POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Towards the adoption of a Vortex Step Method for the aerodynamic evaluations in fixed-wing AWESs flight simulators

Tesi di Laurea Magistrale in
Aeronautical Engineering - Ingegneria aeronautica

Author: **Riccardo Ferrari**

Student ID: 963320
Advisor: Prof. Alessandro Croce
Co-advisors: Prof. Carlo Emanuele Dionigi Riboldi, Filippo Trevisi
Academic Year: 2021-2022

Abstract

This thesis aims at studying the aerodynamics of a rigid wing system for Airborne Wind Energy application, by developing analytical engineering models of low and mid-fidelity aerodynamics. A low-fidelity aerodynamic model is essential during the preliminary and conceptual design of an Airborne Wind Energy System (AWES). The model can support the analysis of local aerodynamic phenomena and can be exploited with different aircraft configurations due to its versatility. It is a particular version of the Vortex Step Method (VSM), in which a five-filament horseshoe vortex is used and includes nonlinearity through interpolating the airfoil polars. The implementation logic of the model is presented shortly after the theory behind it, followed by validation. The latter is based on a build-up approach, starting with the single elliptic wing until the addition of the horizontal tail. The results are obtained from varying important parameters, whose modification is necessary during the conceptual design of AWEs. The Tornado Vortex lattice Method (VLM) is introduced and tested inside an existing simulation framework in the following part of the thesis. To conclude the implementation, the VSM and VLM are compared, showing good behaviour of the VSM both from a numerical result and a computational time point of view. Finally, the vertical tail is implemented; once the tail is complete, control surfaces are also introduced without realistically designing them but by introducing a lift coefficient variation in the airfoil polars interpolation. Finally, the kite's trim on a circular trajectory is analysed. The trim condition's physics and ruggedness are verified by varying specific parameters that affect the state variables, confirming the aerodynamic method's effectiveness. Moreover, the trim has been tested with a simplified tether model and a more complex one extracted from an existing simulator (Delft MegAWEs), showing good behaviour in both cases. The results obtained with the VSM throughout the entire thesis show the potentiality of this method to estimate the aerodynamic characteristics of fixed-wing AWEs and the possibilities of exploiting the model for the conceptual design of these systems.

Keywords: Airborne Wind Energy, Aerodynamic, Vortex Step Method, Vortex Lattice Method.

Abstract in lingua italiana

La tesi si propone di studiare l'aerodinamica degli alianti rigidi utilizzati per l'estrazione di energia eolica ad alta quota, sviluppando modelli matematici ingegneristici di aerodinamica a bassa e media fedeltà. Disporre di un modello a bassa fedeltà è essenziale durante la progettazione preliminare e concettuale di questi sistemi. Esso può essere utile per studiare i fenomeni aerodinamici locali e per analizzare diverse configurazioni. Il modello è un Vortex Step Method (VSM), in cui viene utilizzato un vortice a ferro di cavallo a cinque filamenti e viene inclusa la non linearità attraverso l'interpolazione delle polari dei profili aerodinamici. La logica di implementazione del modulo viene presentata subito dopo la teoria che ne è alla base, seguita da una validazione. Quest'ultima si basa su un approccio incrementale, partendo dalla singola ala ellittica fino all'introduzione della coda orizzontale. Il metodo è stato testato variando parametri importanti, la cui modifica è necessaria durante il progetto concettuale degli AWEs. Nel proseguo della tesi, il Metodo Vortex Lattice (VLM), implementato in Tornado, viene introdotto e testato all'interno di un framework di simulazione già esistente. Per concludere l'implementazione del VSM, esso viene confrontato con il VLM, mostrando un buon comportamento sia dal punto di vista della corrispondenza numerica che da quello computazionale. Infine, viene introdotta la coda verticale e le superfici di controllo, senza però progettarle fisicamente, ma introducendo una variazione di coefficiente di portanza nell'interpolazione delle polari dei profili. Infine, viene analizzato il trimmaggio dell'aliante su una traiettoria circolare. La fisica e la robustezza della condizione di trim vengono verificate variando specifici parametri che influenzano le variabili di stato, confermando l'efficacia del metodo aerodinamico, Inoltre, il trimmaggio è stato testato con un modello di cavo semplificato e con uno più complesso, estratto da un simulatore esistente e mostrando un buon comportamento in entrambi i casi. I risultati ottenuti con il VSM durante la tesi mostrano le potenzialità di questo metodo nello stimare le caratteristiche aerodinamiche di un velivolo e le possibilità di sfruttare il modello per la progettazione concettuale di questi sistemi.

Parole chiave: Eolico di alta quota, Aerodinamica, Vortex Step Method, Metodo Vortex Lattice.

Contents

Introduction

History has taught us that during humanity's entire lifetime, several time phases have followed one another. Temporally speaking, these phases can be divided into past, present, and future.
The continuous evolution in the past has been possible thanks to the strong bond between humans and the environment, and between human and energy resources.
Nowadays, human beings are going through a critical transition phase in which this connection has reached a point of no return. Humanity can no longer think about simply extracting energy without considering all the possible consequences. As humankind, we own the key to averting the worst effect of climate change. Nowadays, humanity is mostly focused on how to obtain all the energy needed to support the enormous increase in population; this point of view needs to change, and what will come can be considered THE greatest challenge humankind has ever faced. The point of no return is tremendously near; furthermore, recent events like the war in Ukraine are creating a fracture inside the energy policies of different countries that, until now, did not think of a renewable solution in this field.
The future is around the corner, and the words that will shape it **must** be "renewable energies."
Humanity can no longer think of climate change as a secondary problem: industrial production processes, mobility, and general behavior need to be immediately modified together with a rapid adaption to all future challenges.

From the electricity production side, renewable energies must overcome conventional fossil fuels. It is true that Europe has set a goal to become climate neutral by 2050 [9] but undergoing events could modify this objective, creating unwelcome consequences since each country will need to design its new strategy, and there is no such thing as a one-size-fits-all approach to the clean energy transition. Reaching net zero by 2050 requires the widespread use of technologies that are not on the market yet. Major innovation must occur over this decade to bring these new technologies rapidly into the market [9].

From this perspective, it is clear that the new renewable sector needs to find a way to help this transition happen in the best possible way. **Airborne Wind Energy (AWE)** has the potential to be the spearhead of this "renewable revolution." Over the past three decades, wind energy has evolved from a niche energy source to an established technology delivering [GW] of power worldwide. While significant research and development efforts are focused on the economical development of floating offshore wind energy systems, these new AWEs systems represent a new way to extract energy from the wind [46], by capturing wind energy at higher altitudes through tethered flying devices. In the last decades, progress has been made, following the lead of Makani, which has developed a 600 kW prototype and tested it offshore and onshore.
All the knowledge obtained in the last decade by Makani Team has been published in three reports.

Motivation and goals

I became aware of this sector only nine months ago. However, I had the opportunity to participate in an International Conference where I met many researchers, universities and companies struggling and pushing to provide different solutions to different AWEs-related problems. For every aspect of this complex machine, including control, aerodynamics, tether and structure, a lot of minor improvements are presented to reach a common solution and set a milestone for possible industrial utilisation. This thesis will exploit a low-fidelity Vortex Step Method to study the advantages and disadvantages of engineering aerodynamics models for AWE systems. Many existing simulation environments are based on simple models such as Look-Up Tables or much more computationally demanding models. Introducing an adaptable and general low to mid-fidelity method can be helpful for conceptual design exploration and better understanding AWESs 'flight mechanics' principal and characteristic phenomena. Once the advantages are known, the method can be used for studying Ground Gen and Fly Gen rigid Airborne wind energy systems in existing simulators.

Structure of the thesis

This thesis is divided into four chapters, the conclusions and future developments.

- In chapter 1, a general description of AWEs is carried on, briefly describing how they work, the different technologies that characterize them and the actual state of the art. A specific introduction of where this work is situated will lead to the second chapter.
- In chapter 2, a study of the simulators available is presented, focusing on the aerodynamic methods presented in each of them, justifying the implementation of the VSM.
- In chapter 3, the MATLAB implementation of the VSM is presented, followed by a validation of the same model with an elliptic wing only, an elliptic wing with dihedral and sweep angle introduction, the wing plus horizontal tail and a comparison with the Tornado Vortex Lattice Method. Finally, the vertical tail is introduced since it is paramount for the kite's trim, which is performed later.
- In chapter 4 a kite trim is performed exploiting the newly implemented aerodynamic method, analyzing the feasibility of the obtained solution, and finding an appropriate starting point for a future study of the dynamic of the kite.
- In chapter 5, the conclusion obtained from the various chapter are summarized, alongside the introduction of future related research and issues to be investigated.

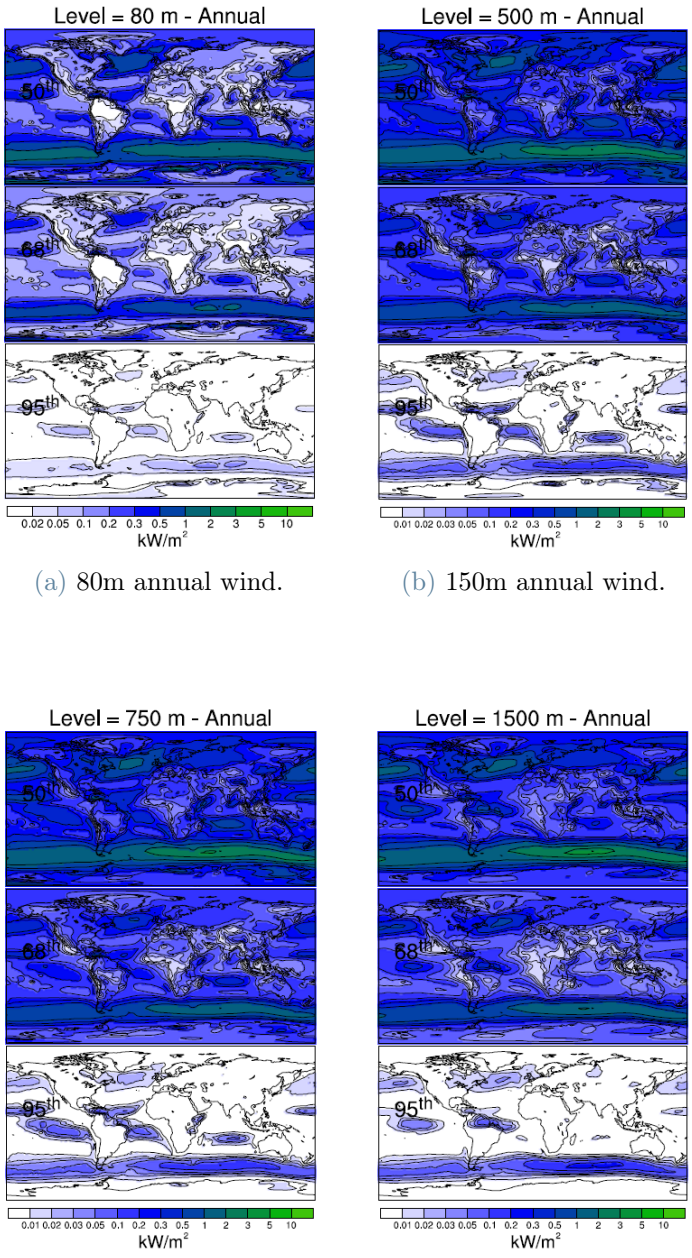
1 | Airborne wind energy sector analysis

The first chapter of the thesis presents a general analysis of the AWE sector.
In section 1.1. a list of the wind energy sector's promising aspects is presented, along with the main idea behind the development of AWEs systems and a suitable collocation for them in the market. Then, a technical overview is shown in section 1.2, analysing all the existing types of AWEs and categories. Finally, the current product and future prototypes are analysed in section 1.3.

1.1. Motivation and concepts

Wind energy consumption is a significant power source because of the wind's broad potential. Outside hydroelectric power, wind energy consumption is the most significant component among renewable energy sources [36]. As stated in the same paper, in 2017, wind energy consumption accounted for 52% of global renewable energy consumption, and solar was only at 21%.
From an environmental point of view, if the commitment to energy transition and wind Energy (WE) diffusion continues to increase worldwide at the same pace, by 2030, WE will be responsible for saving nearly 23 billion tonnes of CO_2, in addition to the benefit of creating millions of new job [17].
As stated by Diòneges et al. [17], from an economic point of view, over the last decade, global onshore wind capacity has reached an installed capacity of 486.79 GW, and with attention shifting to the offshore, this value is only destined to increase.
Nowadays, the energy carried by the wind is exploited primarily by Horizontal axis wind turbines (HAWTs) and, to a lesser extent, by Vertical Axis Wind Turbines (VAWTs). The development these machines followed started onshore, but the latest frontier is to put them offshore, where there is a better quantity and quality of wind, more continuous and intense. The next step should be to take advantage of high-altitude winds, which can only be reached by HAWTs if there is the possibility to increase the already tall

tower structure, highly impacting the costs and relative masses needed to support it. As stated by Archer et al.[7], only a few locations over land experienced median wind power densities more significant than $0.3\,\mathrm{Wm}^{-2}$ at 80 m (typical values for conventional wind turbines). However, as height increases to 1500 m and above, half of the world can guarantee significant power densities. The following resource maps suggest that AWEs have the potential to supply a significant portion of humanity's electricity needs:



(a) 80m annual wind. (b) 150m annual wind.

(c) 750m annual wind. (d) 150om annual wind.

Figure 1.1: Overview of AWEs suitable environment and related winds.

Maps presented in fig. 1.1 describe a perfect scenario for AWEs; Furthermore, one of the most important reasons these devices are so attractive is their theoretical capability of reaching the MW scale with a single plant. This is an essential feature that will help in commercial development, and it is uncommon in renewable energies [13]. In general, several advantages of this system can be found in the literature, and the most important ones are hereinafter presented:

- Drastic reduction in the amount of material involved, mainly due to the absence of a tower, creating a more compact system with lower visual impact; [16]
- Component, logistics and installation cost are expected to fall; [16]
- Maintenance of AWEs system can be done at the ground, consequently, operational costs should be lower; [16]
- Even if long-term commercial operations of AWE power plants still need to confirm it, the energy is expected to be cheaper than that generated with conventional technologies; [15] [21]
- High full load hours and, consequently, more constant energy production, creating a better system integration with less intermittency;
- Higher flexibility, easier logistics;
- New markets: floating offshore smaller platforms and, eventually, re-powering of existing off shore sites;
- Lower carbon footprint: saving concrete and steel resources reduce the system's environmental impact during production, showing a higher material efficiency. Furthermore, material reduction influences all levels of the value chain;

Naturally, everything that comes with advantages is also surrounded by disadvantages. AWEs, in particular, are trying to connect aircraft design with wind turbines, which drags into the overall topic many challenges.:

- Even if much research has been done from the control system point of view, several improvements, especially in autonomous control, must be made to overcome difficulties and provide better security and reliability;
- A lot of public and privates societies offered funds for this research field, and several stakeholders are interested in the topic, but the commercialization of AWE products is still a little bit obscure and uncertain;
- systems behavior in case of extreme conditions have yet to be addressed;

1.2. Technological analysis

Since this thesis is focused on AWEs, they must be introduced properly, starting from a technological point of view and introducing several classifications based on how they fly, how they are made, and finally, how power is extracted. Initially, these devices can be divided considering the *flight mode*:

1. Perpendicular to the tether, referred to as crosswind;
2. Aligned with the tether;
3. In a rotational way;

These devices can be further divided into two fundamental categories:

1. Ground Generation Airborne Wind Energy Systems (GG-AWEs);
2. Fly Generation Airborne Wind Energy Systems (FG-AWEs);

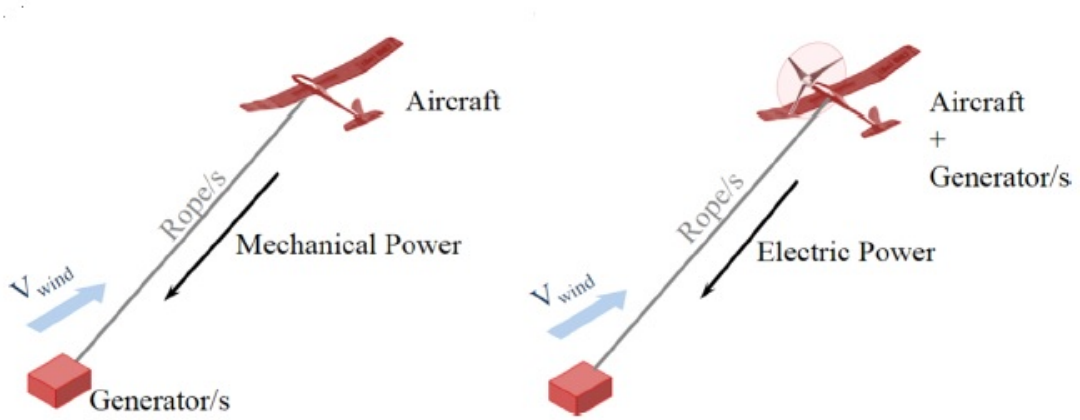
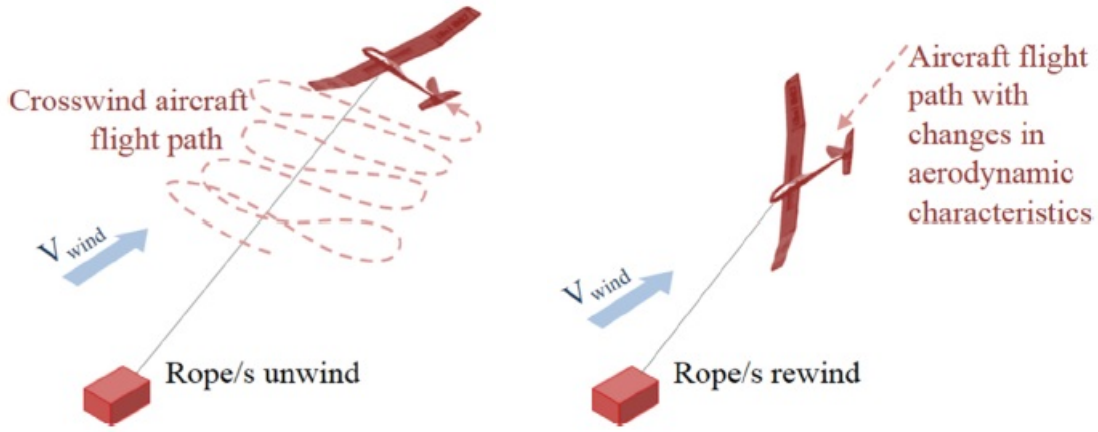


Figure 1.2: FG and GG AWEs - source: [13]

GG-AWEs generates power through an electric generator placed on the ground and connected to the vehicle through a cable or tether. The power is harvested by pulling and unwinding the tether. This typology can be further divided into two categories: the first one is characterized by power production thanks to a moving ground station pulled by multiple kites, whose advantage is the continuous production of energy. For further information on this classification refers to [10], [11], [13] and [30].

The second category has a fixed generator on the ground, and aerodynamic forces (especially lift) acting on the kites provoke the unwind of the cable. During the **reel-out** phase, the generator rotates, producing power. When the tether reaches the max elongation, a **reel-in** phase, also known as recovery phase, starts. This recovery phase is purely passive; the kite's attitude must be changed to minimize energy consumption and avoid wasting the reel-out transition. A representation of these two phases is presented in fig. 1.3.



(a) Reel-out phase. (b) Reel-in phase.

Figure 1.3: Representation of GG-AWEs phases - source:[13]

FG-AWEs produce energy onboard using specially designed turbines connected to onboard generators; successively, energy is transmitted to the ground through one special rope filled with electric cables. The kinetic energy is already converted into electric energy without exploiting a ground generator.

A final subdivision concerns **the type of flying device**, in particular:

- Soft kites, cheaper and easier to repair, the influence of gravity is lower, but they are tremendously hard to fly autonomously and have a lower aerodynamic efficiency.
- Rigid or hard kites, these devices are more expensive and require elaborate launching and landing systems. Gravity is way more influential, but they have higher performance, is long-lasting, and is far easier to control. Moreover, due to the higher lift-to-drag ratio, they can reach very high velocities, which benefits higher power output per wing area. [7].
- A third strange but interesting category is related to hybrid configurations, mixing both rigid and flexible and rigid elements. An interesting hybrid concept is called tensairity and exploits compressed air tubes and tension elements to increase

the maximum wing loading by simultaneously keeping a light weight; for further information, refer to [7].

1.3. State of the art and current scenario

This new sector is evolving at high rates and many institutions are interested in this new technology.

Thanks to [1], a subdivision of all researches, industries and development activities involved in this field can be collected into a single picture (fig. 1.4).

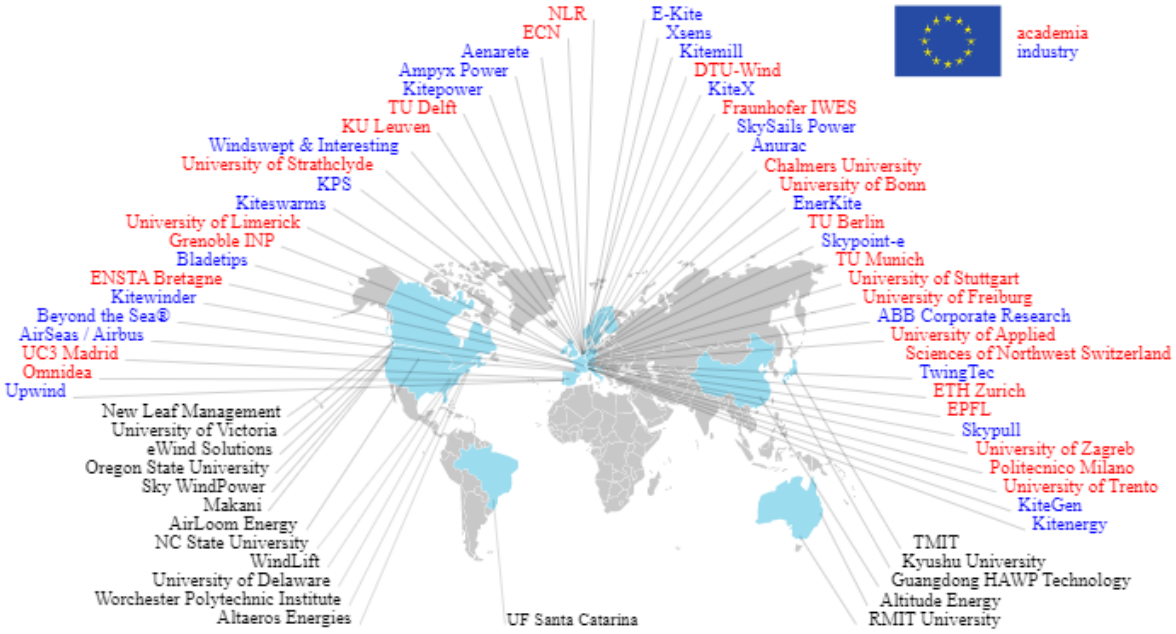


Figure 1.4: Research and activities across the globe: [1]

In fig. 1.5, the main companies that want to enter the market with several devices are presented:

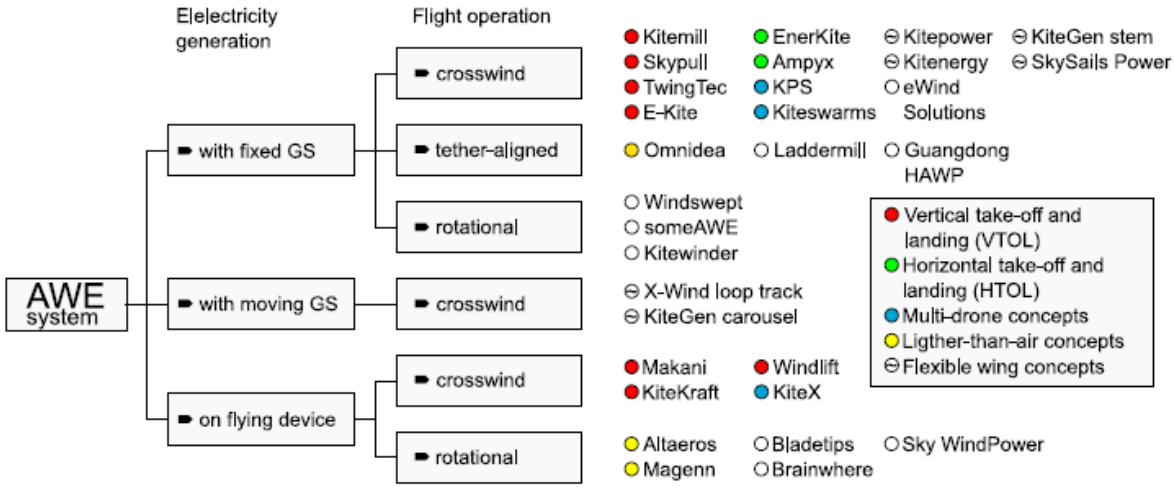


Figure 1.5: Main companies in the sector [46]

The current state of the art shows that most companies and research institutes are focusing on studying AWEs that generate aerodynamic lift by flying crosswind. However, several institutions are also working to understand rotational flight better. Starting by **on flying device**, the most significant effort has been given by Makani, KiteKraft, and Windlift; they exploit Vertical take-off and landing configurations for their devices in a crosswind situation.

- *Makani* is why this field has reached many companies and has much research going on. Even though their work stopped in September 2020, their ten years of effort have been recovered, and their knowledge has been spread thanks to three complete reports. Their technology represents the state-of-the-art for crosswind FG-AWEs.
- *KiteKraft* proposes a box-plane structure flying eight-shaped trajectories designed to maximise rigidity while minimizing weight. The following picture helps in defining the box-plane structure.

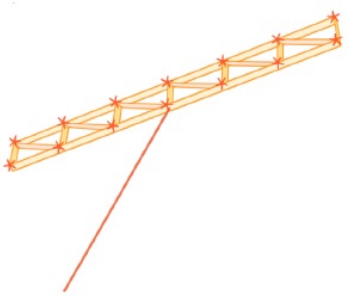


Figure 1.6: Typical boxwing configuration: [13]

- *Windlift* is developing a glider with four rotors suitable for remote applications, for further informations refers to [5].

