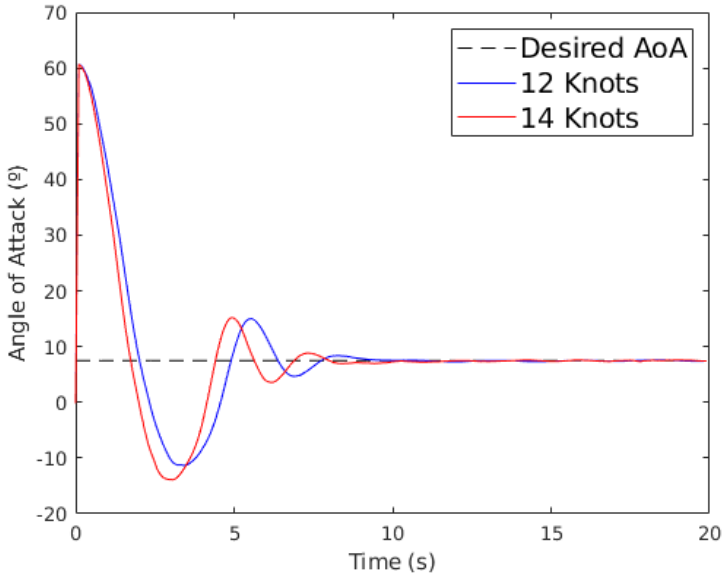


Figure B.2: Wing Sail Angle of Attack with Wind Direction of 30° ($K_p = 0.05$, $K_d = 0.07$, AoA = 7.5°)