Regarding **GG-AWEs with fixed GS**, it is possible to say that this is the most diffuse technology in the market. The leading companies exploiting rigid kites technologies are Ampyx power, Kitemill, TwingTec, KiteGen Research, Kitenergy, SkySails power, TU Delft, Enerkite and Skypull, even though some of these companies are not present in fig. 1.5, several information can be obtained thanks to Cherubini et al. [13]. Further information regarding these companies are hereinafter listed:

- *Ampyx power* is a Dutch company that first developed a pumping glider generator. They are developing and testing two 5.5 m span 'PowerPlanes', automatically controlled with state-of-the-art avionics. Onboard actuators can drive a rudder, an elevator, and up to four flaperons. During a test campaign in November 2012, the system demonstrated an average production of 6 kW with peaks over 15 kW. Nowadays, this company started the design of a 35 m span PowerPlane with a 'wind turbine equivalent' power of 2 MW [13];
- *Kitemill* proposes gliders with embedded propellers that flies circular trajectories and is testing prototypes in the range 30 - 100 kW;
- *Twingtec* has a similar project in terms of design, but it is aiming to create a MW scale system though they developed a 100 kW GG-AWEs;
- *KiteGen Research* is an Italian company (KGR) that exploits a technology based on a C-Kite integrating onboard electronics with sensors and is controlled by two power ropes from a control station on the ground. After several years of research, they focused on a generator called 'KiteGen Stem' with a nominal power of 3 MW [3]. A peculiarity of their product is that this company aims to retract the cable with minimum energy consumption thanks to a manoeuvre called 'side-slip', consisting of a flight mode in which the kite aerodynamic lift force is cleared by rewinding one rope before the other. KGR also plans to use Kitegen Stem technology to produce offshore AWEs;
- *Kitenergy* has a prototype that features 60 kW of rated power [4];
- *SkySails power* is a German company developing a wind propulsion system for cargo vessels based on kites. Using the technology related to these vessels, they created two prototypes, a mobile AWEs having a capacity between 250 kW and 1 MW and an offshore AWEs with a capacity from 1 to 3.5 MW;
- *Enerkite* ([2]) developed an interesting solution, creating a portable pumping kite

generator with rated continuous power of 30 kW.The ground station is installed on a truck through a pivotal joint which allows for azimuthal rotations. The company plans to produce a 100 kW and a 500 kW system.

- *Skypull* is a Swiss company that proposes an innovative approach, based on a particular qadri-rotor "Box-wing" drone able to perform Vertical Take Off (TO) and Landing (LND), and once it reaches the correct altitude it flies eight-shaped trajectories. The structure is the one presented in fig. 1.7 and for further information refers to [6].

Figure 1.7: Skypull box-wing configuration - source: [6]

The last category concerns **GG-AWEs with moving GS**. Several companies are also active in this case, even if they are less than GG-AWEs with fixed GS. In particular, KGR again proposed the concept of a moving GS with a carousel configuration, aiming to recreate a vertical axis wind turbine driven by forces from tethered aircraft. There are no practical prototypes yet, but studies have been using simulation showing that 100 kites with 500 m^2 area could generate 1000 MW of average power in a wind speed of 12 m/s [11]. The considered generator would have a 1500 m radius, occupying a smaller territory with far lower costs.
Kitenergy is also active, working on a straight linear rail fixed on the ground with a pivotal joint. The ground station is mounted on a wheeled vehicle that moves along the straight rail and is in charge of extracting the energy using electromagnetic rotational generators placed on the wheels.

The last sector that is worthy of attention, even if it is not a properly airborne-related

sector, is the extension of AWE systems in **underwater energy harvesting**. Precisely, tethered underwater kites extract energy from ocean currents, tidal flows, or rivers. Ocean kites use rigid wings, fuselages, and control surfaces since weight is less considered. The power generator can be placed either on the kite or the seabed. The principal advantage of this configuration is that power densities in typical marine flows are, generally, six to seven times higher than in the wind flows [46]. Alongside advantages, several problems must be faced, like challenging marine environments, currents, and wave loading, but this is an up-and-coming technology. For example, Minesto's ([5]) **Deep Green** technology consists of a kite-mounted turbine and generator developed commercially with a 500 kW device. Furthermore, an economic analysis of this deep green technology calculated weight/power ratios of 14, compared with 50-600 resulting from conventional marine hydro-kinetic turbine studies.

AWESs concept, as the possible evolution of the wind turbine, is visible in the following picture:

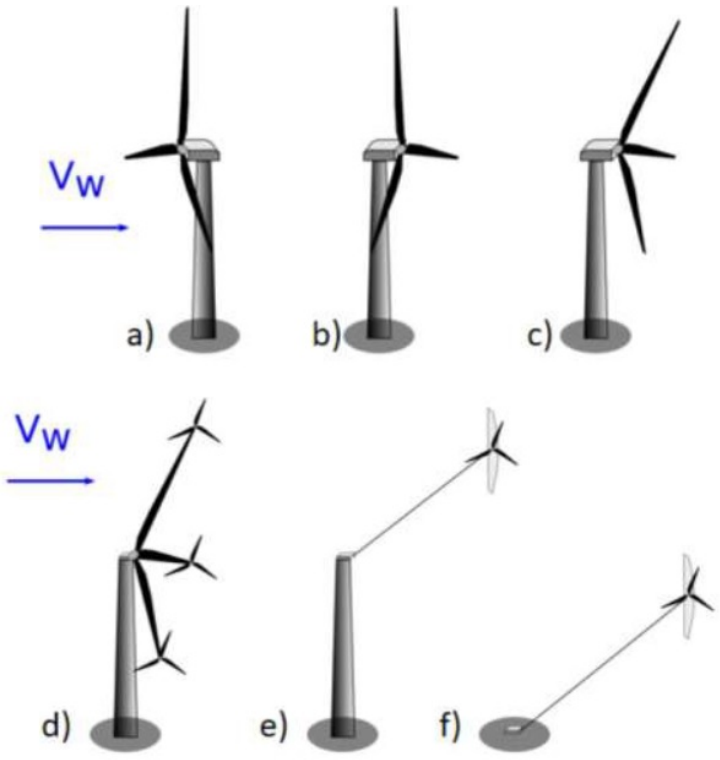


Figure 1.8: Actual AWEs path - Source: [43]

In fig. 1.8 its clear how the evolution of these devices can be seen:

1. In the first part, sub-picture (a), a common HAWT is presented;

2. The following step, presented in sub-picture (b), will be to put the wind turbine into a downwind condition;
3. The next idea will be to put this HAWT into a downwind condition but with load alignment, meaning centrifugal and thrust forces are aligned with the blade direction. Therefore, there is no need to try and balance the flapwise bending moment since only edgewise is present. Thrust is in the downwind direction, and the centrifugal forces are in the radial direction; the vector summation of the two is aligned with the blade direction.
4. The fourth step is to put the generators at the tip, removing them from the nacelle.
5. In the fifth part, always considering load alignment, power production surfaces have been shifted to the tip, just like tip generators. In this situation, neither flapwise nor edgewise bending moments must be transferred to the root. Thanks to power-producing surfaces on the tip, loads will be distributed only here.
6. The final solution consists of moving the tether attachment point to the ground and removing the tower with the related weight and logistic problems.

Thesis theoretical basis and starting point This thesis will develop and enter the world of flight mechanics and aerodynamics of kites, joining researchers worldwide who struggle daily to understand the physics behind AWEs' movements and their behaviour in the sky. An efficient low-fidelity aerodynamic model is paramount to study conceptual design of AWEs. Therefore, this thesis proposes a valid alternative to aerodynamic models available for fixed-wing AWEs analysis, highlighting advantages and possible usefulness of this new method.
The thesis starts with an analysis of the available simulators and related aerodynamics. Consequently, the model's theory and implementation logic is presented, together with a validation of it. It ends with a specific kite's trim on a circular trajectory, creating a potential starting point to be exploited as the initial condition of future dynamic simulations while verifying the functionality of the newly implemented aerodynamic method.

2 | AWEs simulation tools: a state of the art review

The second chapter of this thesis presents an analysis of the available open-source software for studying AWEs. Specifically, a comparison between the most important ones, highlighting differences, similarities and mathematical models that define them. With special attention to aerodynamic models.

2.1. KiteFast

The first code to be studied is KiteFast, a wind turbine simulation tool that builds on FASTv8. FASTv8 is a computer-aided tool for simulating the coupled dynamic response of wind turbines. In the last period, the most suitable option was to switch to openFAST, providing a solid software-engineering framework for FAST, completed with the full documentation, extensive automated regressions, a robust multi-platform, and a compiler build system.

2.1.1. Description and state of the art

KiteFast is a simulator for airborne wind energy systems based on NREL's OpenFAST turbine simulator. It is a **multi-physics** engineering tool for modelling the coupled aero-servo-elastic dynamics of AWEs. Makani engaged with NREL to develop, verify and document this multi-physics engineering model for a megawatt-scale tethered energy kite. They also worked on enhancing this program, called KiteFAST-OS, including floating offshore platform simulations. Before this project, the capability to model the aero-hydro-servo-elastic dynamics of fly-gen AWEs in crosswind flight operations did not exist.
With the ending of Makani, KiteFAST and KiteFAST-OS software were merged into a single code (named KiteFAST) and released as public software. Even though KiteFAST can potentially become the leading engineering tool for the detailed design of fly-gen AWEs,

this thesis will not exploit this software specifically ([22]) due to its high complexity. It is useful, however, to have a basic knowledge and explain the most relevant characteristics of one of the most promising software available for free, especially looking at the promising aerodynamic model included in the code.

2.1.2. Mathematical model and physics involved

Regarding **OpenFAST**, *the aerodynamic* models use wind-inflow data to solve for the rotor-wake effects and blade-element aerodynamic loads, including dynamic stall.
The hydrodynamic models simulate the regular and irregular incident waves and solve for several loads (hydrostatic, radiation, diffraction, and viscous).
The control and electrical system models simulate the controller sensors, logic, and actuators of blade pitch and several other components, including the generator and power-converter components of the electrical drive.
The structural dynamic models apply reactions for the control part, apply the aerodynamic and hydrodynamic loads and eventual gravitational load. Coupling between all models has been achieved thanks to a modular interface.
For **KiteFAST** specifically, the engineering model is substantially the same. Physical phenomena and system couplings, including the wind/aerodynamic excitation and full-system dynamic response (fuselage, wing tail, stabilizers, pylons, nacelles, rotors, tether, ground station, control surfaces and controller) are analyzed, under both normal (for fatigue) and extreme (for ultimate) loading conditions [23].

The configuration of the kite is assumed to have one fuselage, two wings attached to the fuselage, one vertical stabilizer attached to the fuselage and two horizontal stabilizers attached to the vertical stabilizer. Moreover, one or more pylons per wing, each with two rotors (top and bottom), one ground station fixed tether with a bridal connection to the wings, and nominal circular motion of the energy kite, a representation, complete with reference frameworks, is presented hereinafter:

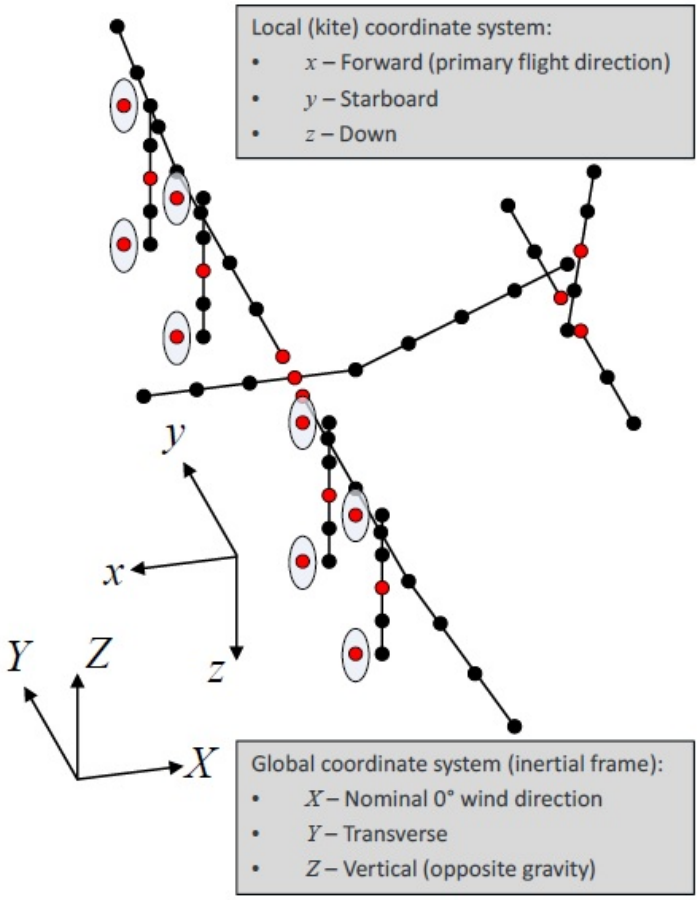


Figure 2.1: KiteFAST reference system and kite model - Source: [22]

KiteFast was developed from:

- the existing MBDyn open-source multi-body dynamics code;
- the existing OpenFAST modules NWTC Subroutine library, AirfoilInfo, InflowWIND, MoorDyn, and TurbSim.
- New source code for generating MDbyn models of the energy kite, computing quasi-steady energy kite aerodynamics and interfacing an energy-kite controller, coupling also the various KiteFAST modules together.

Since there a lot of packages involved and several of them are very important for the physical and mathematical understanding of this simulator, an explanation is given hereinafter:

1. *KiteAeroDyn* is the kite-specific aerodynamics module. The user specifies the general configuration of the kite in the KiteAeroDyn input file. The number of pylons/rotors and the undeflected reference point of each kite component is specified in a body-fixed coordinate system with x pointing forward, y pointing right (when looking in the primary direction of flight), and z pointing downward. This reference point allows for a spatial mesh-to-mesh mapping between the aerodynamics computed by KiteAeroDyn and the structural dynamics computed by MBDyn. The users also introduce several pieces of information regarding wind and control inputs. Thanks to various other packages listed below, this module can compute quasi-steady loads [22]. An existing sub-module inserted in the AeroDyn package is the *AirfoilInfo*; it computes quasi-steady airfoil aerodynamics based on airfoil polars, including interpolation between multiple Reynolds numbers or control surface settings [22]. *KiteVSM* computes quasi-steady induction from lifting lines using the **VSM** (Vortex Step Method), based on Weissinger's method [22]. Finally, *ActuatorDisk* models the rotor of the energy kite as a quasi-steady actuator disk to calculate three forces, three moments, and power dependent on the rotor speed and several other parameters. The loads are derived from Look-Up tables based on user-specified coefficients.

2. *InflowWind* including *TurbSim*, the first module processes the undisturbed wind inflow and supports steady, uniform winds; time-varying winds (from IECWind) and full-field turbulence (from TurbSim). In fact, TurbSim module is a pre-processor for InflowWind that is able to compute full-field stochastic realizations of synthetic wind turbulences. [22]

3. *MoorDyn* is a module of OpenFAST, unchanged in KiteFAST, which models the dynamics of cable elements using lumped-mass dynamics. It is exploited in AWEs analysis of the tethers. [22]

4. The KiteFAST *Controller* interface is the following:

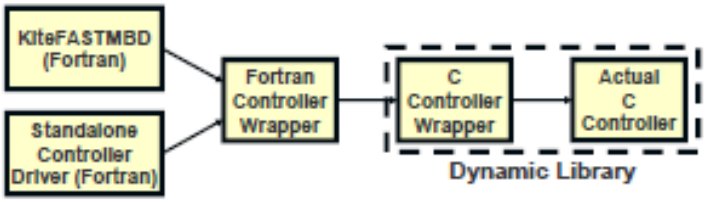


Figure 2.2: KiteFAST controller interface - Source: [22]

5. *MBDyn* is a command-line tool used as the solver for structural dynamics problems. KiteFAST is based on this package, and to exploit it, NREL created a special KiteMBDyn Preprocessor (called PreProc). The most important characteristics of both tools are reported herein: MBDyn uses a 3-node beam as described in [25]. These beams are straight, with a reference axis parallel to any of the kite reference frame axes (x, y, or z) depending on the mechanical component of the kite (wing, stabilizer). PreProc will use the end nodes in the input file to determine the middle point and two intermediate Gaussian points. MBDyn will proceed to calculate internal forces and moments at the Gaussian points. Finally, stiffness matrices provided at the end notes in the kite reference framework are transferred to the Gaussian point coordinates.

2.2. DELFT MegAWES

The second program analyzed is DELFT MegAWES, it is a MATLAB/Simulink model of an airborne wind energy system based on a tethered rigid wing that is operating in pumping cycles producing multiple MW of electricity.

2.2.1. Description and state of the art

This code was born thanks to several researchers, PhD, and master thesis students at Delft university. The motivation was the total absence of a Multi-MW model system for AWES, even though similar systems are present for conventional wind turbines. The thesis in [19] presents the detailed design of the first publicly available Multi MegaWatt Airborne Wind energy reference system. They allow researchers and universities worldwide to access an optimization framework and a fully dynamic system simulation through a relatively low computational effort model.

2.2.2. Mathematical model and physics involved

MegAWES is able to model a **3 DoF & 6 DoF kite dynamics**. The simulink framework includes the following model components:

- Pre-calculated look-up tables for aircraft's aerodynamic behaviour;
- Segmented tether with a single attachment point at the kite's center of gravity;
- Choice between 3DoF point-mass and 6DoF rigid-body dynamic solver;
- Aircraft controller for power generation flight controls and path tracking;
- Set-force controlled dynamic winch.

The optimisation framework behind ([19]) is based on a Fluid-Structure Interaction (FSI) model combined with a flight dynamics simulation model.
The FSI model is based on a 3D linear structural Finite Element model coupled with a potential-flow 3D panel method. The wing mesh and structural components are obtained with MATLAB, while stiffness, mass and inertia matrices are achieved with MSC.Nastran. The algorithm is started thanks to a *system parametrisation* of the most relevant part of the AWES, once all the parameters are initialized, an input file for Nastran is created.
The **structural model** consists of a 3D finite element discretisation modelled with plate (CTRIA3), beam (CBEAM) and rod (CROD) elements [19].
The **Aerodynamic model** consists of a Fluid-Structure Interaction (FSI) algorithm

based on the aircraft's coupled aeroelastic and flight dynamics. With the latest version of the code, the FSI algorithm is switched off in favour of a simple Look Up Table, thanks to which pre-computed static aerodynamic coefficients are obtained and exploited to determine the aerodynamic forces.
The **Control and flight dynamics model** is based on the definition of several sub-models for all the crucial parts. The aircraft is represented by a fixed mass, and the tether can be described thanks to a rigid body approximation, linear spring-damper element or multiple particle systems. Since the purpose is to model Multi-MW systems, the tether increases a lot in thickness and length; consequently, the drag is raised. Finally, the ground station model consists of three components, the generator, the drum and the gearbox. The implementation is based on the conservation of angular momentum principle with the simplification of an infinite stiffness for shafts and gearbox [19].
The **controller and flight path** is based on simplicity, choosing to control and fly the aircraft on a prescribed trajectory in order to make it produce more power. Since the structural model and aerodynamic model do not match their interface, an interpolation between meshes is carried on with two different methods:

1. The thin plate spline method, used to transfer the structural displacement from one mesh to another;
2. The inverse distance weighting, that allows to transfer back the forces on the aerodynamic mesh.

2.3. LT-GliDe

The third program is called LT-GliDe, short for *Linearized Tethered Gliding system Dynamics*. It is a program create by Filippo Trevisi during his first year as a doctorate at Politecnico di Milano, alongside Professor Carlo E.D. Riboldi and Alessandro Croce. It is implemented in MATLAB environment.

2.3.1. Description and state of the art

It is important to recall that this suite is a module of an under-development multidisciplinary design and optimization framework for rigid wing AWES, named T-GliDe (Tethered Gliding system Design). The most important feature behind this code is the **linearization process** involved, explained in detail in the following section. The depicted model allows for a good physical understanding of design choices' effect on performance, especially on stability.

2.3.2. Mathematical model and physics involved

The linearized dynamic of the system is suitable for a parameterized analysis of AWEs. The suite implemented in Matlab is aimed at the eigenanalysis and forward-in-time simulation of a specific geometry [44].
The linearized model allows for a simplified investigation of the influence of the main wing geometry, main wing aerodynamics, control surfaces aerodynamics and geometry, tether attachment, and tether mechanical and aerodynamic properties. Since the power generation mechanism does not primarily impact the dynamics, it can be avoided in order to take into account and model both Ground-Gen and Fly-Gen AWEs. Hereinafter, a brief explanation of the model behind this suite is presented; for precise information and analytic background, please refer to [44].
The six DoF equations of motion are linearized about a fictitious steady-state motion of the AWES in the selected path (circular path has been properly chosen to create an axial-symmetric problem). No real steady-state can be achieved during power generation because of the continuous manoeuvres of AWEs; the fictitious steady-state motion is computed by considering all fluctuating terms as disturbances. Thanks to the selected flight path and considering the made-up steady-state motion, centrifugal forces are balanced by the radial component of the force of the tether, avoiding exploiting lift for turning manoeuvres. Finally, the derivative of external forces and moments are taken with respect to the steady state, formulating the linearized problem [44].
The linearized model allows for a simplified investigation of the influence of main wing geometry, main wing aerodynamics, control surface aerodynamics and geometry, tether attachment position, tether mechanical and aerodynamic properties and mass properties of the AWEs on flight stability. The approach on which the code is based does not guarantee that a stable AWEs design will converge to the prescribed trajectory without control inputs but allows for studying the AWEs system dynamics. At the same time, set to a **state representative of its flight** during the power generation loop.
The stability of the system can be studied through an eigenvalues analysis, and a general roadmap is depicted hereinafter:

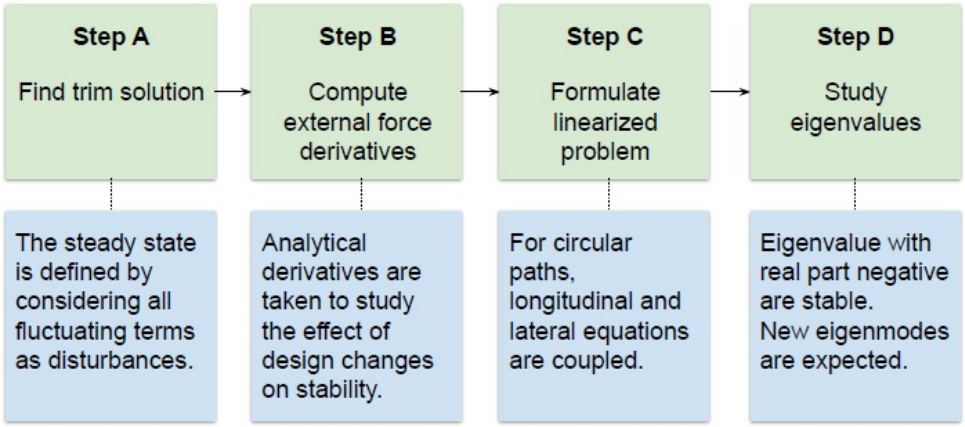


Figure 2.3: Roadmap for stability analysis - Source: [44]

Computationally speaking, for a given AWEs, the codes computes the trim condition (as stated in [44] the "trim" condition is the fictitious steady-state condition), evaluates the external force derivatives and extracts the eigenvalues of the linearized system. The input required by the systems are shown in table 2.1 and table 2.2 and are related to AWEs main wing, horizontal and vertical tails and tether properties.

Main wing		**Horizontal tail**		**Vertical tail**		**Tether**	
A	Wing area	A	H. tail area	A	V. tail area	d_t	Diameter
b	Wing span	b	H. tail span	b	V. tail span	L_t^0	Length at rest
C_{D0}	C_D at zero lift	C_{D0}	C_D at zero lift	C_{D0}	C_D at zero lift	$C_\perp$	Drag coefficient
C_L	Prescribed lift coeff.					E	Young Modulus
Λ	Sweep						
Γ	Dihedral angle						
$f1$,$f2$	y_s location of ailerons						

Table 2.1: Input table 1 - Source: [44]

Geometry		Mass property		Atmosphere	
$\mathbf{X}^b_{OCG}$	Center of mass position	m	Kite mass	V_w	Wind velocity
$\mathbf{X}^b_{OW}$	Main wing root position	I^b_{xx}	Inertia around x_b	β	Mean elevation angle
$\mathbf{X}^b_{OH}$	Horizontal tail position	I^b_{yy}	Inertia around y_b		
$\mathbf{X}^b_{OV}$	Vertical tail position	I^b_{zz}	Inertia around z_b		
$\mathbf{X}^b_{OT}$	Tether attachment position	I^b_{xz}	Off-diagonal inertia		
θ_{SB}	Rotation from $\mathcal{F}_B$ to $\mathcal{F}_S$				

Table 2.2: Input table 1 - Source: [44]

2.4. LAgrangian KiteSimulator

The fourth and last program is called LAgrangian KiteSimulator also known as LASKA. It is a MATLAB code, registered at Registro Territorial de la Propriedad Intelectual de la Comunidad de Madrid in 2017. The authors are Gonzalo Sànchez-Arriga and Alejandro Pastor-Rodrìguez.

2.4.1. Description and state of the art

The work was intially supported bu Fundaciòn BBVA and then continued thanks to the funding of the spanish Minister of Economy. It is based 100% on MATLAB and it is a freely available software for the dynamic analysis of tethered flying vehicles such as kites and fixed-wing drones, applied to airborne wind energy generation [39]. The software includes **four** simulators.
The number 1,2 and 4 consider flexible but inelastic tethers and are based on minimal coordinate Lagrangian formulations and can be used for the analysis of FG and GG systems. The configuration of the mechanical system in the fourth simulator can be defined by the user, who can also select the number of flying vehicles and the properties of the tethers linking them. The code is divided into several modules:

- KiteAcrobat;
- KiteFlex;
- KiteSurf;
- KiteTrain;
- KiteElastic;

Each one will be briefly explained in the following section, for further informations regarding analytic part not useful for this thesis please refers to [31], [38], [39], [40].

2.4.2. Mathematical model and physics involved

The mathematical model for the simulation of AWEs is based on several assumptions, as provided in [40]:

1. The kite is modeled as a rigid body;
2. The tether is divided in segments, each of them is modeled as an inelastic and straight rod;
3. The equation of motion of the mechanical system are derived with Lagrange's formalism and they are not coupled with algebraic constraints;
4. Tether reel-in and reel-out and the deflection of the aerodynamic surfaces are incorporated in the model.

The simulator can be used to study both Ground-Gen and Fly-Gen AWEs, in fact is suitable for most of the AWEs under development. As stated in [39] there are four modules:

1. **KiteFlex**, it is a simulator of FG and GG AWEs. The mechanical system comprises a tethered vehicle with or withour rotors, equipped with a time-varying bridle made of three lines, and attached to the ground by an inelastic and flexible tether.
2. **KiteAcrobat**, considers a two-line kite and aims at kite traction applications such as cargo ship pulling.
3. **KiteSurf**, it simulates the dynamic of four-line kites. It is an important element in the aerodynamic parameter identification algorithm for kites and drones.
4. **KiteElastic**, it is a flight simulator of several tethered vehicles linked by elastic and flexible tethers. It can be used to study the dynamics of AWEs based of train of kites. The equation of motion of this module were obtained by using classical mechanical formulation.

As stated in [40], the simulator considers a drum-generator module that can reel-in and reel-out the tether and controls the tether length. The model incorporates tether flexibility but does not take into consideration the elasticity. From an aerodynamic point of view, three models are presented:

- *Aerodynamic model of the tether*, the simulator considers the aerodynamic force

component normal to each tether segment and ignores the tangential force component and the aerodynamic torque;

- *Aerodynamic model of the kite*, in which the normalized aerodynamic forces and moments about the center of mass are computed exploiting aerodynamic derivatives and reference velocities. The effect of control aerodynamic surfaces (aileron, elevator and rudder) are included in the aerodynamic torque but ignored in the force, except for the lateral force due to rudder deflection;
- *Aerodynamic model of the rotors*, which includes only the aerodynamic force and torque components along the axis of the rotors. The model is based on the airspeed normal to the plane of the blade.

The control vector of the simulator includes variables that are input in real AWEs, such as the length of the main tether, the deflection of aerodynamic control surfaces, and the torque of the motor controller of the rotors.
The fact that a chain of inelastic rods has modelled the tether allows for eliminating the fast longitudinal waves related to elastic tethers.
The model setup yielded a non-stiff set of equations that allows numerical integration with more significant time steps. Another significant physical effect, usually excluded in other AWEs simulators, is considered. Especially tether inertia, aerodynamic drag, and flexibility, as well as particular dynamic effects that are consequences of the high spinning velocity of the rotors [40].

It is very powerful; in fact, it can reproduce TU Delft GG-AWEs by ignoring all the variables related to the rotors and the movable aerodynamic surfaces, alongside the regular operation of Makani's prototype by setting constant values of the control variable and giving a certain number of time histories. Moreover, it is a very *modular* simulator; whatever module is used, inputs and outputs are the same, as can be seen from the following figure:

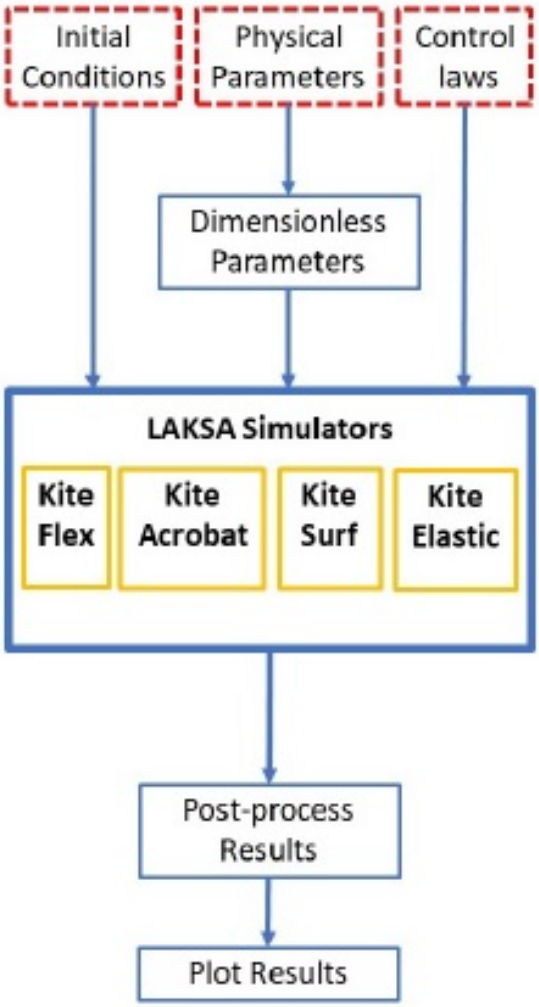


Figure 2.4: LASKA modularity - Source: [39]

2.5. Final comparison and results

A summary of all the codes is present in the final section of this chapter, together with a brief explanation of the final selected environment. The comparison is visible in table 2.3.

Table 2.3: Models' comparison.

Simulator name	Aerodynamic model	Environment
KiteFast	**VSM**	Fortran/Shell/Phyton/Matlab/Roff
MegAWES	LUT / FSI	MATLAB & Simulink
LT-GliDe	Prandtl LLT	MATLAB
LASKA	LUT and derivatives	MATLAB

The selected model simulator will be **Delft MegAWEs**. However, the related aerodynamic module is based on the selection between a specific LUT that is not versatile and an FSI algorithm that is the exact opposite, with very high computational demand. On the other hand, the VSM proposed by Damiani for KiteFast ([14]) is very interesting. Therefore, this thesis focused on implementing a similar version, starting from scratch, on presenting an alternative for the Delft MegAWEs aerodynamic module that, instead, passes from a very intuitive model to a complicated algorithm. The thesis has a double objective:

- Collaborate in modifying the existing simulator, introducing a possible alternative to the aerodynamic module that could be used to obtain results in more general cases and configurations that the LUT cannot analyse with computational requirements typical of low-fidelity potential flow aerodynamic methods;
- Develop an engineering model for low/mid-fidelity aerodynamics that could be exploited generally during the conceptual design of AWEs.

In order to do it, the selected simulator must be computationally efficient, easy to use and the more general possible.
Therefore, KiteFast was discarded due to the high number of programs needed for it to work. Nevertheless, the aerodynamic module is fascinating, and an implementation is discussed in chapter 3.
LT-GliDe is interesting, but due to how it is constructed, it is not useful for a wind sensitivity analysis; furthermore, it is not freely available.
LASKA modularity is intriguing, but it worked for kites and drones equipped with rotors; this is different for this thesis. Furthermore, According to [40], an improvement of the

aerodynamic model of the kite, together with an optimal control strategy, is needed.
The final selection recalls upon **Delft MegAWES**, with the awareness that the Simulink implementation could create problems while integrating the potential method and considering the presence of adaptive and robust control that has been created specifically for the LUT.
Having selected the latter as the sample simulator environment for this thesis, the next step is recalled upon modifying the aerodynamic module. The introduction of this low-fidelity model is necessary for a finer analysis, and it is presented in the following chapter (chapter 3).

3 | Aerodynamic model modifications

In the third chapter of this thesis, the aerodynamic model is implemented, allowing for the initial adjustment of the simulator. The idea is to select as a basis the Delft MegAWEs simulator, mainly created in MATLAB & Simulink.
A modification of the aerodynamic part of the simulator has been performed by implementing a **Vortex Step Method (VSM)** similar to the one used in KiteFast, allowing for a finer aerodynamic analysis with respect to the look-up table (LUT).
The work takes as a basis the VSM implemented in [12] together with [14], with a substantial modification since the simulator is working with hard kites with fixed-wings. Successively, an integration of Tornado **Vortex Lattice Method (VLM)** inside the Delft simulator is carried on, verifying if an already validated potential method is compatible and making possible adjustments in the case of a negative result.
A final comparison between the two methods is made, showing how the two behave in simplified cases. The modularity of the presented VSM is related to the possible introduction of the **Lifting Line Theory (LLT)** simply modifying some of the VSM inputs. The LLT is considered for the short computation time but does not give accurate results for unconventional geometries, such as wings with high dihedral and sweep angles and low aspect ratio.
The Look-Up Table (LUT) is a simplified method with a specific tune on a particular kite. Introducing the Vortex Step Method will allow for the expansion of the possible analysable cases while remaining computationally efficient.

3.1. Example kites: geometries and characteristics

In order to perform the best possible comparison, the kite geometry must be easy to identify and fixed. Two kites are herein introduced:

- **MegAWES airborne wind energy system**, already present inside the simulator. The primary geometric information are reported in table 3.1, for further parameters please refers to [20]:

Table 3.1: Geometry and mass properties of MegAWEs kite.

	Main wing	Horizontal tail	Vertical tail		Mass & Inertia
S (m^2)	150.5	21.3	8.4	m	6885 (kg)
b (m)	42.5	7.6	3	I^b_{xx}	5.768e+5 (kg m^2)
Airfoil	-	NACA 0012	NACA 0012	I^b_{yy}	0.8107e+5 (kg m^2)
Γ	0°			I^b_{zz}	6.5002e+5 (kg m^2)
Λ	2°				

- **Zefiro ultralight aircraft**, presented inside [44] with the main parameters visible in table 3.2:

Table 3.2: Geometry and mass properties of the ultralight aircraft Zefiro.

	Main wing	Horizontal tail	Vertical tail		Mass & Inertia
S (m^2)	14.3	1.1	1.49	m	530 (kg)
b (m)	15.2	2.44	1.6	I^b_{xx}	2104 (kg m^2)
C_{D0}	0.0176	0.0176	0.0176	I^b_{yy}	1122 (kg m^2)
Γ	$-5°$			I^b_{zz}	3134 (kg m^2)
Λ	5°			I^b_{xz}	91 (kg m^2)

Zefiro kite has been chosen for the final comparison, visible in section 3.7.

3.2. Lifting Line Theory and Vortex Step Method introduction and considerations

In this section, two methods are introduced, with a particular embellishment on the VSM, being the final solution for the implementation inside the simulator.
For the purpose of identifying the aerodynamic characteristic of an aircraft, the tools used today are mainly based on Lifting Line Theory, Vortex-step, Vortex-Lattice and Panel methods, excluding simplified Look Up tables and CFD. The three methods are based on the ability of discretising the flow by combinations of fundamental solutions at certain places on the wing and body. The difference between them lies on the position and amount of these solutions.
The drawback of VSM and VLM is simple. No knowledge whatsoever is used on the shape of the airfoil and this results in significant errors for configurations with unconventional airfoils, like slow-speed small aircraft.
On the other hand, a simplified LLT assumes a linear slope for the lift coefficient of the airfoil section that composes the wing, typically close to 2π but significant errors could occur for analysis with unconventional airfoils and near-stall flow conditions.
The LLT base model can be modified to consider the nonlinear nature of the lift slope, especially for angles near stall.
As reported in [42], there is a way of making the lifting line method capable of working with nonlinear slopes.
The method assumes an initial bound vorticity (Γ) distribution, which is used to calculate the induced velocities and angle of attacks along the span. The angle of attack is used to look at the relative lift coefficient (C_l) using the lift data for each section. From the distribution of the lift coefficient, the circulation distribution is recomputed using the Kutta-Jouwkosky (KJ) theorem. This iteration is carried on until Γ converges, as explained in [28].

The first to propose this alternative was Weissinger [47]. The method proposed by Weissinger exploits a single chordwise row of horseshoe vortices, with the bound vortex positioned at each section's local quarter-chord. The boundary condition of zero normal flow is imposed at the control point, located at three-quarter chord.
The latter is fundamental, since the downwash in this point, in the 2D case, is equal to the zero-lift line angle of attack, allowing for the lift force's correct magnitude. The choice of $\frac{3}{4}c$ condition derives from Pistolesi's theorem, which states that the zero-lift angle of attack is approximated by the tangent to the camber line in the $\frac{3}{4}c$ location.
This method is the so-called **Vortex Step Method** (VSM), and approximate LLT with

different contributions from the work of Munk [29], Pistolesi, Wieghardt and several others.
In particular, the VSM developed by Rannenberg ([33]) is considered for this thesis. It considers sweep and low aspect ratio and imposes the tangential flow condition at $\frac{3}{4}$c like Weissinger. As explained by Damiani ([14]), this method promises accurate solutions for both low and high aspect ratio wings of various shapes, including dihedral and sweep angle variation.
The main differences between the LLT and the VSM are summarized herein:

- The VSM approximates the resulting vortex sheet with a finite number of horseshoe vortex;
- The control points locations, differently from the LLT ($\frac{1}{4}c$), is at $\frac{3}{4}c$ but in the direction of the freestream.

The horsehoe vortices are visible in fig. 3.1 in which a fixed-wing kite is presented with a finite series of horseshoe vortex distributed along the wing.

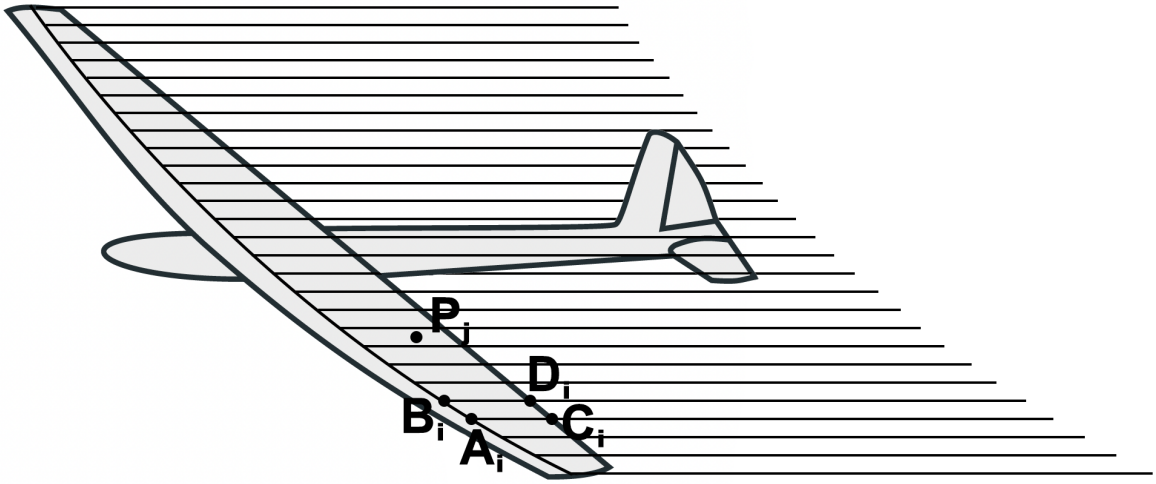


Figure 3.1: Horseshoe vortices distributed following the convention of the VSM. A_i and B_i are the starting points of the bound filament positioned at $\frac{1}{4}c$. C_i and D_i represent the end of wing panel and are situated on the TE. P_j is a generic panel control point located at $\frac{3}{4}c$

.

Moreover, by imposing the nonpenetrating wall condition at $\frac{3}{4}c$, the incidence angle in

the control point is equal to:

$$\alpha_{in} = \alpha_G - \arctan\left(\frac{dx}{dy}\Big|_{\frac{3}{4}c}\right) \tag{3.1}$$

A scheme is visible in fig. 3.2

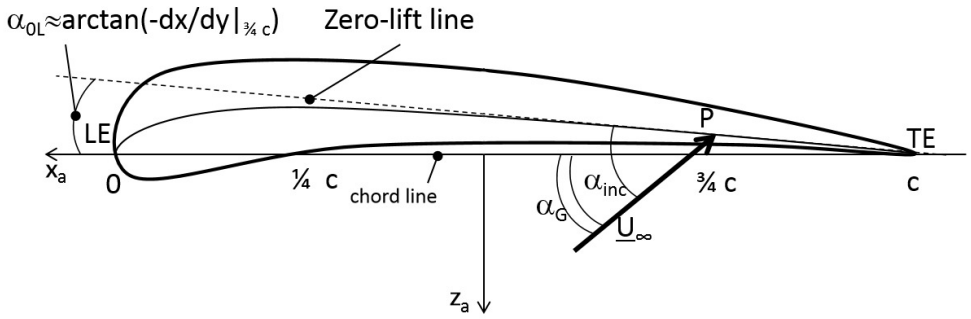


Figure 3.2: Representation of Pistolesi's theorem with a generic arifoil. With local airfoil reference frame displayed. [14]

With α_{0L} known from Pisotlesi's theorem being equal to: $\alpha_{0L} = \arctan(-\frac{dx}{dy}|_{\frac{3}{4}c})$

Assumptions: The assumptions made regarding the flow properties of this method are the same used in [12], reported herein for completion:

- The flow can be divided into two regions (inner and outer). The flow on the inner region represents the airfoil properties while the flow outside this region is inviscid, irrotational and incompressible in order to reimain in the potential flow definition;
- The KJ theorem is fulfilled in each wing section, linking the two regions;
- The flow is quasi-steady, every flow condition is solvable in the spatial domain only;
- The starting vortex is far downstream, and its influence can be neglected.

The induced velocity at the control point location (P_j) can be obtained by summing the contributions from all horseshoe vortices with associated circulation Γ_i. The control point, together with a clear representation of the bound circulation are visible in fig. 3.3.

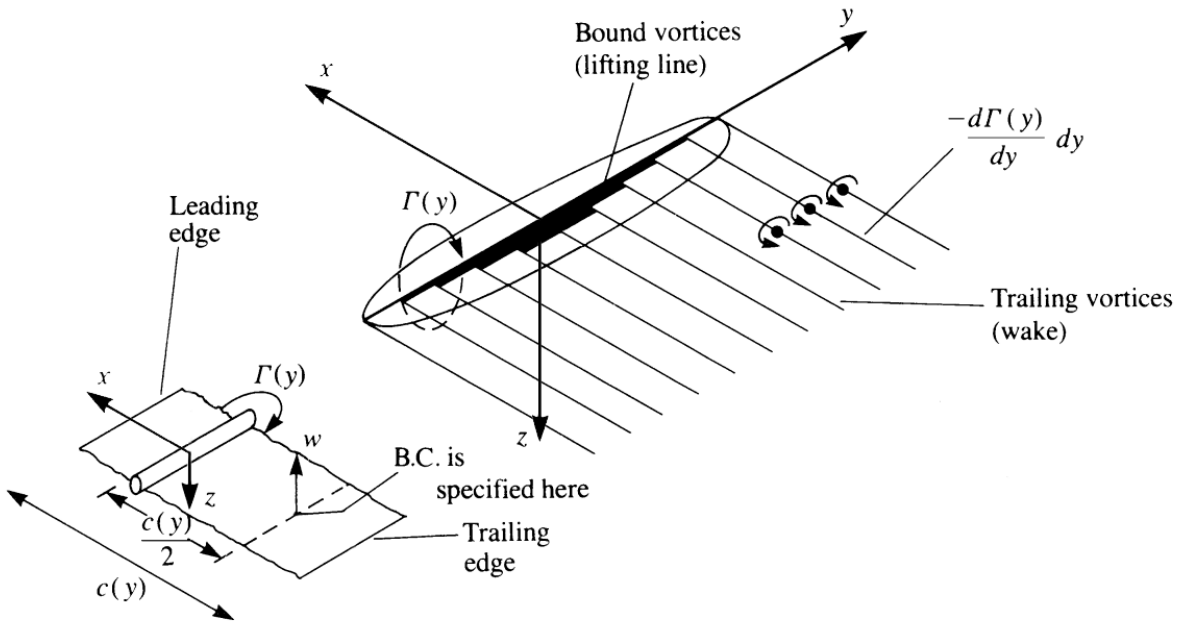


Figure 3.3: Representation of the lifting line horseshoe vortex on the wing (right) and bound circulation with B.C.(on the left): [24]

In this thesis, the reference frame follows a specific flight mechanic convention to be consistent with the reference frame exploited in the VLM. This reference frame will allow for a spatial mesh-to-mesh mapping between the aerodynamics and most of the structural dynamics program in future dynamic simulations.
Furthermore, the aerodynamic code uses the same reference frame to relate the 3D flow to the 2D airfoil polars.

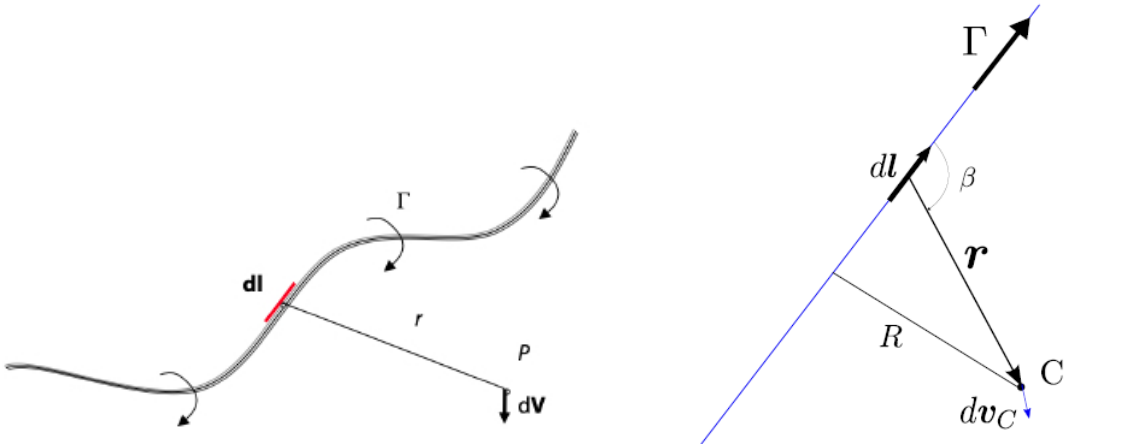
3.2.1. Computation of induced velocities and vortex core correction

The magnitude and direction of the flow field induced by a vortex filament can be computed with Biot-Savart Law (BS). This law is one of the most fundamental relations in the inviscid, incompressible flow theory. A mathematical expression describing how a vortex filament induces a flow field in the surrounding space can be achieved thanks to BS. The velocity induced by a segment (dl) to an arbitrary point c is defined by eq. (3.2)

$$d\mathbf{v}_C = \frac{\Gamma}{4\pi}\frac{d\mathbf{l}\times\mathbf{r}}{|\mathbf{r}|^3}, \quad \mathbf{v}_C = \frac{\Gamma}{4\pi}\int_1^2 \frac{d\mathbf{l}\times\mathbf{r}}{|\mathbf{r}|^3} \tag{3.2}$$

The graphical representation of the previous equation is visible in fig. 3.4a for a general

case with a curved three dimensional vortex filament and in fig. 3.4b with a generic straight segment.



(a) Curved 3D vortex filament of strength Γ.

(b) Biot-Savart Law with a generic straight segment of strength Γ.

Figure 3.4: Representations of the Biot-Savart law.

The analytical representation of BS referring to fig. 3.5 is visible in eq. (3.3). In which the airspeed $\mathbf{U}_{A,B}$ generated by a straight vortex filament of strength Γ is computed, considering just the vortex present on the filament **AB**. Starting from eq. (3.2), and by a matter of analytical substitution, eq. (3.3) can be obtained.

$$\mathbf{U}_{A,B}(\mathrm{P}) = \frac{\Gamma}{4\pi} \frac{\mathbf{AP} \times \mathbf{BP}}{|\mathbf{AP} \times \mathbf{BP}|^2} \left[\mathbf{AB} \cdot \left(\frac{\mathbf{AP}}{|\mathbf{AP}|} - \frac{\mathbf{BP}}{|\mathbf{BP}|} \right) \right] \tag{3.3}$$

For a straight filament along the direction of the airspeed, going from point C to infinity ($P_\infty = A + s\hat{v_a}$), where $\hat{v_a}$ is the unitary vector of airspeed, the induced velocity is:

$$\begin{aligned} \mathbf{U}_{A,\infty}(P) &= \lim_{s\to\infty} \frac{\Gamma}{4\pi} \left(s\hat{v_a}^T \left(\frac{A + s\hat{v_a} - P}{|A + s\hat{v_a} - P|} - \frac{A - P}{A - P} \right) \frac{(A - P) \times s\hat{v_a}}{|(A - P) \times s\hat{v_a}|^2} \right) \\ &= -\frac{\Gamma}{4\pi} \left(1 + \hat{v_a}^T \frac{\mathbf{AP}}{|\mathbf{AP}|} \right) \frac{\mathbf{AP} \times \hat{v_a}}{|\mathbf{AP} \times \hat{v_a}|^2} \end{aligned} \tag{3.4}$$

By Isolating a single horseshoe vortex geometry, it is easier to describe the formulae to compute the velocity induced by the generic *i-th* horseshoe vortex.

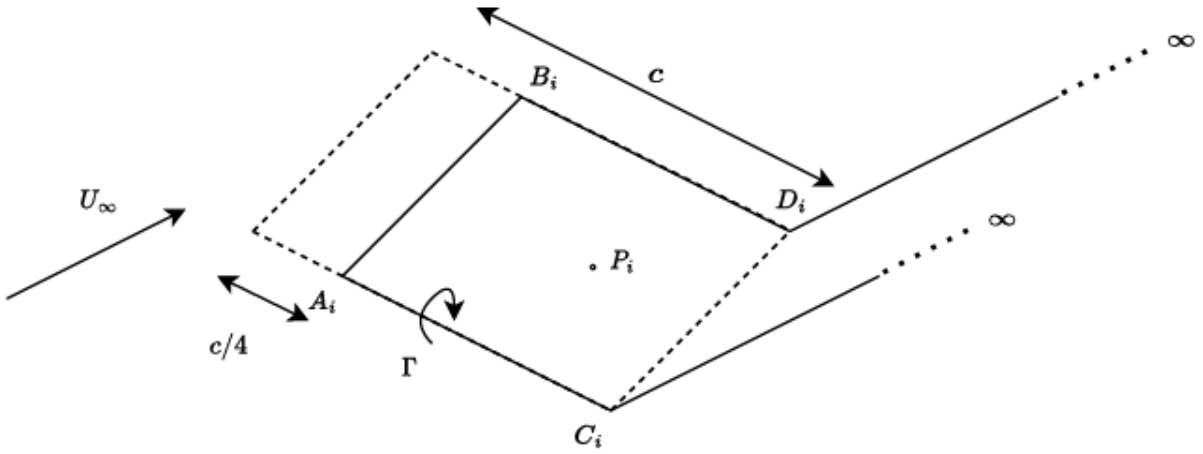


Figure 3.5: Horseshoe vortex geometry represented with straight lines - Source:[12]

As visible in fig. 3.5, in the VSM implemented in this thesis, a particular geometry for the horseshoe vortex is used, in which the two trailing vortices (A_iC_i and B_iD_i) follow the airfoil chordwise direction until the trailing edge, and from there, they follow the direction of the inflow velocity towards infinity. For more information and a detailed description, please refers to [33].
The more straightforward case with three filaments is also available in the code. Damiani proposed it ([14]), and consists in two trailing vortices that are shed from the ends of the bound vortex towards infinity.
Using a formulation with **five** filaments, the velocities induced by a horseshoe i to a point j could be described using eq. (3.5), easily achievable from fig. 3.6.

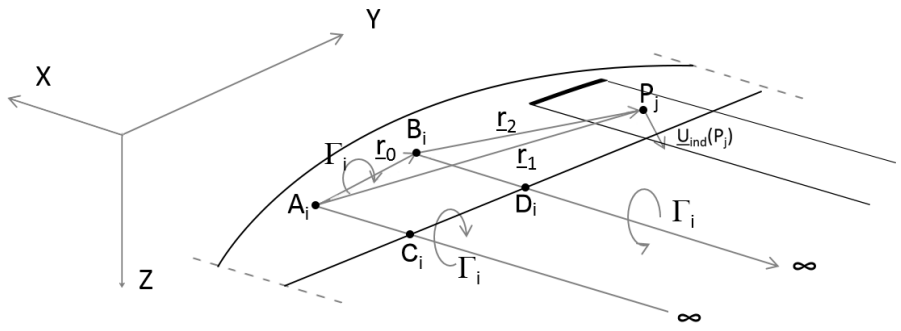


Figure 3.6: Contribute of the generic $i-th$ vortex to the induced velocity at the generic $j-th$ control point ([14])

The points A_i, B_i, C_i, D_i and P_j are visible in fig. 3.5. The equation is the following (eq. (3.5)):

$$
\begin{aligned}
\mathbf{U}_{A_iB_i}(\mathrm{P}_j) &= \begin{cases} \frac{\Gamma}{4\pi}\frac{\mathbf{AP}\times\mathbf{AP}}{|\mathbf{AP}\times\mathbf{BP}|^2}\left[\mathbf{AB}\cdot\left(\frac{\mathbf{AP}}{|\mathbf{AP}|}-\frac{\mathbf{BP}}{|\mathbf{BP}|}\right)\right] & \text{if } \frac{|\mathbf{AP}\times\mathbf{AB}|}{|\mathbf{AB}|}>\varepsilon_1 \\ \frac{|\mathbf{AP}\times\mathbf{AB}|}{|\mathbf{AB}|\varepsilon_1}\mathbf{U}_{A_iB_i}(\mathrm{P}'_j) & \text{otherwise} \end{cases} \\
\mathbf{U}_{C_iA_i}(\mathrm{P}_j) &= \begin{cases} \frac{\Gamma}{4\pi}\frac{\mathbf{CP}\times\mathbf{AP}}{|\mathbf{CP}\times\mathbf{AP}|^2}\left[\mathbf{CA}\cdot\left(\frac{\mathbf{CP}}{|\mathbf{CP}|}-\frac{\mathbf{AP}}{|\mathbf{AP}|}\right)\right] & \text{if } \frac{|\mathbf{CP}\times\mathbf{CA}|}{|\mathbf{CA}|}>\varepsilon_2 \\ \frac{|\mathbf{CP}\times\mathbf{CA}|}{|\mathbf{CA}|\varepsilon_2}\mathbf{U}_{C_iA_i}(\mathrm{P}'_j) & \text{otherwise} \end{cases} \\
\mathbf{U}_{B_iD_i}(\mathrm{P}_j) &= \begin{cases} \frac{\Gamma}{4\pi}\frac{\mathbf{BP}\times\mathbf{DP}}{|\mathbf{BP}\times\mathbf{DP}|^2}\left[\mathbf{BD}\cdot\left(\frac{\mathbf{BP}}{|\mathbf{BP}|}-\frac{\mathbf{DP}}{|\mathbf{DP}|}\right)\right] & \text{if } \frac{|\mathbf{BP}\times\mathbf{BD}|}{|\mathbf{BD}|}>\varepsilon_2 \\ \frac{|\mathbf{BP}\times\mathbf{BD}|}{|\mathbf{BD}|\varepsilon_2}\mathbf{U}_{B_iD_i}(\mathrm{P}'_j) & \text{otherwise} \end{cases} \\
\mathbf{U}_{A_i\infty}(\mathrm{P}_j) &= \begin{cases} \frac{\Gamma_i}{4\pi}\frac{1+\frac{\mathbf{AP}\cdot v_a}{r1}}{|\mathbf{AP}\times\hat{v}_a|^2}\mathbf{AP}\times\hat{v}_a & \text{if } |\mathbf{AP}\times\hat{v}_a|>\varepsilon_2 \\ \frac{|\mathbf{AP}\times\hat{v}a|}{\varepsilon_2}\mathbf{U}_{A_i\infty}(\mathrm{P}'_j) & \text{otherwise} \end{cases} \\
\mathbf{U}_{B_i\infty}(\mathrm{P}_j) &= \begin{cases} \frac{-\Gamma_i}{4\pi}\frac{1+\frac{\mathbf{BP}\cdot v_a}{r1}}{|\mathbf{BP}\times\hat{v}_a|^2}\mathbf{BP}\times\hat{v}_a & \text{if } |\mathbf{BP}\times\hat{v}_a|>\varepsilon_2 \\ \frac{-|\mathbf{BP}\times\hat{v}a|}{\varepsilon_2}\mathbf{U}_{B_i\infty}(\mathrm{P}'_j) & \text{otherwise} \end{cases}
\end{aligned}
\tag{3.5}
$$

Where A,B,C, and D has been introduced before, ε_1 and ε_2 are the core radius of the bound vortex and trailing vortices, respectively. P'_j is the radial projection of P_j and it is needed for the vortex core correction.
Using a formulation with **three** filaments, the velocities induced by the horseshoe vortex

are described in eq. (3.6).

$$\mathbf{U}_{A_iB_i}(\mathrm{P}_j) = \begin{cases} \frac{\Gamma}{4\pi}\frac{\mathbf{AP}\times\mathbf{AP}}{|\mathbf{AP}\times\mathbf{BP}|^2}\left[\mathbf{AB}\cdot\left(\frac{\mathbf{AP}}{|\mathbf{AP}|}-\frac{\mathbf{BP}}{|\mathbf{BP}|}\right)\right] & \text{if } \frac{|\mathbf{AP}\times\mathbf{AB}|}{|\mathbf{AB}|} > \varepsilon_1 \\ \frac{|\mathbf{AP}\times\mathbf{AB}|}{|\mathbf{AB}|\varepsilon_1}\,\mathbf{U}_{A_iB_i}(\mathrm{P}'_j) & \text{otherwise} \end{cases}$$

$$\mathbf{U}_{A_i\infty}(\mathrm{P}_j) = \begin{cases} \frac{\Gamma_i}{4\pi}\frac{1+\frac{\mathbf{AP}\cdot\hat{v}_a}{r1}}{|\mathbf{AP}\times\hat{v}_a|^2}\mathbf{AP}\times\hat{v}_a & \text{if } |\mathbf{AP}\times\hat{v}_a| > \varepsilon_2 \\ \frac{|\mathbf{AP}\times\hat{v}_a|}{\varepsilon_2}\,\mathbf{U}_{A_i\infty}(\mathrm{P}'_j) & \text{otherwise} \end{cases}$$

$$\mathbf{U}_{B_i\infty}(\mathrm{P}_j) = \begin{cases} \frac{-\Gamma_i}{4\pi}\frac{1+\frac{\mathbf{BP}\cdot\hat{v}_a}{r1}}{|\mathbf{BP}\times\hat{v}_a|^2}\mathbf{BP}\times\hat{v}_a & \text{if } |\mathbf{BP}\times\hat{v}_a| > \varepsilon_2 \\ \frac{-|\mathbf{BP}\times\hat{v}_a|}{\varepsilon_2}\,\mathbf{U}_{B_i\infty}(\mathrm{P}'_j) & \text{otherwise} \end{cases} \tag{3.6}$$

All the previous cases have a singularity as the point P approaches the vortex filament, where the velocity tends to infinity as the radius goes to zero. In order to avoid it, a **vortex core correction** is applied in which a vortex core radius is given to each filament.
The vortex core radius ε_1 for the trailing vorticity is given by eq. (3.7)

$$\varepsilon_1 = \sqrt{4\alpha_0\nu\frac{\mathbf{r}_\perp}{U_\infty}} \tag{3.7}$$

Where:

- ν is the air kinematic viscosity;
- α_0 is the Oseen parameter ([8]), equal to 1.25643;
- $r_\perp$ is the position vector from either A_i, B_i, C_i or D_i to the projection of P_j onto the respective vortex centerline: $\mathbf{r}_\perp = (\mathbf{r}\cdot\xi)\,\xi$.

The vortex core correction analytic characterization is visible in fig. 3.7, in which the vortex boundaries and core are represented together with the geometric parameters.

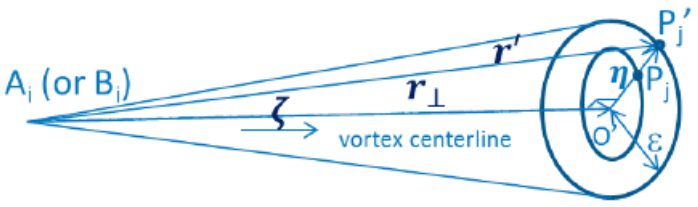


Figure 3.7: Trailing vortex core correction geometric parameters: [14]

3.2.2. Solving for the Circulation Distribution

Aerodynamic influence coefficient (AIC) matrix The purpose of the AIC matrix is to achieve a way to calculate the relative velocity in each wing section in a fast and efficient way. The dimension is NxN with N number of horseshoe vortices. Each matrix element represents the velocity a horseshoe vortex induces to a certain control point, assuming a unit circulation strength of all vortices.
Since the AIC matrix is a function of the Mach number and, sometimes, the set of the initial boundary condition approximated, it is not rare that the computation of the AIC matrix could lead to significant error in the transonic regime and when dealing with hinge moments. The transonic regime is completely outside of AWEs flight envelopes but it should be known that a possible premultiplying correction factor matrix can be used in the case a transonic regime must be analysed.
The user may introduce this matrix, which could be based on experimental data or CFD. The VSM introduced for the thesis is based on an iterative process, and solving the systems of equation presented in eq. (3.8) speeds up the iterative process considerably ([12]):

$$\begin{aligned} \vec{u} &= AIC_u\vec{\Gamma} \\ \vec{v} &= AIC_v\vec{\Gamma} \\ \vec{w} &= AIC_w\vec{\Gamma} \end{aligned} \tag{3.8}$$

This thesis focuses on a fixed-wing hard kite; therefore, the implementation of **the tail** is necessary. The AIC matrix is modified accordingly, introducing the cross-interaction between the tail and wing and vice versa.
During the first iteration of the program, the simplest possible case is presented with only wing and horizontal tail implemented.

Differently from the typical VSM, which enforces the slip condition: $\mathbf{U}_{rel} \cdot \mathbf{n}|_{\frac{3}{4}c} = 0$, we make use of the more generic lifting-line fundamental equation (eq. (3.9)) in order to create a constraint for $\Gamma(y)$:

$$\mathbf{f} = \rho\,|\mathbf{U}_\infty \times \mathbf{\Gamma}(y)| - \frac{1}{2}\,\rho\,|\mathbf{U}_{rel} \times \hat{z}_{airf}|^2\,c\,C_l(\alpha, \delta_f) = 0 \tag{3.9}$$

where:

- ρ is the air density;
- $\mathbf{U}_{inf}$ is the free stream velocity vector;

- $\mathbf{\Gamma}(y)$ is the circulation vector;
- $\mathbf{U}_{rel}$ is the relative air velocity
- $\hat{z}_{airf}$ is the unit vector along the airfoil z-axis;
- c is the chord length;
- $C_l(\alpha, \delta_f)$ is the 2D lift coefficient as a function of α (effective angle of attack seen by the airfoil) and δ_f (airfoil's flap or aileron deflection).

The non linearity is present in $C_l(\alpha, \delta_f)$ and in $\mathbf{U}_{rel}^2$ terms.
In eq. (3.9) the unknowns are $\bar{\Gamma}(y)$, U_{rel} and α. Since the latter two can be expressed as a function of the induced velocity, therefore, of Γ, the unknown is only one. By specifying a control point P_j, it is possible to write:

$$\mathbf{U}_{rel}(\mathrm{P}_j) = \mathbf{U}_{\infty}(\mathrm{P}_j) + \mathbf{U}_{ind}(\mathrm{P}_j) \tag{3.10}$$

$$\begin{aligned}\mathbf{U}_{ind}(\mathrm{P}_j) = \sum_i \mathbf{U}_{ind,i}(\mathrm{P}_j) = \sum_i [\mathbf{U}_{A_iB_i}(\mathrm{P}_j) + \mathbf{U}_{A_iC_i}(\mathrm{P}_j) + \\ \mathbf{U}_{B_iD_i}(\mathrm{P}_j) + \mathbf{U}_{D_i\infty}(\mathrm{P}_j) + \mathbf{U}_{C_i\infty}(\mathrm{P}_j)]\end{aligned} \tag{3.11}$$

$$\alpha_j = \arctan\left(\frac{\mathbf{U}_{rel,j} \cdot \hat{x}_{airf,j}}{\mathbf{U}_{rel,j} \cdot \hat{y}_{airf,j}}\right) \tag{3.12}$$

The term $C_l(\alpha, \delta_f)$ includes the generic polar curve of an airfoil. The power of this method lies here, the VSM can account for **nonlinear** effects of the airfoil (beyond the simple flat plate) and also considers the presence of flap or moving-surfaces deflection ([14]).
However, this method does not converge to a correct solution as it is written. According to [14], the reason is that, in the lifting line sense, eq. (3.9) should be enforce at the $\frac{1}{4}c$ control point. However, this is not the case for this VSM since the control point location is at $\frac{3}{4}c$ in order to account for the effect of camber.
The significant modification introduced to obtain a convergence is related to the addition of a corrective two-dimensional contribution to the induction. The new term can be represented and computed as in eq. (3.13) and eq. (3.14):

$$\begin{aligned}\mathbf{U}_{ind,i}(\mathrm{P}_j) = \mathbf{U}_{A_iB_i}(\mathrm{P}_j) + \mathbf{U}_{A_iC_i}(\mathrm{P}_j) + \mathbf{U}_{B_iD_i}(\mathrm{P}_j) + \\ \mathbf{U}_{D_i\infty}(\mathrm{P}_j) + \mathbf{U}_{C_i\infty}(\mathrm{P}_j) - \mathbf{U}_{A_iB_i2D}\end{aligned} \tag{3.13}$$

With:

$$\mathbf{U}_{A_iB_i2D} = \frac{\Gamma_i}{2\pi} \frac{\mathbf{r_0} \times \mathbf{r_3}}{|\mathbf{r_0} \times \mathbf{r_3}|} r_0 \, \delta_{ij} \tag{3.14}$$

The equations are derived from fig. 3.8. In which is possible to see:

- $\mathbf{r_3}$ is the position vector from the midpoint of A_iB_i to P_j
- $\mathbf{U}_{A_iB_i2D}$ is the contribution to the induced velocity by the 2D bound vorticity aligned with the segment $\mathrm{A}_i\mathrm{B}_i$, but it is considered only for control point within the local vortex element. In fact, the kronecker's delta is present in the formula.

In this way, eq. (3.9) can be enforced at each control point where the relative velocity (U_{rel}), the angle of attack (α) and the lift coefficient (C_l) are expressed in terms of Γ_i.

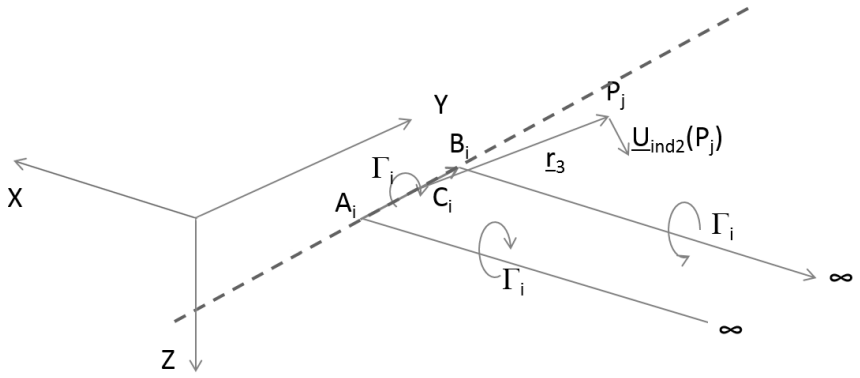


Figure 3.8: Contribute of the generic $i-th$ vortex to the induced velocity at the generic $j-th$ control point: [14]

The resulting non-linear system of equations will be solved exploiting a numerical method. For the purpose of this thesis, given the potentiality of the MATLAB environment, the function fsolve is used.
Further developments could open the way to different numerical analysis and modification of the numerical solver.

3.2.3. 2D airfoils database and interpolation

Finally, the nonlinear VSM implemented for this thesis uses airfoils data to account for **viscosity's effects** and **changes in the local airfoil geometry**.
Several possible models are analyzed:

- **2D panel method**, this method is exploited in the airfoil analysis code called XFOIL [18]. This model shows excellent accuracy compared with higher fidelity methods when analysing common airfoil shapes.
- **CFD**, if an extensive database of higher fidelity analysis is available.

For the purpose of this thesis, a simplified model is used, introducing arbitrary constant values for α_{airf}, C_l, C_D and C_m. The introduction of a CFD database or 2D panel method could be considered for future developments of the method.

3.3. Implementation of the VSM

An entirely new code based in MATLAB is written. The inputs required from the aerodynamic code have been completely modified from the original Python code since the focus is on hard kites and the tail and the control surfaces must be considered:

- The apparent velocity dependency on the angle of attack and sideslip has been removed, introducing the relative airspeed, which depends on kite speed in the body reference frame and wind speed; From a kite analysis point of view, this is a better solution, allowing for a finer analysis and helps in catching specific dynamics of the aircraft;
- the geometry of the kite, in which several modifications are introduced:
  - The kite is rigid and not inflatable, the kites presented in section 3.1 are used;
  - Several modifications to the geometry are introduced, in particular, a function returning the coordinates of a swept wing with a diheadral angle and elliptic chord distribution;
  - The horizontal and vertical tails are introduced;
  - Control surfaces are introduced.
- The billowing is not considered;
- The spanwise airfoil geometry, defined by the sectional thickness and camber, is not necessary for the purpose of the thesis.

3.3.1. Logic of the aerodynamic module

The internal logic of the aerodynamic code is presented and follows the scheme visible in fig. 3.9:

- **Generate the coordinates** of the wing and tail, relying upon a specific kite geometry. The coordinates are inserted into N structures (ranging from one to three, recalling the traditional lifting surfaces) to keep the program computationally efficient and clean;
- **Create the wing and tail's geometry** and the definition of vortex filaments, control points and relevant vectors for each section. All these parameters are put into specific structures (w - wing, h - horizontal tail and v - vertical tail) to keep the program computationally efficient and clean while making them easily extractable;
- **Generate the arfoils** along the various lifting surfaces, introducing the values for α_{airf}, C_{l_0}, C_{D_0} and C_{m_0}.
- **Setup of the AIC matrix**, including, if needed, the induction of the tail;
- **Initialize the circulation (Γ) distribution** and begin the iterative process;
- **Compute the relative velocity at each wing section**, with the previous circulation distribution, to obtain the relative angle of attack at each section;
- **Interpolate** the aerodynamic coefficients from 2D airfoil data for each wing section. Then use the coefficients to compute the circulation at each wing section thanks to KJ theorem;
- **Perform a check for convergence criteria**, if it does not converge the program restart from point 3, computing Γ as a combination of the old and new circulation, using a relaxation factor to stabilize the solution. The solution is achieved through several iterations of matlab 'fsolve' function;
- Once the circulation, forces and directions are computed, **integrate along the wing** to obtain global force coefficients of the wing and post process the results.

A flowchart is visible in fig. 3.9; the first green block represents the inputs (numerical and strings to be set before launching the code). Simply by changing an input string is possible to switch from the VSM to the LLT, also implemented in the code. The three parallel green boxes represent the next step of creating the coordinates, generating the geometry and the airfoils.
The blue blocks represent the 'hard' computational core of the program, starting with the setup of the AIC matrix according to the number of lifting surfaces, followed by initialisation of the circulation, computation of the induced velocity using BS and interpolation of the aerodynamic coefficients exploiting airfoil polars. Finally, if the convergence is met, the integration of the forces along the lifting surfaces is done, concluding with the

post-process.

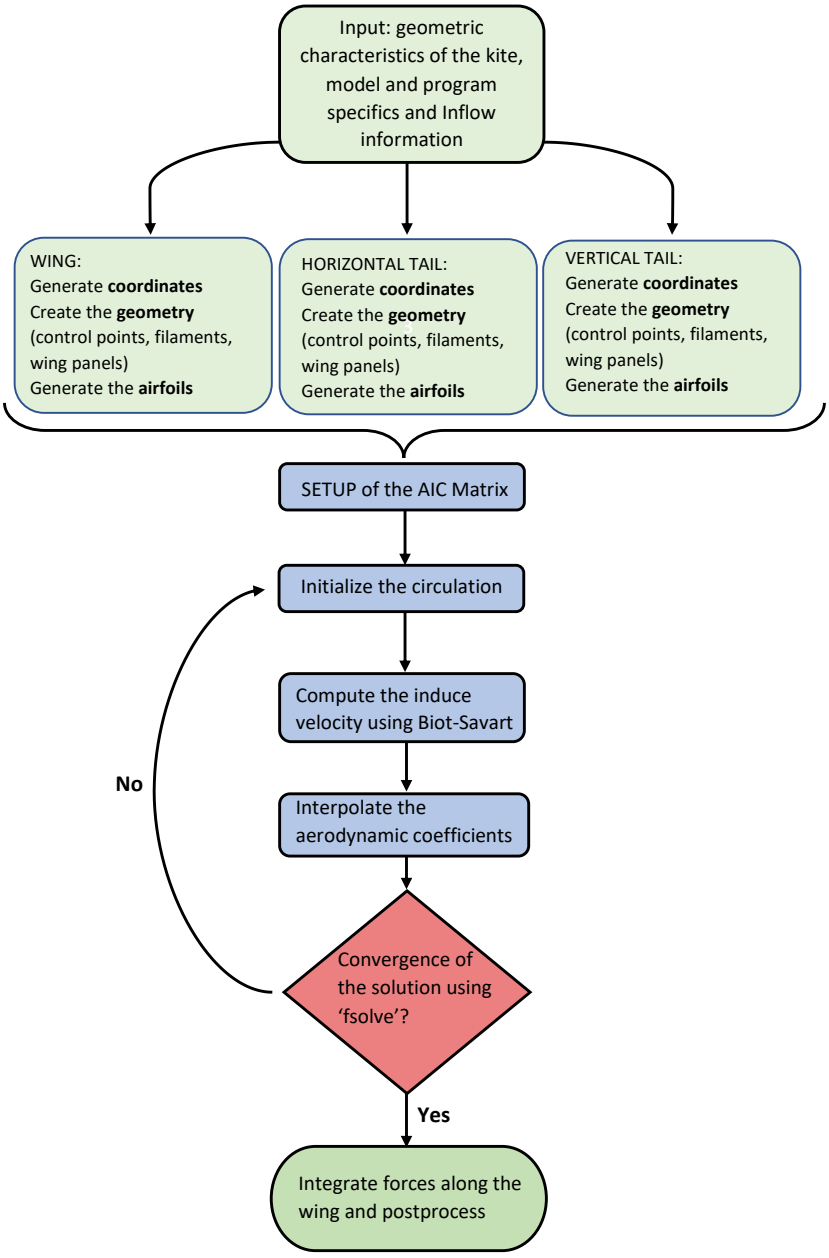


Figure 3.9: Flowchart of the VSM

3.4. Vortex Step Method Validation

Implementing a new MATLAB code requires validation to verify that the implemented model can predict forces and moments and if it is computationally efficient to be exploited during conceptual design, where many analyses must be launched and versatile, allowing for the variation of several parameters.

3.4.1. Elliptic wing - Comparison with analytical LLT

The analytical solution for an elliptic wing is known. Therefore, validation could be performed. The geometry of the wing is the one of the Zefiro kite (visible in table 3.2), with a wing AR of 16.17. The initial simplified case is performed with the same wing removing sweep and dihedral angle and without considering the horizontal tail.
Several aerodynamic sections along the wing are tested, and the kite is moving with a single airspeed component along x (in body axis) that assumes values of 20, 45 and 60 m/s. For every V_x variation, a wind speed is introduced with a single component along z to excite the lift, ranging from -4 m/s up to -10 m/s with a -1 m/s step. Furthermore, a variation in the number of sections along the wing is introduced.

The analytical solution for an elliptic wing exploiting LLT can be obtained easily from an aerodynamic book. The lift and induced drag coefficients can be computed by:

$$C_L = \frac{2\pi}{1 + \frac{2}{AR}}\alpha \tag{3.15}$$

$$C_{Di} = \frac{1}{\pi AR\, e}C_L^2 \tag{3.16}$$

where:

- α is the inflow angle of attack;
- AR is the aspect ratio of the wing
- e is an efficiency factor, for an ellipse it is unitary otherwise it is lower than 1.

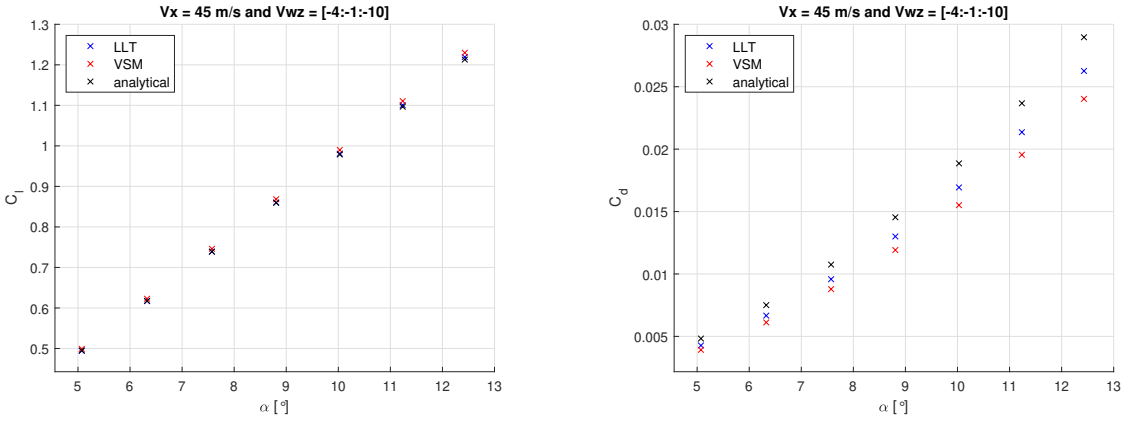
Hereinafter, the paramount results are presented through a series of graphs, especially:

1. In fig. 3.10 it is possible to see a comparison between C_{l_α} for the analytical LLT, the LLT obtained from the code and the VSM obtained from the code, together with the same representation for C_{d_α};
2. The other two cases in which the V_x have been changed to 60 m/s and 20 m/s are

presented in fig. B.3 and fig. B.4 respectively.

3. A test case with $V_x = 45$ m/s is also presented while varying the Aspect Ratio. The behaviour of the drag coefficient with respect to the angle of attack is shown in fig. 3.11.

4. The case with $V_x = 45$ m/s is selected for an additional test to study how the program adapts while varying the number of aerodynamic sections along the wing (results are visible in fig. 3.12 and fig. 3.13).



(a) Lift coefficient with respect to angle of attack variation for an elliptic wing.

(b) Drag coefficient with respect to angle of attack variation for an elliptic wing.

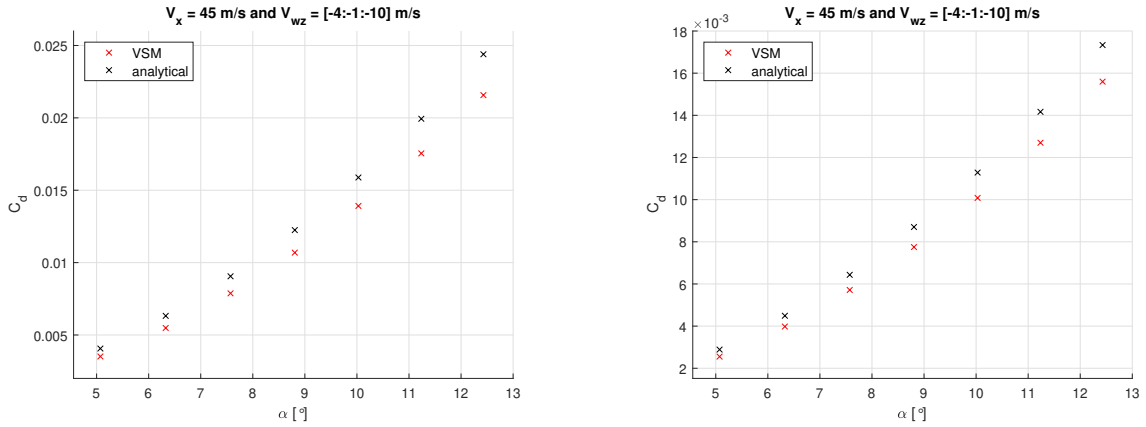
Figure 3.10: Elliptic wing - case with Vx = 45 m/s , variable wind speed along z-axis and 11 aerodynamic sections along the wing.

In fig. 3.10a the variation of lift coefficient with respect to the angle of attack is presented. The percentage error is equal to 0.8 % for $\alpha = 5°$ and reaches a value of 1.3 % for $\alpha = 12.5°$ for the VSM. Regarding the LLT, it is even more precise, with a 0.5 % of error for $\alpha = 12.5°$.

In fig. 3.10b the variation of drag coefficient with respect to the angle of attack is presented. The percentage error is equal to 18 % for $\alpha = 5°$ and is reduced to 17 % for $\alpha = 12.5°$ for the VSM. Regarding the LLT the error is smaller, with a 9 % for $\alpha = 12.5°$. This is a big discrepancy, especially for the VSM, but the discretisation is small, therefore several increments of it are tested later on.
In appendix B the same analysis for $V_x = 60$ and $V_x = 20$ m/s are reported (fig. B.3 and fig. B.4).

Moreover, the test case with $V_x = 45$ m/s is performed with an aspect ratio equal to 16, and the increase of the aspect ratio is tested to verify if the results obtained from the

VSM will tend to reach the value of the LLT. The increase is kept in a reasonable range of AR, and the results from aspect ratio equal to 20 and 30 are presented below.



(a) Drag coefficient with respect to angle of attack variation for an elliptic wing with AR = 20.

(b) Drag coefficient with respect to angle of attack variation for an elliptic wing with AR = 30.

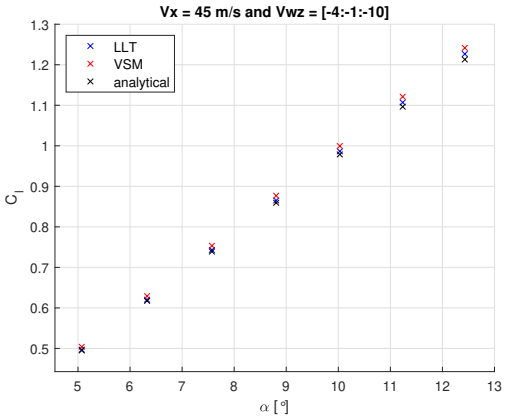
Figure 3.11: Drag coefficient with respect to angle of attack variation (AR = 20 on the left and AR = 30 on the right).

In fig. 3.11a, the percentage error for $\alpha = 5°$ is reduced to 13.5 % and 11.6 % for $\alpha = 12.5°$. By further increasing the AR up to a value of 30, the percentage error for $\alpha = 5°$ is equal to 11.6 % and 10 % for $\alpha = 12.5°$.
While increasing the AR the results are tending to the analytical LLT values. The errors are summarized in table 3.3:

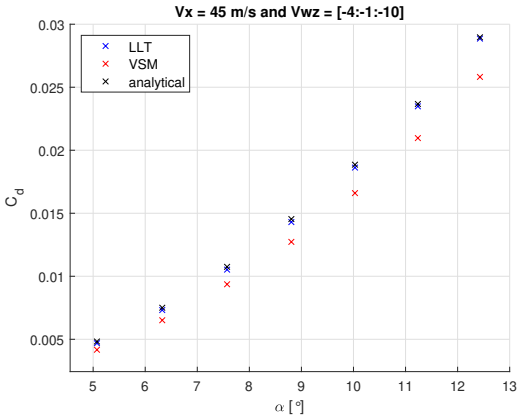
Table 3.3: C_{D_α} error summary with varying Aspect Ratio.

	AR = 16	AR = 20	AR = 30
% Error for $\alpha = 5°$	18 %	13.5 %	11.6 %
% Error for $\alpha = 12.5°$	17 %	11.6 %	10 %

The following graphs represents the same variation with a $V_x = 45$ m/s, the same variation for V_{wz}, but with an increase in the discretisation along the wing.



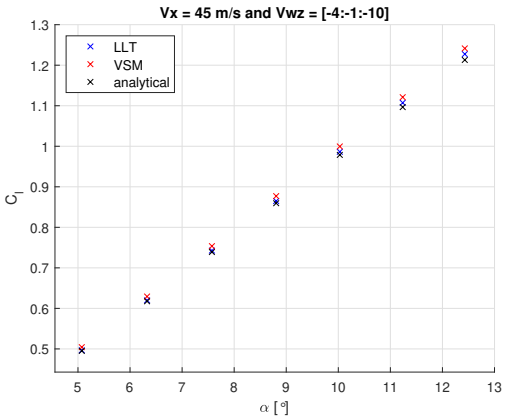
(a) Lift coefficient with respect to angle of attack variation for an elliptic wing.



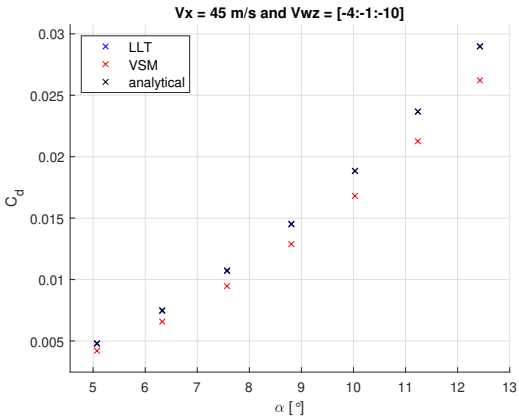
(b) Drag coefficient with respect to angle of attack variation for an elliptic wing.

Figure 3.12: Elliptic wing - case with Vx = 45 m/s , variable wind speed along z-axis and 21 aerodynamic sections along the wing.

From fig. 3.12 it is visible how the discrepancy in the drag coefficient values is far minor using the LLT included in the code, leading to very similar results compared to the analytic LLT, with a percentage error of 0.36 % for the higher angle of attack in the graph. The VSM still presents a discrepancy, but this is due to the different control points in which the forces are evaluated, specifically, the $\frac{3}{4}c$ for the Vortex Step Method.



(a) $C_l - \alpha$ Elliptic wing.



(b) $C_d - \alpha$ Elliptic wing.

Figure 3.13: Elliptic wing - case with Vx = 45 m/s , variable wind speed along z-axis and 31 aerodynamic sections along the wing.

The resulting errors for the previous cases are summarized in table 3.4:

Table 3.4: C_{D_α} error summary.

	N = 11	N = 21	N = 31
% Error using LLT	9 %	0.36 %	0.23 %
% Error using VSM	17 %	10.7 %	9.3 %

The results are reported in a logarithmic graph in fig. 3.14

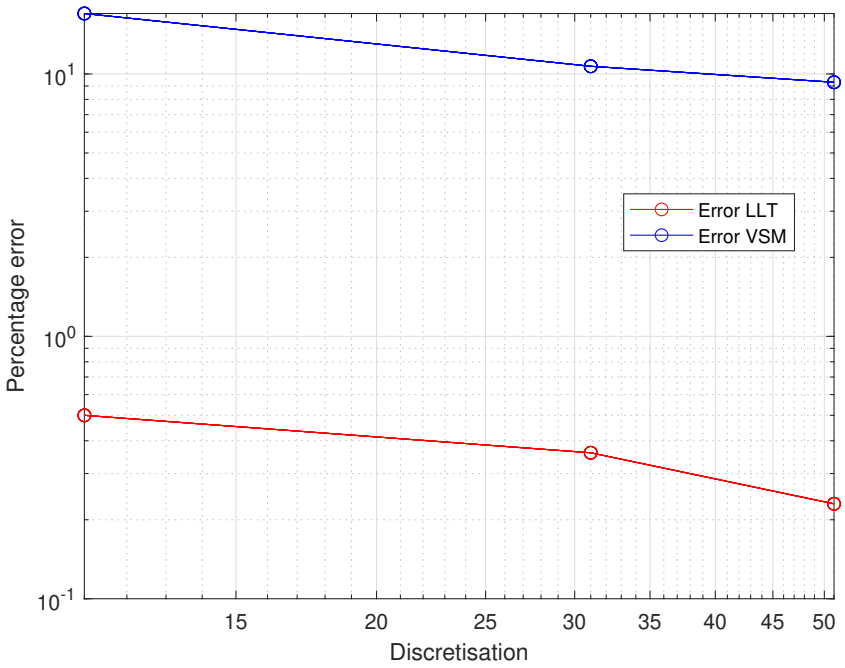


Figure 3.14: % Error reduction while increasing the discretisation - logarithmic representation.

The error of the LLT is reducing a lot, reaching very low values with a discretization of just 21 sections. The VSM is reducing with a slower rate, with a tendency of stabilising on a certain value. This was expected since LLT and VSM are different methods that works with different control points.
It is challenging to state the model's effectiveness only considering the previous validation since the comparison is made with the analytical LLT. However, in classical LLT, the drag coefficient is generally overestimated, therefore the difference visible in both fig. 3.10a and fig. 3.10b is expected. This effect is more visible at lower AR; the percentage error is reduced while increasing the AR as reported in table 3.3.
According to Pistolesi's theorem, it is possible, by implementing the boundary conditions,

to obtain the correct distribution of the magnitude of the forces. However, this does not mean correct orientation of the forces.
According to [12], two important conclusions can be dragged for VSM:

- The magnitude of the quasi-steady lift should be determined by the flow at three-quarter chord location (same as Pistolesi's theorem);
- The direction of the quasi-steady aerodynamic forces should be determined by the flow at the quarter chord location.

3.4.2. Elliptic wing with dihedral and sweep

The second case of validation is related to the introduction of the dihedral angle and sweep angle on the same wing presented before. The values for the angles are equal to the values of Zefiro kite. The wing is visible in fig. 3.15 and fig. 3.16

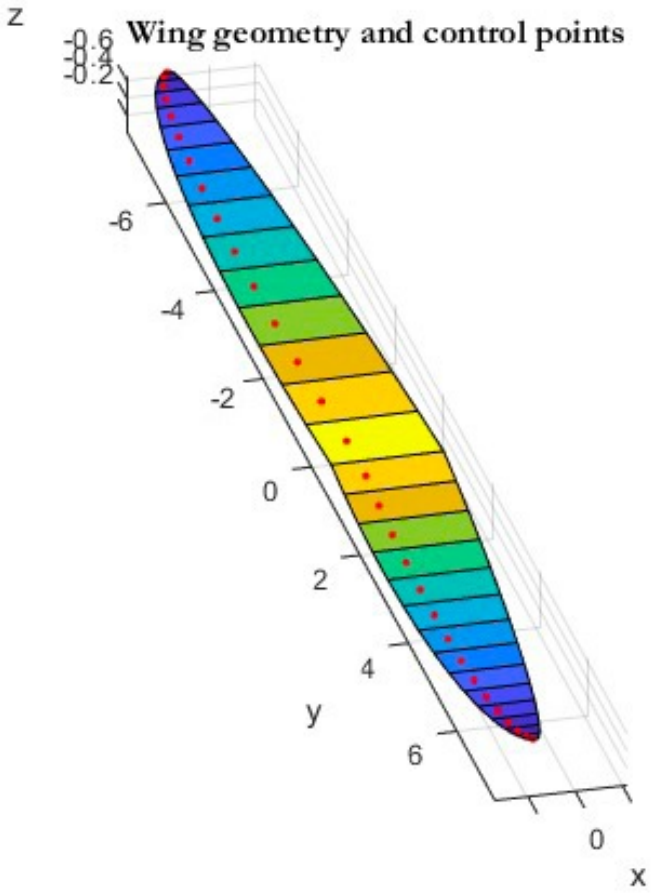


Figure 3.15: Elliptic wing with dihedral and sweep angle.

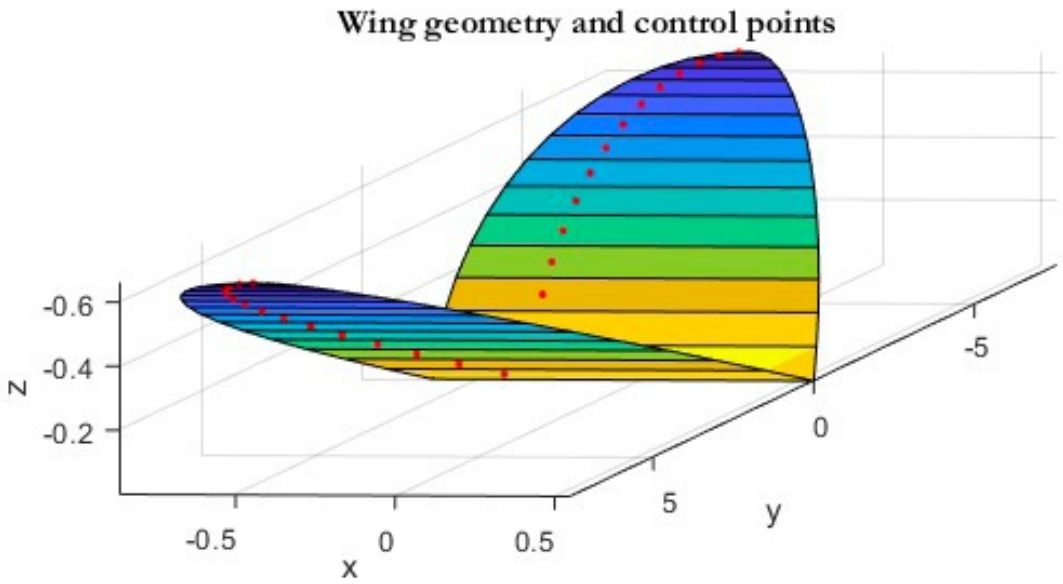


Figure 3.16: Elliptic wing with dihedral and sweep angle - different point of view.

Introducing the dihedral angle is fundamental, especially for the conceptual design of AWEs. It stabilises the aircraft around the roll axis, and an AWE system is subjected to a very chaotic environment while producing energy in a circular or eight-shaped trajectory. An aerodynamic model able to incorporate this angle correctly could be exploited for many initial analyses.
The same goes for the sweep angle; in fixed-wing AWEs, flight stability is most likely achieved thanks to the introduction of this angle.
In order to verify if the code can behave while these angles are modified, two validations were performed:

1. Representation of the aerodynamic moments relation with the dihedral angle;
2. A validation exploiting aerodynamic stability derivatives. Thanks to Roskam ([35]) and Sadray ([37]) analytic methods, a computation of the stability derivative concerning the sideslip is done. At the same time, a finite difference analysis is used to estimate the same derivatives in the program. A final comparison between the two

is carried on to verify if the code can respond to a sideslip modification. The same derivatives obtained with Tornado VLM are presented to justify the result further.

The introduction of the dihedral and sweep has been categorized with two different analyses.

The dihedral angle strongly influences the dihedral effects, which is the amount of roll moment produced in proportion to the amount of sideslip. Several sideslip angle variations are tested with a variation in the y-component of the aircraft speed, introducing a proper sideslip during the trajectory. The results are visible in fig. 3.17, in which an airspeed component along the y-axis has been added. The kite airspeed y-component (V_y) is modified, continuously increasing the sideslip angle and observing if, with a dihedral angle of 5°, the rolling moment is increased accordingly. The kite airspeed x-component (V_x) is kept constant. The results are visible in fig. 3.17

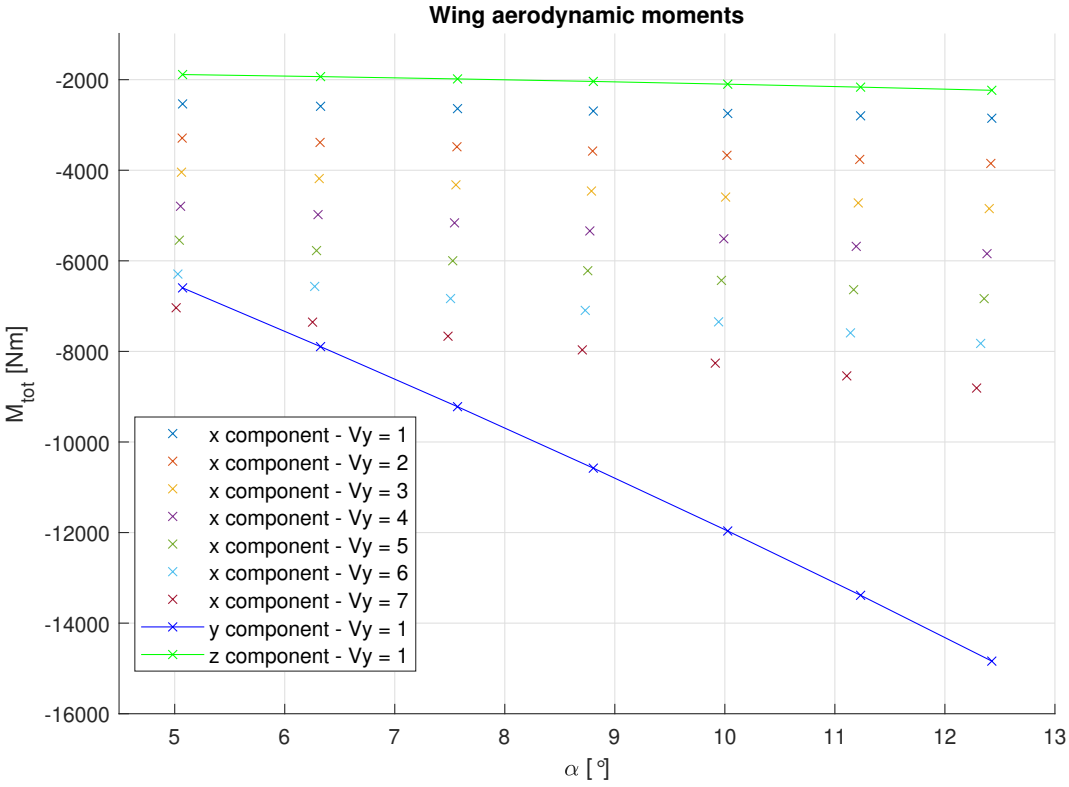


Figure 3.17: Moment components variation with respect to the angle of attack with a fixed dihedral angle of 5°.

It is visible how the rolling moment (x-component) is modified while changing the y-component of the kite airspeed. Therefore, the introduction of the dihedral as an input is correctly written by the code and the geometry is modified accordingly.

For the second test, it is possible to isolate the contribute of the wing, allowing for a quantitative comparison between the analytic stability derivative and the one estimated from the code. The derivative taken into considerations is the wing-fuselage stability derivative due to the angle of sideslip $C_{l_{\beta_{wf}}}$. Several simplifications are introduced for the analytic formulation related to the fuselage, since Zefiro does not have a proper fuselage defined.
For the initial comparison the dihedral angle is equal to 5°.
The initial value of the airspeed y-component (V_y) is equal to 2 m/s, the x-component (V_x) is equal to 45 m/s and the wind z-component (V_{wz}) is equal to -4 m/s. The results obtained from three different cases are reported, note that the Roskam analytic formula is simplified compared to the one present on the book since the tail and the fuselage are not considered in this case.
In particular, the results from Roskam is visible in eq. (3.17), obtained from [35].

$$\begin{aligned}\mathbf{Cl}_{\beta_{\mathbf{wf}}} = 57.3\,[C_{L_{wf}}\{(C_{l_\beta}/C_L)_{\Lambda_{c/2}}\,(K_{M_\Lambda})(K_f) + (C_{l_\beta}/C_L)_A\} \\ + \Gamma\{(C_{l_{\beta/\Gamma}})K_{M_\Gamma} + (\Delta C_{l_\beta}/\Gamma)\} = \quad \mathbf{-0.01702}\end{aligned} \tag{3.17}$$

With:

- $C_{L_{wf}}$ that is the lift coefficient of the wing fuselage combination. It could be taken equal to Cl_w;
- $(C_{l_\beta}/C_L)_{\Lambda_{c/2}}$ that is the wing sweep contribution;
- K_{M_Λ} is the compressibility correction factor, obtained from empirical estimations and more or less null;
- K_f is a fuselage correction factor.
- $\Lambda_{c/2}$ is the wing semi-chord sweep angle;
- $(C_{l_\beta}/C_L)_A$ is the aspect ratio contribution;
- Γ is the geometric dihedral angle of the wing;
- $(C_{l_{\beta/\Gamma}})$ is the wing dihedral effect;
- K_{M_Γ} is the compressibility correction to dihedral;
- $(\Delta C_{l_\beta}/\Gamma)$ is the fuselage induced effect on the wing height;

The result from Tornado is obtained exploiting the complete version with interface available on MATLAB: $C_{l_\beta}^{Tor} = \mathbf{-0.0199}$.

Finally the result estimated from the VSM with a starting y-component of the kite speed equal to 2 m/s and a step of 0.025 m/s is equal to: $C_{l_\beta}^{VSM} = \mathbf{-0.014}$.

Other situation have been tested while varying the dihedral angle on the wing:

Table 3.5: C_{L_β} comparison Tornado VLM - VSM - Roskam.

	Tornado VLM	**VSM**	**Roskam analytic**
C_{L_β} ($\Gamma = -5°$)	-0.019	-0.014	-0.017
C_{L_β} ($\Gamma = -7°$)	-0.131	-0.127	-0.135
C_{L_β} ($\Gamma = -9°$)	-0.167	-0.284	-0.197

The value obtained for $\Gamma = -9°$ is slightly higher than expected, but the stability derivatives estimated with the code are acceptable. Therefore, the code can consider the dihedral angle as a geometric input and return results that are modified accordingly to the angle variation.

3.4.3. Elliptic wing with dihedral and sweep - Zefiro horizontal tail

In the following subsection, the third test case is presented. The horizontal tail has been introduced considering the geometric parameters of the Zefiro kite. AWEs fly in complex trajectories, and the horizontal tail is paramount, both from a stability and a control point of view. Understanding how the code reacts to the addition of a secondary lifting surface is paramount. Correct behaviours open the way to many possible analyses that could be done during the system's conceptual design.
The wing is kept elliptic, and the speed analysed is the same as the previous subsection, with the number of sections on the wing equal to 21 and 11 sections on the horizontal tail. The wing is positioned four meters behind the wing, but no translation on the y and z-axis is present.
The configuration of the kite is visible in fig. 3.18 and fig. 3.19

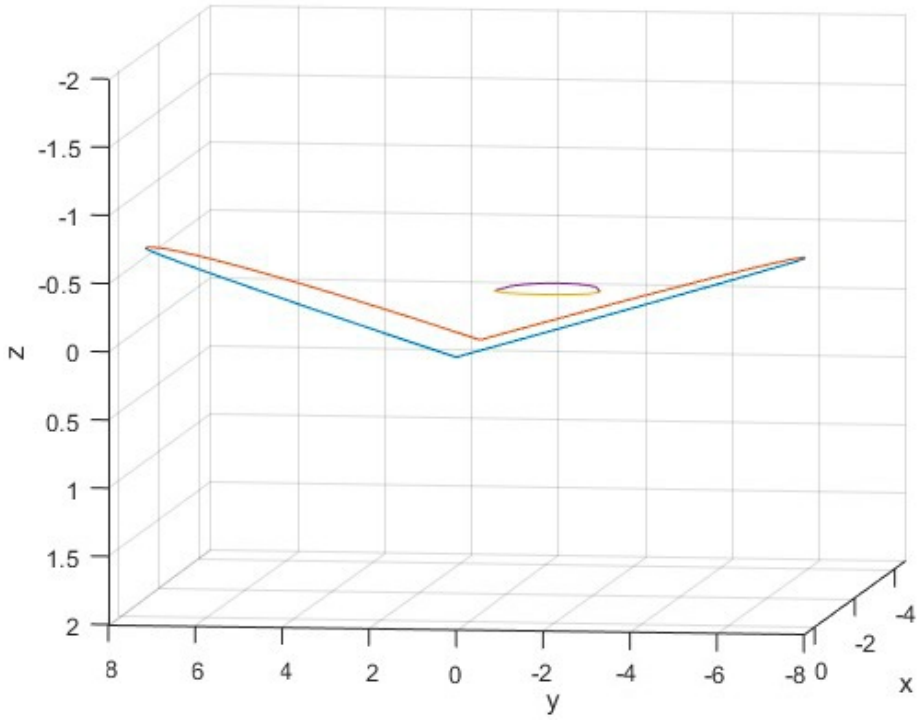


Figure 3.18: Configuration visualisation highlighting the dihedral angle .

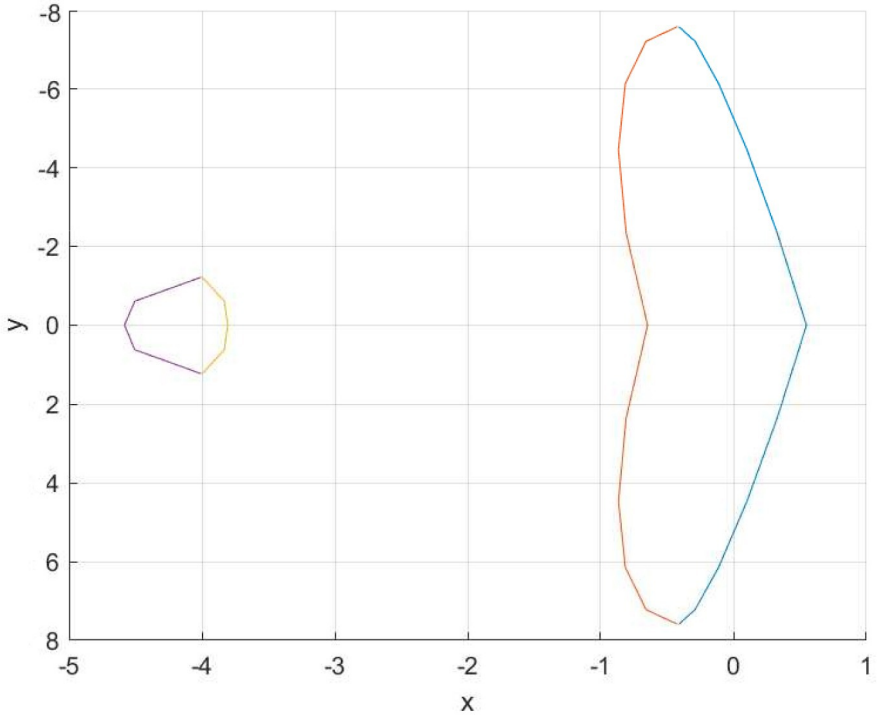


Figure 3.19: Configuration visualisation on the x-y plane.

The two analysis performed in order to verify the functionality of the code are the following:

1. Moment representation with the single wing and with the addition of the horizontal tail;
2. A comparison between the downwash angle computed thanks to [32] and the same angle estimated exploiting the induced velocities obtained from the VSM code.

The first validation test is selected to verify the effective increase in the moment due to the introduction of the traditional tail, which should help stabilise the kite. Two cases are analysed:

- An ideal case, with null moment coefficient (C_m) values for the airfoil interpolation, showed in fig. 3.20;
- A more realistic case, with a NACA 4412 on the wing and a NACA 0012 on the horizontal tail, visible in fig. 3.21.

In fig. 3.20, a comparison of the pitching moment obtained with the single wing and the pitching moment after the horizontal surface introduction is presented for an ideal case.

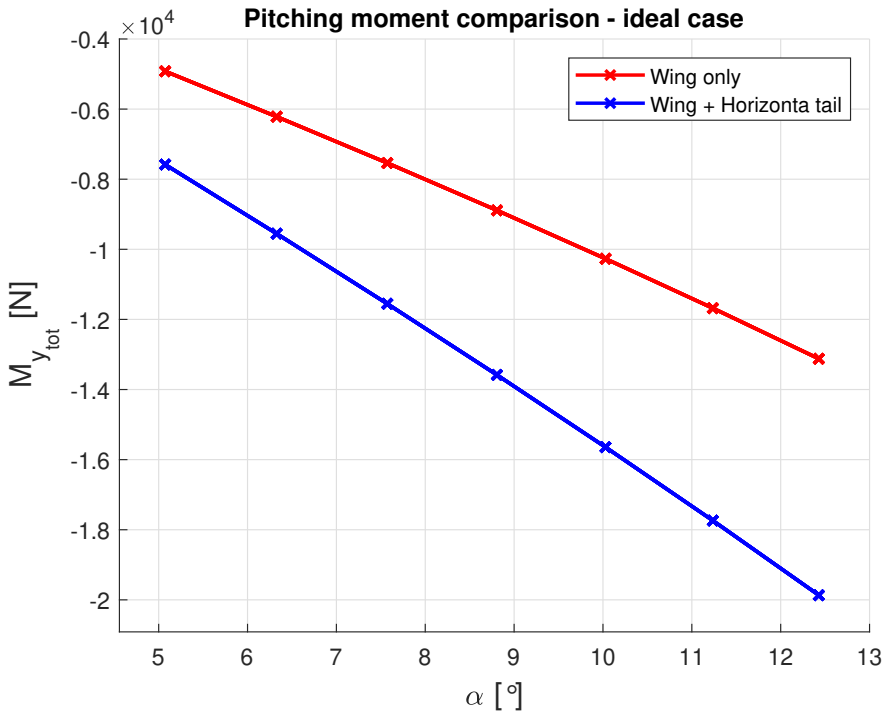


Figure 3.20: Single wing and wing plus horizontal tail pitching moment with respect to angle of attack variation (ideal case).

Whereas, in fig. 3.21, the realistic comparison with the airfoil introduction is presented.

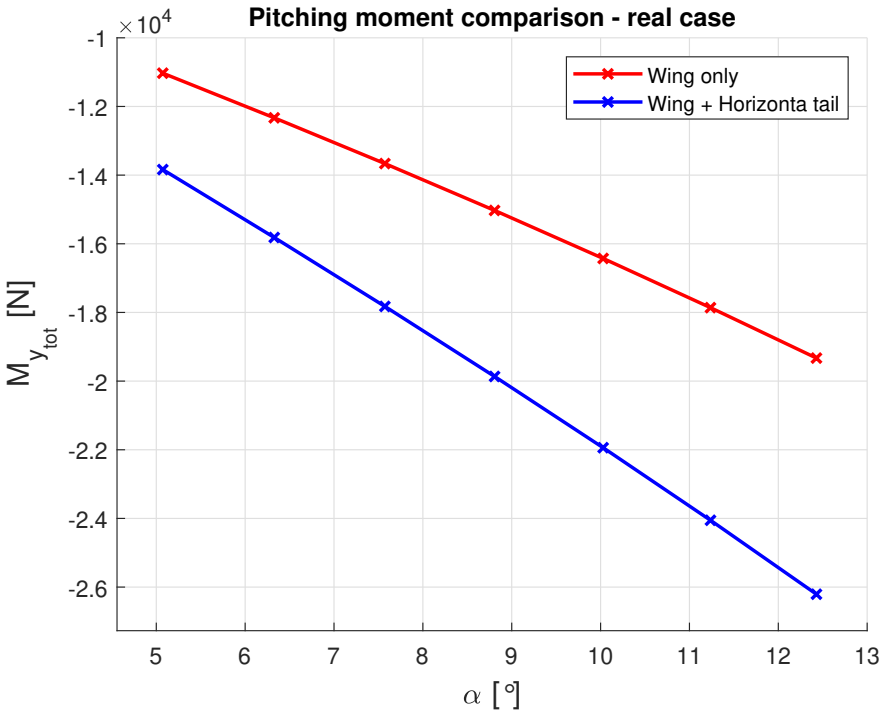


Figure 3.21: Single wing and wing plus horizontal tail pitching moment with respect to angle of attack variation (realistic case).

Two conclusions can be derived from the graphs; the implementation of the horizontal tail is correct since the global pitching moment of the aircraft is increasing. Moreover, introducing realistic value for the airfoil polar modifies the results, leading to a more negative pitching moment.

This was expected since the moment could be generally written, according to momentum transport rule, as:

$$\mathbf{M}_P = \mathbf{M}_Q + \mathbf{F} \times (Q - P) \tag{3.18}$$

In eq. (3.18), the value related to $\mathbf{M}_Q$ is null in all the ideal cases, since the value of the airfoil moment coefficient (C_m) is null by default. In fig. 3.21, the value is not null since a realistic arifoil (NACA 4412) is introduced on the wing.

The final validation for the horizontal tail consists of comparing an analytical estimate of the low-speed Downwash angle on the aft-tail ([32]) and the results estimated from the code, exploiting induction speed.
This step is necessary to verify that the program can catch the presence of the tail and

evaluate the induction between surfaces accordingly.
The analytic formulation presented in the paper is based on Prandtl's classical LLT and accounts for the effects of wing planform shape and tail position.
The downwash is a major effect to capture since it decreases the effective angle of attack of the tail.
This analytic formulation does not account for the dihedral angle and it is visible in eq. (3.19).

$$\varepsilon_d(\bar{x}, \bar{y}, \bar{z}) = -\frac{V_y(\bar{x}, \bar{y}, \bar{z})}{V_\infty} = \left(\frac{k_v k_p}{k_b}\right)\left(\frac{C_{lw}}{AR_w}\right) \tag{3.19}$$

where:

- k_v and k_b depends on the planform shape of the wing and were determined analytically to be equal to 1.0 and $\frac{\pi}{4}$ for an elliptic wing.
- k_p is a position factor that accounts for spatial variations in downwash. It can be computed from equation eq. (3.20):

$$k_P(\bar{x}, \bar{y}, 0) = \frac{2k_b^2}{\pi^2(\bar{y}^2 + k_b^2)}\left[1 + \frac{\bar{x}\,(\bar{x}^2 + 2\bar{y}^2 + k_b^2)}{(\bar{x}^2 + \bar{y}^2)\sqrt{\bar{x}^2 + \bar{y}^2 + k_b^2}}\right] \tag{3.20}$$

  with:

  - $\bar{x} = \frac{2x}{b}$;
  - $\bar{y} = \frac{2y}{b}$;
  - $\bar{z} = \frac{2z}{b}$;

  With x, y and z representing the distance between the tail and the wing, and b the span of the wing itself.

The results for an elliptic wing are visible in fig. 3.22.

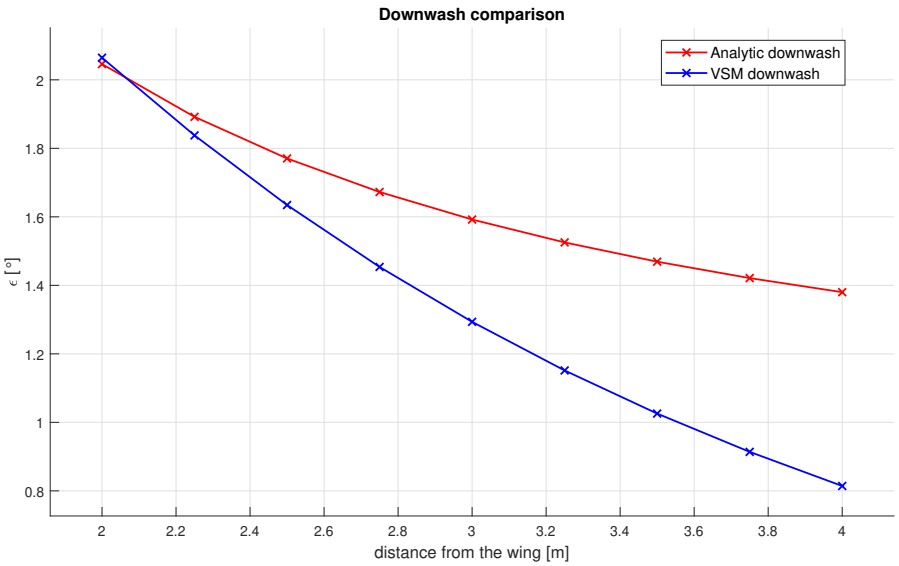


Figure 3.22: Comparison between analytic and VSM downwash angle for an elliptic wing with no dihedral angle nor sweep angle.

The trend of the VSM Downwash angle is similar to the analytic one. It tends to diverge while increasing the distance from the wing, this could be trace back to the analytic model. The latter is based on a single horseshoe vortex configuration with the vortex extending from the tips of the wing to infinity, while in the VSM code, several five filaments horseshoe vortex are presented, leading to a more conservative solution.

Finally, the sweep angle is introduced by comparing the downwash angle estimated with the code with an analytic method similar to the one described before but with a correction to account for the sweep. k_v and k_b are affected by the sweep, which modifies both the strength and spacing of the wingtip vortices, together with a modification of the spanwise vorticity distribution along the wing.
According to [32], the series solution to Prandtl's classical lifting-line equation does not apply to a swept wing. Therefore, it is not possible to obtain a closed-form solution for k_v and k_b due to the absence of a solution for the vorticity distribution.
The following analysis is based on a closed-form approximation of the effect induced by moving the bound vortex closer to the tail. It is based on the vortex model suggested by McCormick in [26] with a simplifying assumption that the wing tip vortices trail downstream from the wing quarter-chord in a direction parallel to the x-axis. The downwash angle is approximated as visible in eq. (3.21)

$$\varepsilon_d(\bar{x}, \bar{y}, 0) = -\frac{V_y(\bar{x}, \bar{y}, 0)}{V_\infty} = \left(\frac{k_v\, k_p\, k_s}{k_b}\right)\left(\frac{C_{l_w}}{AR_w}\right) \tag{3.21}$$

where:

- k_s is the wing sweep factor, it depends on the planform shape of the wing and the position of the tail and it can be computed from eq. (3.22).

$$k_s = \left[1 + \frac{\bar{x} - \bar{s}}{\bar{t}} + \frac{\bar{x}(\bar{r} + \bar{t})(\bar{t}_0^2 - \bar{x}^2)}{\bar{r}\bar{t}(\bar{r}\bar{t} + \bar{r}^2 - \bar{x}\bar{s})}\right] / \left[1 + \frac{\bar{x}(\bar{r}^2 + \bar{t}_0^2 - \bar{x}^2)}{\bar{r}^2\bar{t}_0}\right] \tag{3.22}$$

With:

$$\bar{r} \equiv \sqrt{\bar{x}^2 + \bar{y}^2} \tag{3.23}$$

$$\bar{s} \equiv k_b \tan\Lambda \tag{3.24}$$

$$\bar{t} \equiv \sqrt{(\bar{x} - \bar{s})^2 + \bar{y}^2 + k_b^2} \tag{3.25}$$

$$\bar{t}_0 \equiv \sqrt{\bar{x}^2 + \bar{y}^2 + k_b^2} \tag{3.26}$$

According to [32], the results for y = 0 agree with the results presented by McCormick for the special case of an elliptic wing. Therefore, $k_b = \frac{\pi}{4}$ also for this analysis.
The results for an elliptic wing with $\Lambda = 5°$ are visible in fig. 3.23

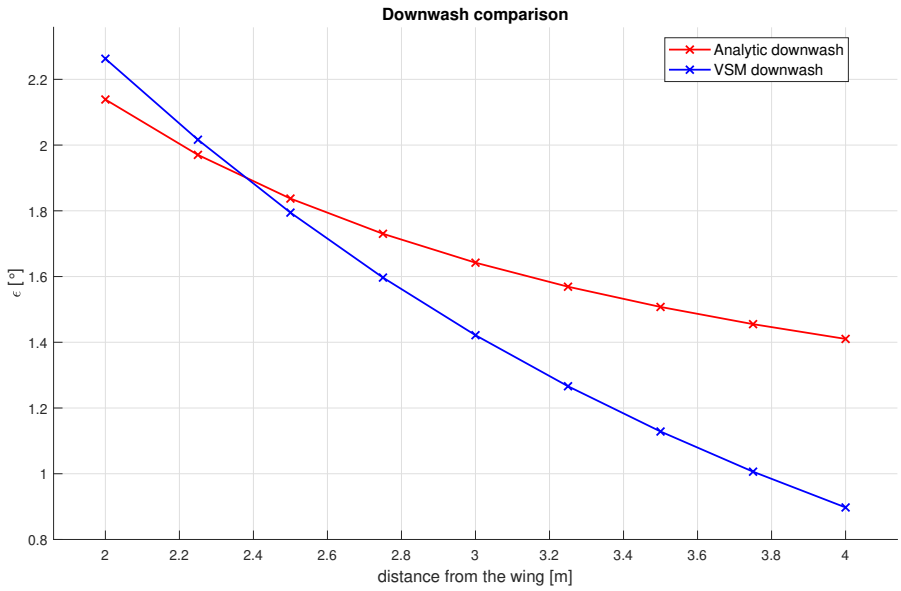


Figure 3.23: Comparison between analytic solution and VSM downwash for a wing with sweep angle.

It is possible to see how the behaviour is similar to the one without the sweep, with an initial downwash angle that is slightly higher for both the analytical and the VSM results.

The code can account for and predict the downwash angle (ε) while increasing the distance between the horizontal tail and the main wing. Notwithstanding, future iterations and modifications of the code could help improve the tail interaction and introduce a more complex way of studying the effect of sweep or dihedral angle on the downwash itself.

3.5. Tornado VLM - Reasons for the selection - Cautions

The Tornado VLM is considered a role model given its similarity with the VSM and high versatility. If the results obtained with the VSM are similar to a very used program like Tornado VLM, then the implementation possibility of the VSM is verified. Tornado is based on the vortex lattice theory, stemming from the potential flow theory. Moreover, the wake coming off the trailing edge of every lifting surface is flexible and changes shape according to the prescribed flight condition.
According to [27], Tornado performs a central difference calculation using the pre-selected state and disturbing it by a small amount (usually 0.5 degrees) to calculate the first-order derivatives.
Tornado is mainly used for conceptual design and supports multi-wing designs, with swept, cambered, twisted and tapered wings with or without dihedral angle. Any number of wings can be utilised, as well as any number of control surfaces.
These were the main reasons behind the attempts with the VLM.
Moreover, AWEs are still being researched; many different aircraft types can be exploited in tests and simulations. This method can be generally used for every aircraft without relying on a Look-up table.
Furthermore, it is open source and can be accessed and modified to improve algorithm performances.
Several **cautions** are presented in [27], especially:

1. As a primary assumption, the vortex lattice theory is used in a small angle of attack regimes;
2. Attention must be put when large rotational speeds are reached.
3. Compressibility effects are neglected, as are thickness effects of the lifting surfaces.

Potential flow methods are the best candidates for degree of accuracy and speed. They can provide a level of accuracy similar to CFD simulations while significantly reducing the computation time. Therefore, the selected method recalls this definition, including the previously described VSM and the VLM.
The entire Tornado program is complicated and a lot of functions are useless for this thesis. The simplified main script is presented in appendix A. The most important among the necessary sub functions are presented immediately after (appendix A).

3.6. VLM integration with MegAWEs simulation framework

The VLM method has already been extensively validated, therefore a direct integration with the simulator aerodynamic submodule (visible in appendix A in fig. A.1 and fig. A.2) is tested to verify the implications of direct integration of a potential method in the simulation environment while the VSM is being validated.
Once the method is introduced and adapted to the same geometries exploited using the LUT simulator (Delft kite described in section 3.1), an initial test is performed according to some simple control surface deflection inputs, understanding the implementation possibilities of this method.
The **sample time** during the simulation was kept equal to 0.5 s, rather than 0.005 s; the latter is used for the simulations with the Look-Up table.
Alongside the different computational time, other inputs are required; in fact, VLM works in a different reference frame and requires:

- The kite speed in the body reference frame;
- The wind speed in the body reference frame;
- The angular rates of the kite in body reference frames;

The reference frame is the typical body reference frame used in flight mechanics. It is defined with $\hat{x}$ along the root chord line, pointing forward towards the tip of the kite. $\hat{y}$ in the spanwise direction, pointing right in the direction of the semi-wing and $\hat{z}$ pointing down. The origin coordinate is situated in the attachment point of the tether.
The same reference system is exploited for the implementation of the VSM.

The next step consisted of connecting the VLM with the existing control module, verifying the integration with both control loops.

Control verification The control generally introduced in [20] and specifically in [34] is based on an inner and outer loop. During the first tests, several problems occurred with the VLM integration.
Specifically, some aerodynamic force and moment peaks were dimensionally wrong, and the simulation could not converge.

A feasible solution introduced to identify this problem was to put in parallel VLM and LUT blocks, closing the loop for the kite dynamics on the LUT. (visible in fig. A.3).
Exploiting the forces coming from the LUT, the VLM was working, showing an extreme

behaviour in several time steps during the simulation; with peaks of $1e+23$ for aerodynamic forces and moments, contrarily to results obtained from the LUT, correctly showing the reel-in phase and with dimensionally corrected values for the three components of the aerodynamic forces.

This led to the verification of several aspects that could be critical for the VLM (an the VSM too):

- The imported geometry and the panelisation of the aerodynamic surfaces;
- An incompatibility with the controller module;
- A numerical problem with the VLM itself.

In order to perform a correct comparison between LUT and VLM, the same geometry is kept, exploiting the kite presented in [20].

The problem has been found inside the **fastdw** function in which the AIC matrix is generated. The resulting matrix is ill-conditioned, and the obtained gamma has absurd values that are way higher than the previous time steps, causing the force peaks.
An immediate correction is obtained by introducing a fictitious value of the circulation every time the matrix results ill-conditioned, leading to the aerodynamic forces resulting in the correct order ($-1e+4$ [N]).
However, as visible from fig. 3.24, the magnitude is slightly wrong, with the VLM being half of the values obtained with the LUT, and with the x-component that is negative for the VLM and positive for the LUT.

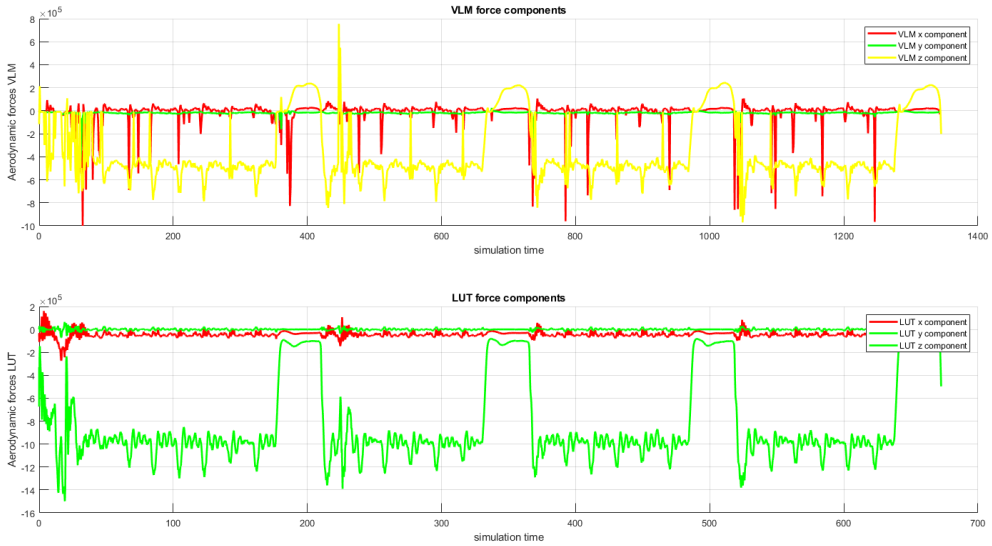


Figure 3.24: Aerodynamic forces comparison: VLM vs LUT

Due to several still present disturbances, manual tuning of the VLM is conducted to find a possible global solution for a correct implementation of the potential method rather than a soft option.

3.6.1. Tuning of the VLM

The following analysis consists of extracting the VLM from the simulator and creating a new simplified model (in Simulink) in which the deflections are constant and null to exclude the 'control' part of the block. All the inputs are manually modified.
The final aim is to manually tune the VLM, introducing conditions near the model's limitations and trying to recreate the excessive peak in the aerodynamic forces and moments. Hence, it is fundamental to understand whether a numerical problem is included in the functions or a flight condition near the potential method limitations is present.
Recreating a situation during the trajectory where the limitation of the VLM is violated could help in this sense.

The scheme is reported in fig. A.4, where it is possible to see all the possible combinations that could be imposed:

- Constant input;
- Sine input;
- Ramp input;

- Square and sawtooth input;

A series of different cases were introduced, allowing for specific analysis.

Constant speed for the kite - null angular rates - Sine variation of the wind
Many attempts are made, trying to catch an excessive variation of the forces. A simplified yet valuable example is reported herein;
It is presented as a case in which **Vx = 25 m/s**, **Vy = 10 m/s** and all the other input parameters are null, except for the x-component of the wind, which is a sine signal in the range [-25:25] m/s in amplitude with a frequency of 0.8 rad/s.
A peak in the aerodynamic forces and moments appears in a specific point. The explanation should be searched inside **state.AS** structure in the MATLAB algorithm of the simplified version of the VLM, presented in algorithm A.1.

The x-component of the total airspeed will be more or less null near the force peak; this is compatible with the assumption made for the VLM method, which is working correctly for small angle of attack. With a null airspeed on one geometric panel, the angle of attack could be high and not small, leading to a numerical failure of the VLM while performing the inversion of the matrices.

This analysis showed that implementing the VSM directly in the Delft simulator would also have led to a massive and unwarranted amount of work in modifying the control system. Therefore, it was decided to run the aircraft trim in MATLAB, testing the robustness of the VSM directly in the same environment. However, the trim chapter's cable model is extracted from MegAWEs since it is complete, functional and validated.
In section 3.7, a comparison between the VLM and the VSM is presented, showing how the latter yields almost identical results to the VLM for a high aspect ratio wing, contrarily to what is known from the inviscid case, where the VLM gives better results for low aspect ratio wings.

3.7. Comparison VSM - Tornado VLM and final results

This section is related to the actual comparison between the two potential aerodynamic models. The VLM is translated into a MATLAB script, simplifying, even more, the Simulink version presented before and introducing the same inputs as the VSM, in particular:

- Kite speed in body components (V_x, V_y, V_z);

- Angular rates (p, q, r);
- Wind speed components (V_{wx}, V_{wy}, V_{wz});

The test case has the following characteristics:

- No control surfaces;
- The vertical tail is not considered for the VSM;
- $V_x = 45\,m/s$ for the kite with a wind z-component ranging from -4 m/s up to -10 m/s with a -1 m/s step.

The results with Zefiro kite geometry (AR = 16) are visible in fig. 3.25.

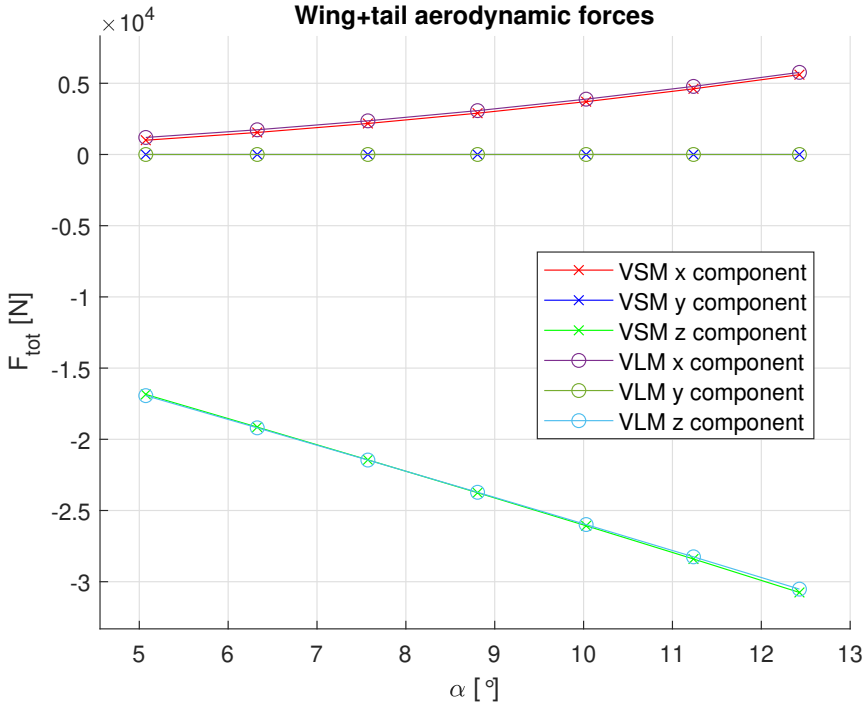


Figure 3.25: Wing and horizontal tail aerodynamic forces with respect to angle of attack variation for the Zefiro kite: VLM vs VSM.

The results are satisfying, especially on the z-component in which a very good match is visible.

Moreover, in fig. 3.26 the same analysis is performed for Delft kite.

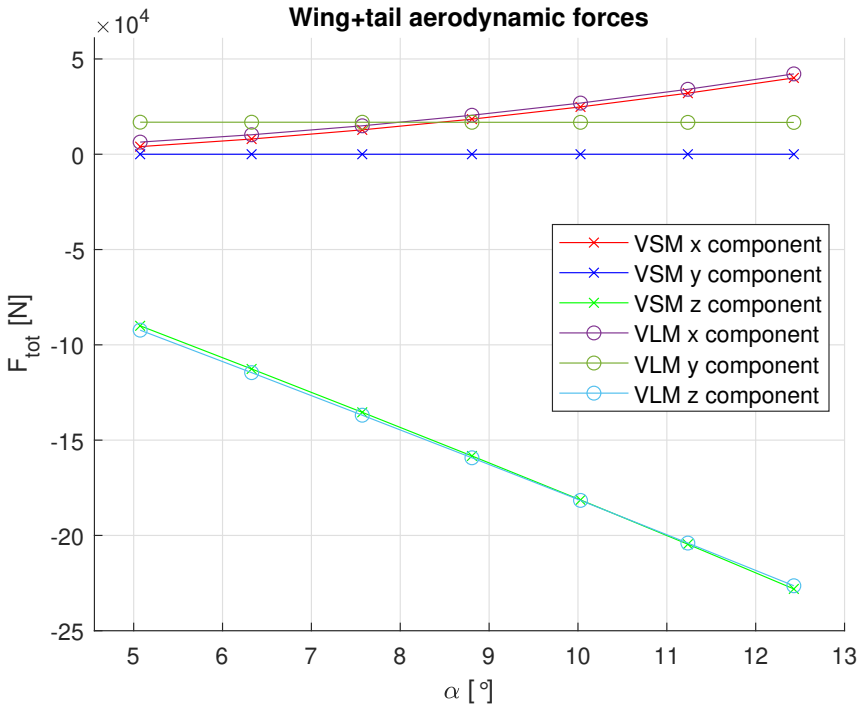


Figure 3.26: Wing and horizontal tail aerodynamic forces with respect to angle of attack variation for the Delft kite: VLM vs VSM.

The results are also coherent when changing the kite's geometry completely, especially on the z-component. The difference in the y-component, which should be equal to the previous case, is due to a slightly different geometry utilised while uploading the parametrisation input file for the simplified version of the VLM.

The results obtained with the VSM are very similar compared with the Tornado VLM. The only inconsistency is visible, in both figures, in the x-component of the aerodynamic forces, which could be investigated further on. However, for the validation, the viscous part was not paramount.

It is essential to highlight that the computational time of the VSM depends on the discretisation of the various lifting surfaces. The results of the VSM code, visible in fig. 3.25 and in fig. 3.26, are obtained with a computational time of 0.27 s for each iterative step, with this slightly reducing with the modification of the cross-inductions (0.26 s). If, at every iteration, the circulation is initialised with the final value of the previous step, the process can be sped up considerably (0.08 s).
The VLM has a lower computational time than the VSM, with a 0.08/0.1 s for each iteration step. A time that could be reached by the VSM if the circulation is not re-initialised to 0 but to the previous step value.

The results for the previous analysis are summed up in table 3.6. The computational time is considered for every step since the wind z-component is the usual vector ranging from -4 m/s up to -10 m/s. While increasing the value, the angle of attack is increased accordingly, with an increment of the computational time.
As a final reminder, the computation time steps presented in the table for the VSM are comprehensive of the circulation taken from the previous step and not re-initialising it to zero.

Table 3.6: VSM and Tornado VLM computational time comparison.

	Time per step (s)	
Model configuration	$\alpha = 5°$	$\alpha = 12.5°$
Tornado VLM (Simplif.)	0.08	0.24
VSM (Only wing)	0.12	0.13
VSM (Wing + H. Tail + all cross inductions)	0.27	0.27
VSM (Wing + H. Tail + only wing induction)	0.25	0.26

3.8. Vertical tail introduction

The final step before proceeding is the introduction of the vertical tail for the Vortex Step Method. The code is created so that a simple modification of the settings would translate into introducing a third lifting surface with its geometry and position.
Several considerations are made for the vertical tail geometry to simplify the computation, especially the shape of the vertical tail, which is equal to the Zefiro tail concerning the surface and the span values but with an elliptic chord distribution.
Therefore, the implementation is more straightforward, even though the dimension is exaggerated for a typical vertical tail. This modification entails the possibility of relying on previously created functions, with a specific change for the vertical tail normal and tangential versors.
The vertical tail is visible in fig. 3.27

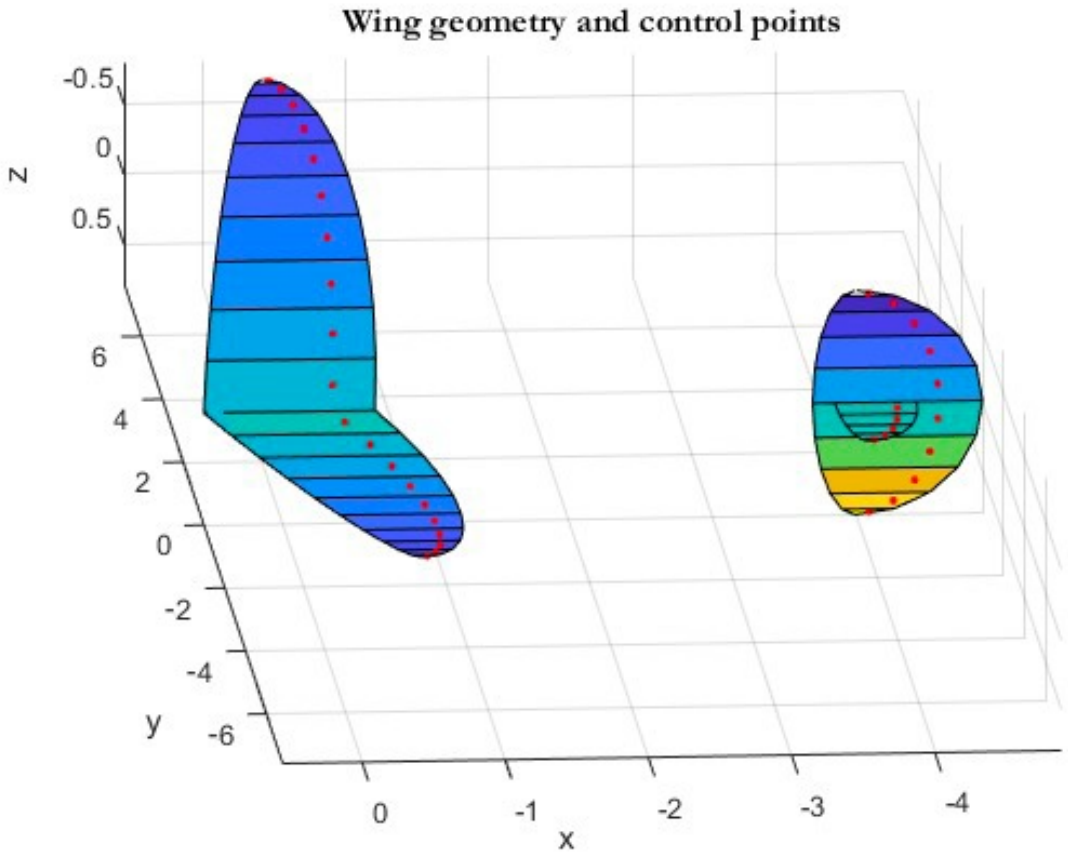


Figure 3.27: Wing and complete tail geometry together with the control points.

The vertical tail is also paramount since the next step consists of the trim of the kite. With a vertical tail and rudder, the trim is possible, and feasible results can be obtained.

As a final remark, the computational time for the VSM is re-computed once the vertical tail is introduced (table 3.7).

Table 3.7: VSM and Tornado VLM computational time comparison.

	Time per step (s)	
Model configuration	$\alpha = 5°$	$\alpha = 12.5°$
VSM (Wing + Full Tail + all cross inductions)	1.06	0.46
VSM (Wing + Full Tail + only H.tail - V.Tail cross induction)	0.57	0.46
VSM (Wing + Full Tail + only wind induction)	0.6	0.46

Once the vertical tail is introduced, each step's computational times are higher than the Tornado VSM. Further iterations are needed to better integrate the vertical tail inside

the aerodynamic model, also from a computational point of view.

3.9. Final considerations

During this chapter, two aerodynamic modules are introduced and analyzed.
The implementation and validation of the VSM are the main topics of this chapter, together with the comparison with the VLM.
The validation process followed a build-up approach, starting from a single elliptic wing and the correspondent analytical solution until the addition of the horizontal surface. All the parameters tested, including dihedral and sweep angle, are important while analyzing AWEs. The final comparison with the VLM demonstrates the potential of this method, which, although it has a higher computational time considering the full tail, still returns results very similar to a method widely used in conceptual aircraft design.
The VSM is reliable and computationally efficient. It includes the polar curves from the airfoils, and this aspect could be modified to extract information from a CFD database if needed. Moreover, different airfoils for different lifting surfaces could be analyzed.
Even though it is a **vortex filaments method**, it can account for the presence of flaps and control surface deflections due to non-linear contribution. Therefore, it helps estimate the effects of control surface deflections and capture local aerodynamic phenomena.
In a **Vortex panel method** like the VLM, the main drawback is that thickness and viscosity are neglected. Several approaches could be adopted to overcome this problem. However, this increases the computational time. Hence, 3D panel methods, considering the thickness, could be introduced, but they are not so convenient for modelling kite aerodynamics.
Finally, Comparing the VSM with a Look-Up Table, the advantages are clear: the VSM adapts to general cases and can be used for complex geometries analysis.

4 | Trim of the kite and dynamic simulation

A simulation could be performed now that the aerodynamic model is implemented and validated. In order to do so, the kite's trim must be investigated to find a reasonable initial guess for a possible simulation. Thanks to the attempt to introduce the VLM, several precautions were introduced together with some simplifications:

1. The control must be done from scratch or removed since introducing a potential method into an already robust and adaptive control system is impossible.
2. The aerodynamic module will not be tested in the Simulink framework since several cell and structures could create problem with code generation and the simulation process. Future modification of the program could be implemented in such a way the interaction with Simulink is smooth.

Given the completeness of the tethered model used in Delft MegAWEs, the same is exploited for the trim analysis.

Tether model A quasi-static lumped-mass tether model is implemented based on [48]. The main assumptions are:

1. Elastic vibrations are neglected;
2. The biggest effect on cable tension is related to mass and drag.

The main advantage with this model is the significant reduction in computational expanse, needed for real-time simulations. The quasi-static approach computes the steady state shape and corresponding tension forces throughout the tether by means of a shooting process ([20]) from the ground station towards the kite. A Trust-Region Dogleg Method alternates the tether force and direction at the ground station until the tether end is coincident (a magnitude of less than 1×10^{-6} for the distance between the two) with the kite position.
The state vector is $[\theta_n, \phi_n, \mathrm{T}_n]$ with T_n being the tension force in the wind reference

frame at the ground station. As stated in [20], exploiting spherical coordinates limits the solution to a positive definite force magnitude, making the Dogleg method much more stable and the convergence much faster.
The implementation or modification of the tether model is not an objective of the current thesis. Nevertheless, future iteration and modification could help improving the simulation introducing more stable and faster algorithms.
The tether characteristics are presented in table 4.1

Table 4.1: Detailed tether characteristics.

Parameter	Value	Unit
Diameter (d)	0.0297	m
Linear density (ρ_t)	0.6729	kg m^{-}1
Normal drag coefficient (C_N)	1.2	-
Axial elastic modulus (E)	116×10^9	GPa
Number of masses (N_P)	15	-

Wind field and environmental aspects A constant wind is used during the first integration of the aerodynamic module. Knowing how the wind speed varies with altitude is essential for accurate power predictions, but this is not paramount for the initial simulations. If the simulation converges and the VSM performs correctly, a modification of the wind could be introduced, allowing for the study of the kite as a sensor.

Control surface deflection introduction In order to proceed with a simplified yet valuable simulation, control surfaces must be introduced. Since they have not been physically designed yet, the idea is to account for the effect of control surface deflection through a lift coefficient offset (ΔC_L) introduced while interpolating the airfoil polars.
This is a simple control directly presented inside the aerodynamic module. The deflection for the aileron is accounted for by introducing a ΔC_L along the entire semi-span of the wing as if it is a full-wing aileron. The elevator and the rudder are considered along the entire tailplane surfaces.
It is a simplified yet effective solution needed for the next steps of the simulation, including trim and numerical integration of non-linear equations.

4.1. Generalized Trim of the kite

The next step consists of the generalized trim of the kite. Several assumptions and observations must be made. Gravitational and aerodynamic forces related to the mean elevation angle should act periodically on the kite due to the circular flight path. However, they are discarded to reach the trim condition in this thesis. Therefore, considering a fictitious steady state (called trim), centrifugal forces are balanced by the radial component of the tether force [44]. Moreover, power generation is not model in this thesis.

4.1.1. Coordinate systems

Four coordinates systems are used during the trim analysis:

- The body reference system - $\mathcal{RF}_B$, already introduced within the aerodynamic module implementation. It does not depend on the operational regime and might be centered in principle in any point of the aircraft. The x axis is pointing forward, the y axis is pointing right along the semi-wing and the z-axis is consequently obtained.
- The ground reference system - $\mathcal{RF}_G$, which is centered at the ground station and it is inertial. Z_G points upwind and X_G toward the ground. The latter could be considered as the wind reference system.
- A rotating reference frame - $\mathcal{RF}_R$, which is defined such that it moves on a circumference of constant radius R_0. The x-axis (X_R) points along the tangential direction and the y-axis (Y_R) outward, defining the rotor plane.
- The stability reference system - $\mathcal{RF}_S$, which is centered at the tether attachment point and is moving and rotating like the kite. This is the reference system in which the equation of motions are written, creating therefore a compact formulation.

For further information regarding the reference systems, refers to [44].

Consequently, the rotation matrices describing a coordinate transformation between the various reference frame are defined herein, starting by the rotation matrix from the $\mathcal{RF}_G$ to $\mathcal{RF}_R$.

$$\mathbf{R}^G_{GR}(\Psi,\beta) = \mathbf{R}^G_{G'R}(\Psi)\,\mathbf{R}^G_{GG'}(\beta) = \mathbf{R}^G_{y,GG'}(\beta)\,\mathbf{R}^{G'}_{z,G'R}(\Psi) \tag{4.1}$$

The rotation visible in eq. (4.1) is defined by two sequential planar rotations, associated with the mean elevation angle (β) and the angular position (Ψ) of the rotating reference system. In the case studied for the trim, a simplified case is exploited, with a **null**

elevation angle.

In eq. (4.2) the general rotation matrix from $\mathcal{RF}_R$ to $\mathcal{RF}_S$ is presented. It is made of three sequential planar rotations fig. 4.1.

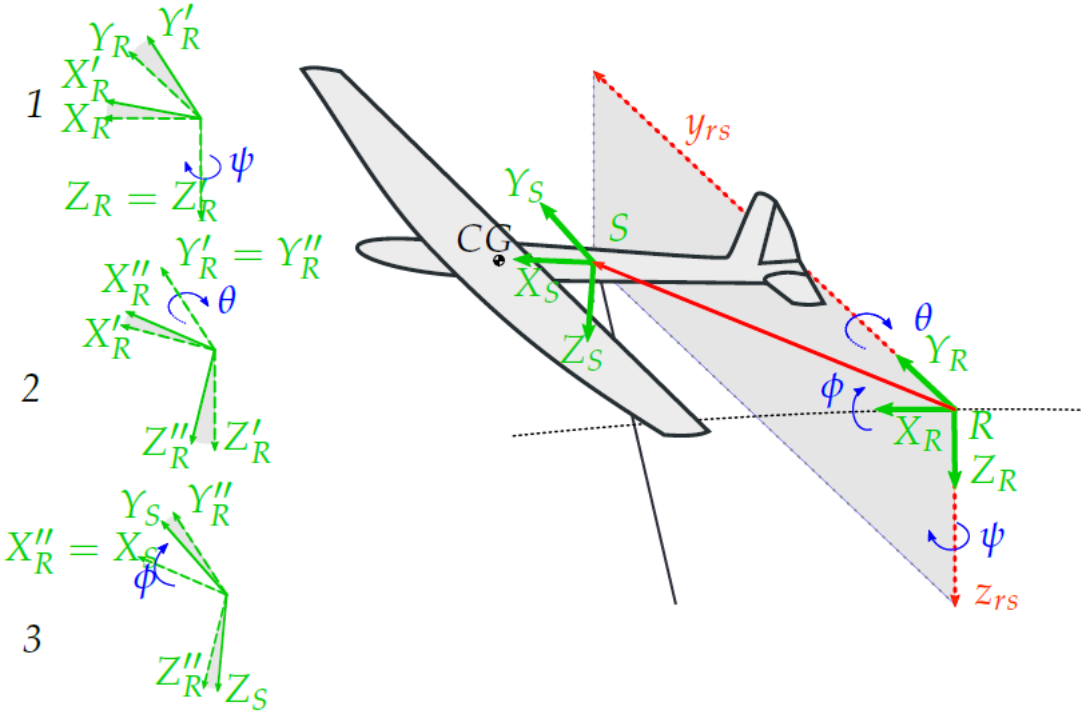


Figure 4.1: Rotating coordinate system and stability coordinate systems. [44]

$$\mathbf{R}^R_{RS}(\phi,\theta,\psi) = \mathbf{R}^R_{R''S}(\phi)\,\mathbf{R}^R_{R'R''}(\theta)\,\mathbf{R}^R_{RR'}(\psi) = \mathbf{R}^R_{z,RR'}(\psi)\,\mathbf{R}^{R'}_{y,R'R''}(\theta)\,\mathbf{R}^{R''}_{x,R''S}(\phi) \tag{4.2}$$

The visual representation of both $\mathcal{RF}_G$ and $\mathcal{RF}_R$ frames is showed in fig. 4.2

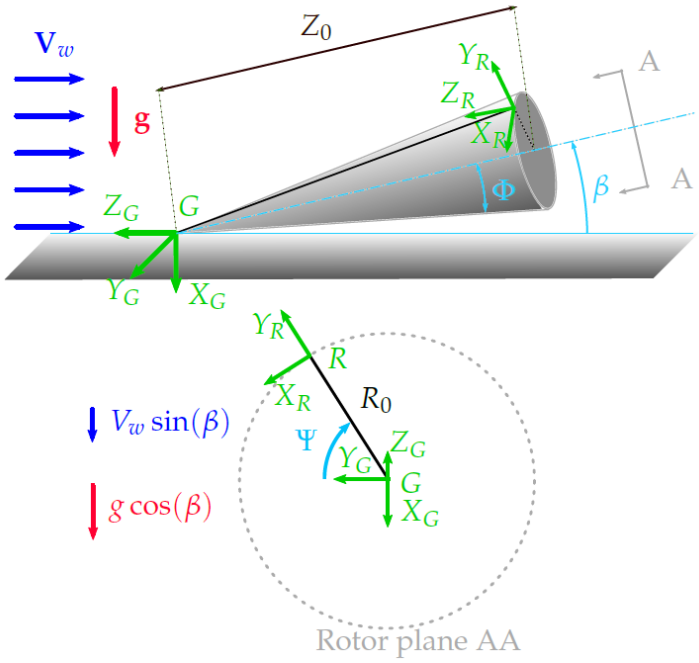


Figure 4.2: Ground coordinate system and rotating coordinate system. [44]

Also in this second case, several simplifications are introduced thanks to a specific selection of the initial state for the trim condition. In this thesis, $\mathcal{RF}_R$ and $\mathcal{RF}_G$ are aligned, the rotation matrix is much simpler and involves simply the pitch angle. Since the gravity is neglected, Ψ does not influence the dynamic of the kite, therefore, taking $\Psi = 0$ is like considering Ψ equal to every other value. A representation of the following configuration is visible in fig. 4.3

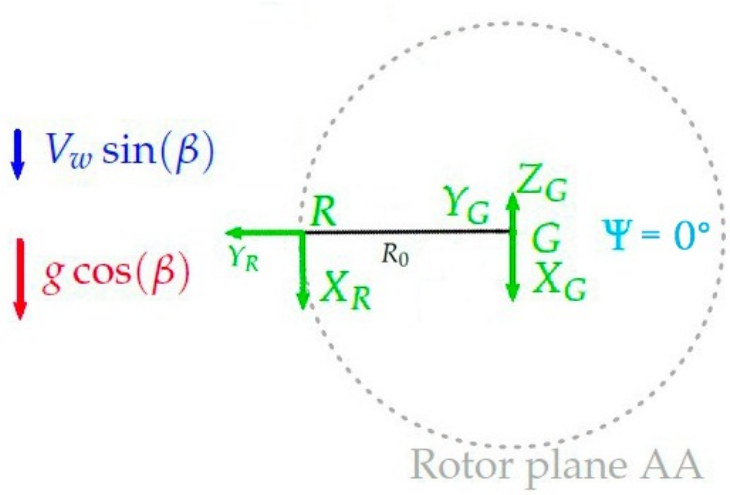


Figure 4.3: Trim condition

Moreover, in steady-state conditions, the Rotating reference frame and the stability reference frame coincide, translating into a particularly useful coordinate system for studying perturbations around a steady-state condition. For the analytic part and the matrix notation refers to Appendix A, in [44].

4.1.2. Equations of motion in the stability reference frame

A trim condition is defined as a flight trajectory for which the generalized velocity vector measured in the body reference frame and the input vector remain constant in time:

$$\omega^B = constant, \quad \delta = constant \tag{4.3}$$

Under several hypothesis (i.e. flat Earth) an aircraft is able to maintain this condition. Trimming the kite is useful to obtain an optimal initial condition for the integration of non linear equations.
Six equations are taken into consideration, related to six state variables that describes the motion of the kite in the simplified situation described before:

- U_0 is the kite airspeed in the stability reference frame;
- R_0 is the radius of gyration;
- ε_t is the tether deformation parameters;
- Three δ values, including the three lift coefficient offsets (ΔC_L) generated by the elevator, ailerons and rudder.

The pitch angle of the kite is prescribed; in this way, we are trying to trim the kite while maintaining a specific pitch angle. It translates into finding the six state variables that are able to trim the kite.
Once the non-linear system with six unknowns and six equations is written, the trim problem can be solved through an iterative process, starting with a suitable initial guess.
As a final remark, an appropriate aerodynamic model should be used for evaluating the respective term in the non-linear equations. The selected model is the previously implemented VSM.

Disclaimer: It is paramount to specify that a numerical problem appeared while analysing the trim, probably due to the introduction of the vertical tail and its interaction with the other surfaces or the control surfaces while interpolating the airfoil polars. Therefore, the LLT implemented inside the VSM code is exploited to find the trim solution, being aware

that once the numerical problem is resolved, the same trim analysis could be launched simply by modifying an input.

Position To formulate the equations of motion in $\mathcal{RF}_S$, the position of the ground station with respect to the S frame must be found. Since the condition in which we are trimming the kite is fairly simple, there is no difference in rotating the G reference frame to the R reference frame. Moreover, due to the steady-state condition, $\mathcal{RF}_S$ and $\mathcal{RF}_R$ coincides.

Angular velocity Due to the simplifications introduced before, the angular velocity $\omega = [\mathrm{p}, \mathrm{q}, \mathrm{r}]^{\mathrm{T}} = \left[0, 0, -\frac{\mathrm{U}_0}{\mathrm{R}_0}\right]$.

Relative wind speed The wind velocity is along the Z_G axis and a constant, uniform wind field is considered for this simulation. However, the VSM/LLT is entirely defined in body reference frame, requiring a rotation of the wind vector before the actual computation. Once the aerodynamic forces and moments are obtained, the correspondent vectors are rotated back into the stability reference frame.

Equations of motion Since the stability frame is centered on the tether connection, the AWEs center of mass can be elsewhere. Under the previous hypothesis, the equation of motion is the following:

$$\begin{aligned} &\begin{bmatrix} m\mathbf{1} & -m\mathbf{TCG}^{\times} \\ m\mathbf{TCG}^{\times} & \mathbf{I} \end{bmatrix} \begin{pmatrix} \dot{\mathbf{V}} \\ \dot{\boldsymbol{\omega}} \end{pmatrix} + \\ &\begin{bmatrix} \boldsymbol{\omega}^{\times} & \mathbf{0} \\ \mathbf{V}^{\times} & \boldsymbol{\omega}^{\times} \end{bmatrix} \begin{bmatrix} m\mathbf{1} & -m\mathbf{TCG}^{\times} \\ m\mathbf{TCG}^{\times} & \mathbf{I} \end{bmatrix} \begin{pmatrix} \mathbf{V} \\ \boldsymbol{\omega} \end{pmatrix} = \begin{pmatrix} \mathbf{F} \\ \mathbf{T}_t \end{pmatrix}, \end{aligned} \tag{4.4}$$

where:

- m is the kite mass;
- $\mathbf{1}$ is the identity matrix;
- $\mathbf{TCG}$ is the position of the center of mass in the S reference frame;
- $\mathbf{I}$ is the inertia tensor by components in S (obtained from the inertia known in the body reference frame);
- $\mathbf{F}$ and $\mathbf{T_t}$ are the external forces and moments (during the trim just aerodynamic and tether forces are present).

- the symbol $\times$ represents the skew-symmetric tensor of that vector.

Non-dimensional Equation of Motion A diagonal matrix is introduced to make the equations of motion non-dimensional, the matrix is obtained from [44] and it is visible in

$$\mathbf{S} = diag\left([1, 1, 1, b, c, b]\right) \tag{4.5}$$

where b is the reference wing span and c the reference chord. The unit force is defined as

$$\mathtt{F} = \frac{1}{2}\rho A u_0^2, \tag{4.6}$$

where ρ is the air density, A is the reference wing area and u_0 the AWEs linear velocity (evaluated at the tether anchor point). A non-dimensional form of 6-dimensional equation of motion is:

$$\frac{\mathbf{S}^{-1}}{\mathtt{F}}\mathbf{m}\dot{\mathbf{u}} + \frac{\mathbf{S}^{-1}}{\mathtt{F}}\mathbf{u}^{\times}\mathbf{m}\mathbf{u} = \mathbf{f}, \tag{4.7}$$

wherein:

$$\mathbf{m} = \begin{bmatrix} m\mathbf{1} & -m\mathbf{TCG}^{\times} \\ m\mathbf{TCG}^{\times} & \mathbf{I} \end{bmatrix}, \qquad \mathbf{u}^{\times} = \begin{bmatrix} \boldsymbol{\omega}^{\times} & \mathbf{0} \\ \mathbf{V}^{\times} & \boldsymbol{\omega}^{\times} \end{bmatrix}, \qquad \mathbf{u} = \begin{pmatrix} \mathbf{V} \\ \boldsymbol{\omega} \end{pmatrix}, \tag{4.8}$$

and

$$\mathbf{f} = \frac{\mathbf{S}^{-1}}{\mathtt{F}} \begin{pmatrix} \mathbf{F} \\ \mathbf{T}_t \end{pmatrix} = \left[\tilde{X},\ \tilde{Y},\ \tilde{Z},\ \tilde{L},\ \tilde{M},\ \tilde{N}\right]^{\mathrm{T}}. \tag{4.9}$$

The initial trim condition is obtained with a simplified tether model, in which the tether forces balance the centrifugal and inertial forces. The final aim is to introduce the tether model presented in [20], explained at the start of the chapter.

4.1.3. Trimmed solution

A few assumptions are made to ease the procedure of trimming the aircraft.
The main wing is elliptic and the induced velocities generated by the lifting surfaces are computed thanks to the LLT.
Note that the dihedral angle (Γ) is positive with the right wing pointing down, while the sweep angle (Δ) is positive with right wing pointing backwards.
Since the equations of motions have been written in a coordinate system centered in the

tether attachment, the tether is not generating any moment.

The initial guess is very simple, with a tethered AWEs in motion with an imposed velocity $V = [50, 0, 0]$ m/s, a prescribed pitch angle ($\theta = 0°$), null values for the ΔC_L of the control surfaces and a simple tether model:

- Tether initial length: $L_0 = 145$m;
- Tether diameter: $d_t = 0.0196$cm;
- Tether drag coefficient: $C_\perp = 0.8$;
- Young modulus: $E = 116e + 9$ Pa

The airspeed used as an input is defined in the stability reference frame, the inertia of the kite is defined in the body reference frame and the VSM/LLT is working in the body system too. In order to keep a consistency with reference frames, a rotation matrix between B and S is defined.
The latter is obtained using the prescribed angle of the kite. The angle will be called θ_{BS} for clearance and it is defined as the angle around the Y_B axis, and describes a rotation from $\mathcal{RF}_B$ to $\mathcal{RF}_S$.

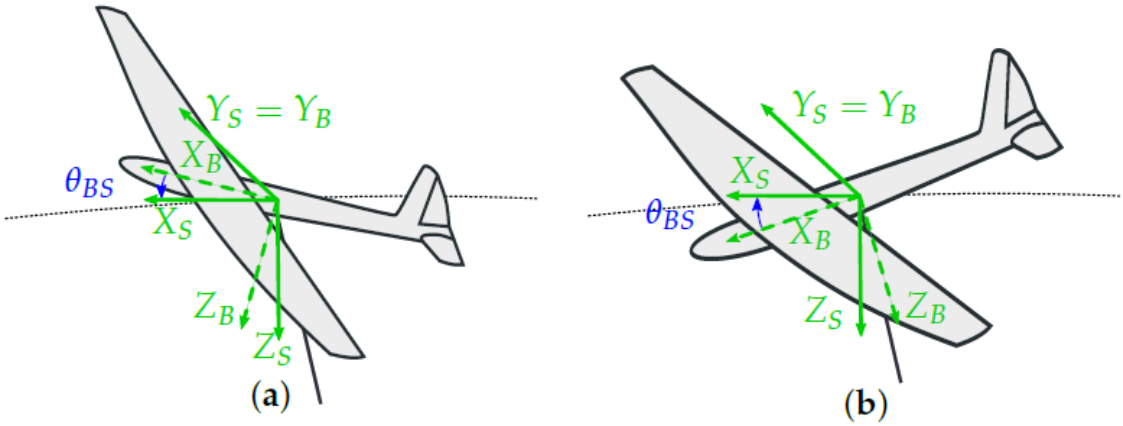


Figure 4.4: Body reference frame ($\mathcal{RF}_B$) and stability reference frame ($\mathcal{RF}_S$) with positive pitch angle θ_{BS} - source: [44]

Trim condition This first test case is of paramount importance to verify the effectiveness of the trim condition, the integration with the VSM and to verify the physics behind, understanding the feasibility of the trim solution itself. This condition is visible in table 4.2

Table 4.2: $\theta_{BS} = 0°$ and $U_{0_{guess}} = 50$ m/s trim condition.

	$\mathbf{U_0}$ (m/s)	$\mathbf{R_0}$ (m)	$\boldsymbol{\varepsilon_t}$ (mm/m)	$\mathbf{\Delta C_{L_e}}$	$\mathbf{\Delta C_{L_a}}$	$\mathbf{\Delta C_{L_r}}$
@Trim	40.9	143.4	0.509	-0.1038	0.0025	-0.0323

4.1.4. Physics of the trim condition and the initial guess

The pitch angle of the kite is prescribed as a null value, the initial value for the tether deformation parameter is equal to 0.0623 while the initial Radius of gyration is $R_0 = 146.4$ m, given the tether initial length ($L_t^0 = 145$ m) and $\Phi_t = 20.7°$.
Once the trim is reached, the value of the airspeed is reduced, this was expected since the pitch angle is fixed and the kite must reduce its speed to match it.
The radius (R_0) is increasing, the final trimmed trajectory is larger compared to the initial one.
The values of the control surfaces' lift coefficients stored in δ are feasible. Longitudinally speaking, the lift coefficient offset for the elevator (ΔC_{L_e}) is negative, recalling $\Delta C_{L_e} = C_{L_{\delta_e}} \delta_e$ a negative value is expected.
Concerning the vertical tail, that is meant to trim the aircraft over the circular path, a positive offset for the lift coefficient (ΔC_{L_r}) is obtained.

Regarding the actual forces and moments acting on the kite at trim, they are as expected:

Table 4.3: Forces acting on the kite at trim.

[kN]	F_{ax}	F_{ay}	F_{az}	F_{Tx}	F_{Ty}	F_{Tz}	F_{in_x}	F_{in_y}	F_{in_z}
@Trim	1.67	0	-16.7	-1.67	-6.16	16.71	0.0086	6.17	0

Table 4.4: Moments acting on the kite at trim.

[kN m]	M_{ax}	M_{ay}	M_{az}	M_{Tx}	M_{Ty}	M_{Tz}	M_{in_x}	M_{in_y}	M_{in_z}
@Trim	0.0083	-0.0175	-1.367	0	0	0	0	0.0074	1.234

As visible from table 4.3, the aerodynamic forces are compatible with the reference system: the lift is negative accordingly, the y-component of the forces acting on the tether is dragging the kite into the centre of the circular path, and the centrifugal force is positive, pushing the kite out of the trajectory.
Concerning the moments, visible in table 4.4, the values are feasible and the tether is not generating any moment due to the positioning of the $\mathcal{RF}_S$ origin on the tether attachment point.

4.1.5. Trim solution with pitch angle variation

Once the feasibility of the trim trajectory with a null pitch angle has been investigated, the next step is to test the robustness of the trim condition and of the six equations of motion used as constraints. Several test cases are analyzed, starting by the different trim conditions that can be reached varying the prescribed pitch angle, with a particular focus on the airspeed.

Table 4.5: Trim condition with variable pitch.

	$\mathbf{U_0}\,(m/s)$	$\mathbf{R_0}\,(m)$	ε_t (mm/m)	$\mathbf{\Delta C_{L_a}}$	$\mathbf{\Delta C_{L_e}}$	$\mathbf{\Delta C_{L_r}}$
$\theta_{BS} = -2°$	37.6	151.8	0.377	-0.0876	0.0092	-0.0168
$\theta_{BS} = -1°$	39.3	147.6	0.441	-0.0954	0.0060	-0.0263
$\theta_{BS} = 0°$	40.9	143.4	0.509	-0.1038	0.0025	-0.0323
$\theta_{BS} = 1°$	42.6	139.4	0.584	-0.1126	-0.0014	-0.0283
$\theta_{BS} = 2°$	44.1	135.6	0.663	-0.1220	-0.0056	0.0009
$\theta_{BS} = 3°$	45.5	131.9	0.748	-0.1314	-0.0101	-0.1301
$\theta_{BS} = 4°$	46.7	128.3	0.837	-0.1420	-0.0153	-0.1942

The results are summarized in table 4.5, herein a representation of the behaviour of the parameters with respect to the pitch angle variation is presented, followed by a brief explanation.

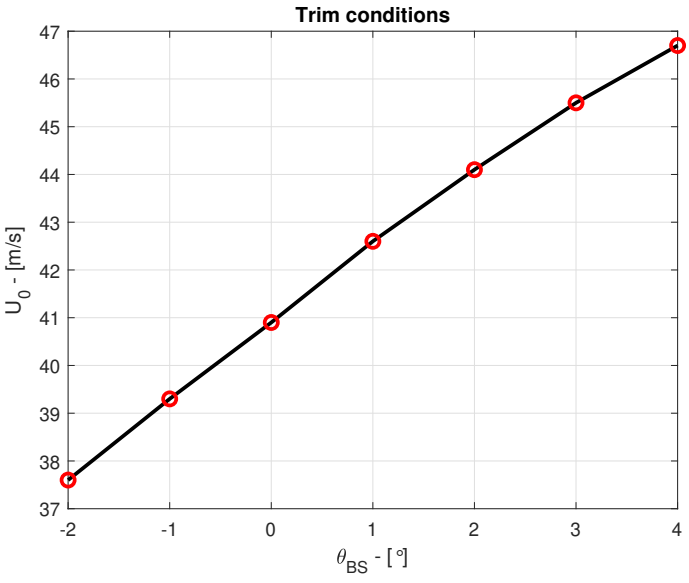


Figure 4.5: Airspeed behaviour with respect to pitch angle variation

With low and positive pitch angles the convergence is pretty fast and the results are coherent, with $\theta_{BS} = 2°$ some problems started to rise with the integration with the aerodynamic module, increasing the computational time at each step. Nevertheless, all the values are showing an expected behaviour, with the increase of the airspeed while increasing the pitch.

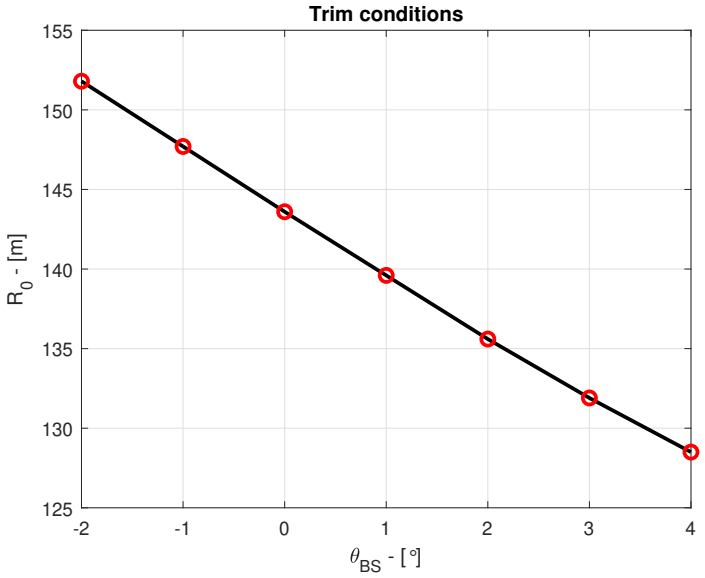


Figure 4.6: Trajectory radius behaviour with respect to pitch angle variation

Also when looking at the Radius variation, a decrease of the radius of the trajectory is obtained while increasing the pitch angle.

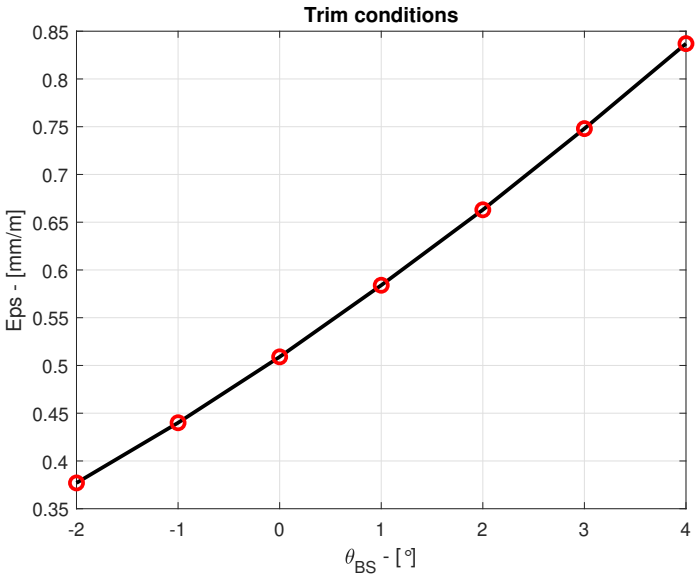


Figure 4.7: Tether deformation parameter behaviour with respect to pitch angle variation

The tether deformation parameter is globally increasing with the increment of the pitch angle. This is expected since the aerodynamic forces are increasing, and the tether force must be greater to balance the system. This influences the deformation due to the constant Young modulus and surface area.

Finally, the control surfaces are analysed in fig. 4.8. It is possible to see the delta's in the lift coefficient needed to trim the kite. The behaviour is presented as a function of the pitch angle; therefore, every trim condition is visible as a point in the graph.

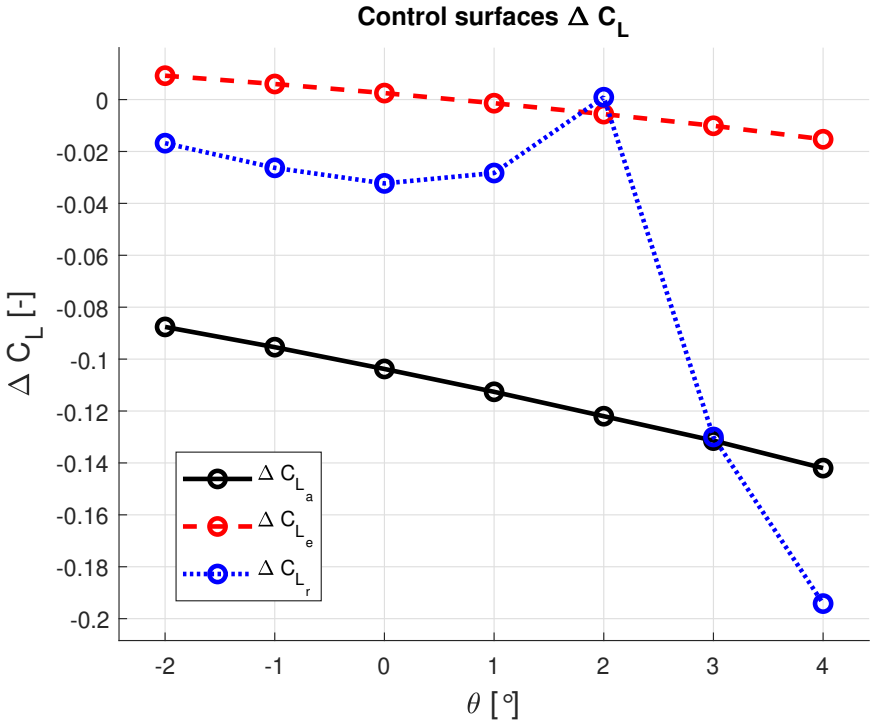


Figure 4.8: Lift coefficient increments for aileron (black solid line), elevator (red dashed line) and rudder (blue dotted line) with respect to pitch angle variation.

It is possible to see the low lift coefficient offset for the elevator used to trim the kite; this is due to the trim analysis being performed with the LLT, leading to a symmetrical problem. Regarding the rudder, the variation in the lift coefficient starts with suitable values but ends with an unexpected peak followed by a drastic reduction of the value. Even so, the trim condition was achieved without an increase in the computational time. The sudden reduction is related to the same numerical error affecting the vertical tail while analysing the trim with the VSM. Nevertheless, as already stated, the trim condition could be achieved by exploiting LLT since this error cancels out when considering the problem as symmetric.

The plot of the total lift coefficient values of the vertical and horizontal tail including

the lift coefficient offset obtained from the elevator and rudder is even more helpful. In the following pictures, for every pitch angle, the lift coefficient (C_L) distribution over the surface is reported.

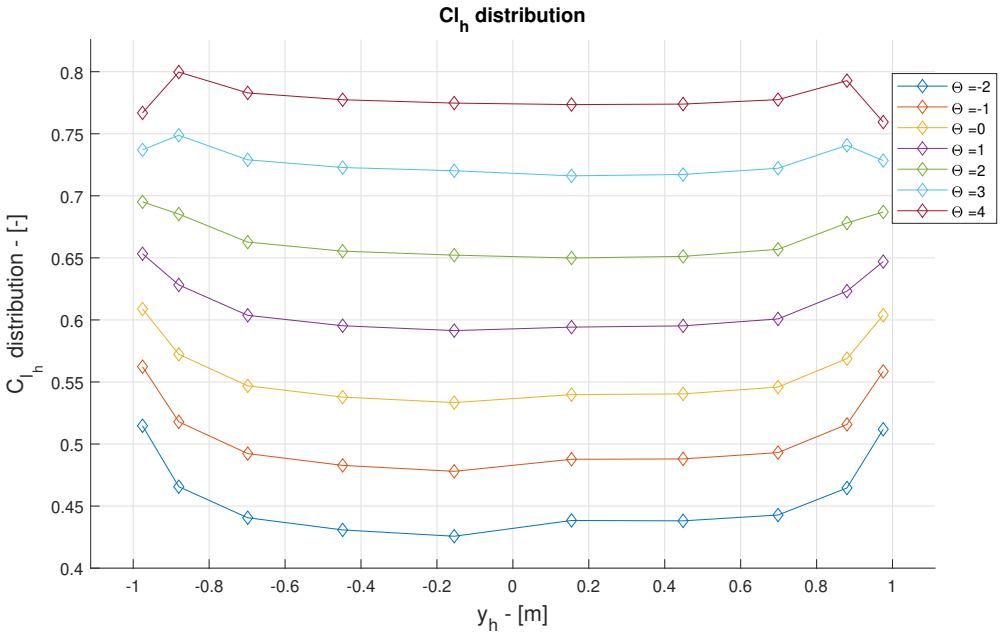


Figure 4.9: Horizontal surface total lift coefficient distribution including elevator contribution along the span of the horizontal tail

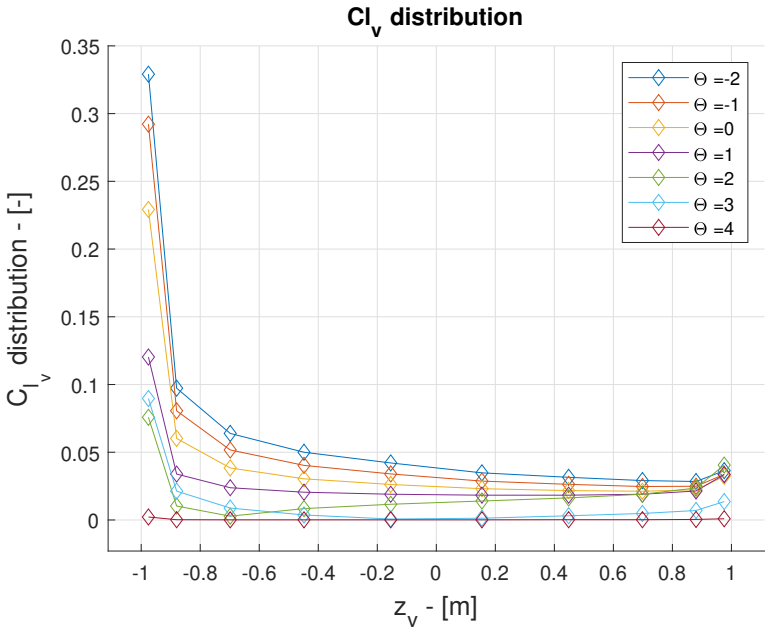


Figure 4.10: Vertical surface total lift coefficient distribution including elevator contribution along the span of the vertical tail

From both fig. 4.9 and fig. 4.10, it is possible to see that the distribution along both surfaces is coherent with all the pitch angles.
The most important thing to notice is that all the extreme values for the lift coefficient (C_L) are found in the correspondence of the first element of the vector identifying the span distribution.
This could be traced back to a possible numerical error inside the AIC matrix population according to the control surface discretisation or a problem in the numerical evaluation while passing from the horizontal surface to the vertical surface inside the VSM program. The error is mitigated while performing analysis with the LLT since the problem is symmetric. This again demonstrates why the trim analysis was performed with LLT.

4.1.6. Trim solution with kite mass variation

The second analysis taken into consideration is related to the variation of the kite mass, which influence several tether parameters and the inertia. In order to produce a straightforward analysis, the inertia moments are unvaried and the mass is incremented only of a ± 20 %. Furthermore, the pitch angle is prescribed to a null value.

Table 4.6: Trim condition with mass variation.

	$\mathbf{U_0}\,(m/s)$	$\mathbf{R_0}\,(m)$	$\varepsilon_t\,(mm/m)$	$\mathbf{\Delta C_{L_a}}$	$\mathbf{\Delta C_{L_e}}$	$\mathbf{\Delta C_{L_r}}$
$m = 424\,\text{kg}$	40.9	128.6	0.502	-0.1110	0.0022	0.0547
$m = 450.5\,\text{kg}$	40.9	132.4	0.504	-0.1090	0.0023	0.0329
$m = 477\,\text{kg}$	40.9	136.3	0.505	-0.1071	0.0023	0.0111
$m = 503.5\,\text{kg}$	40.9	139.9	0.507	-0.1054	0.0024	-0.0107
$m = 530\,\text{kg}$	40.9	143.4	0.509	-0.1038	0.0025	-0.0323
$m = 556.5\,\text{kg}$	40.9	146.8	0.511	-0.1023	0.0025	-0.0539
$m = 583\,\text{kg}$	40.9	150.1	0.513	-0.1009	0.0025	-0.0754
$m = 609.5\,\text{kg}$	40.9	153.3	0.515	-0.0995	0.0026	-0.0967
$m = 636\,\text{kg}$	40.9	156.5	0.516	-0.0983	0.0026	-0.1180

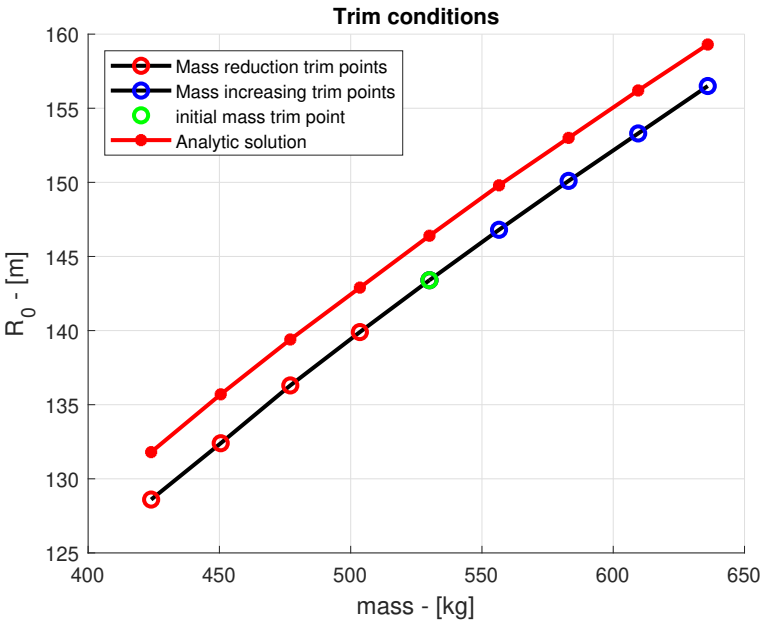


Figure 4.11: Radius variation with the mass.

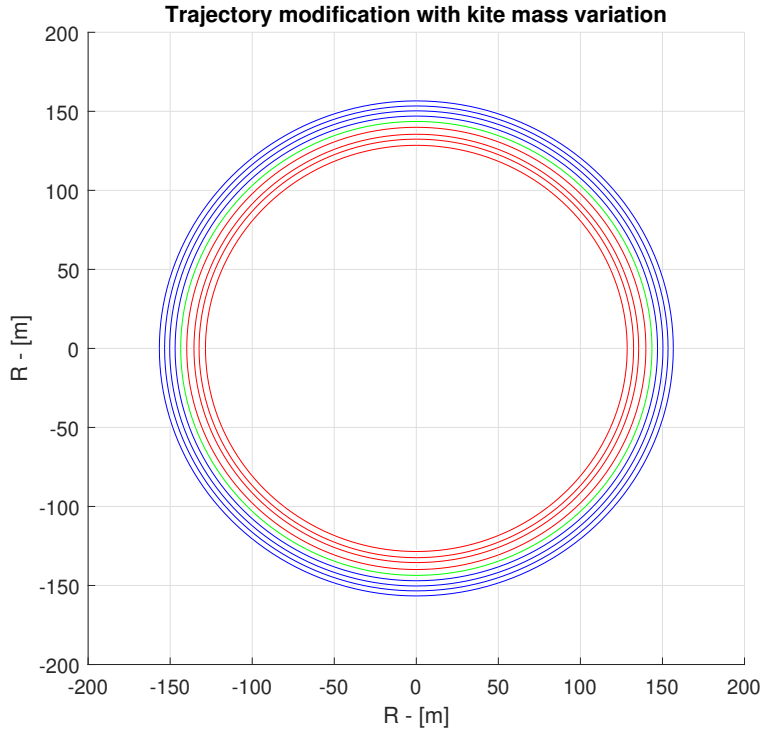


Figure 4.12: Visual representation of the trajectories.

While increasing the mass up to a +20%, the computational steps required to trim the aircraft are reducing.

Since the mass is the only variable and the inertia are unchanged, what was expected is that the radius of the circular trajectory will increase while increasing the mass and vice versa. From table 4.6 it is possible to see the variation of the radius (R_0), highlighted in red, while all the other parameters are strictly the same.
In fig. 4.11, a comparison between the radius of gyration obtained at trim and the same results obtained analytically is presented, demonstrating good behaviour and a slight difference, while in fig. 4.12, the trajectories are visualized.
While testing the lower values of the mass, the trim condition has been reached with an higher number of iterations. This could be traced back to the initial condition:

$$M = \frac{m}{\frac{1}{2}\rho C_L A L_t^0} \tag{4.10}$$

$$\Phi_t = \arccos\left(-\frac{M}{2} + \frac{\sqrt{M^2+4}}{2}\right) \tag{4.11}$$

The mass is influencing the parameter M (eq. (4.10)) that is influencing the angle Φ_t, related with the initial guess of the radius (R_0). Helping the solver with an oculated initial guess helps in reaching the convergence, increasing just the mass could move away the initial value from an optimal guess.
Nevertheless, the results are satisfying since the reduction of the mass implies a reduction of the circular trajectory radius.

4.1.7. Trim solution with tether attachment point variation

Since we are analysing the trim, the tether attachment could be shifted in order to investigate the relative effect. This is not possible while investigating the dynamic of the kite, but could be useful for the trim, to analyze if the trajectory is modifying accordingly and how much.
Just to keep the results simple and straightforward, the only shift is done on the x coordinate of the point.
Moreover, also during this analysis the pitch angle is null.

Table 4.7: Trim condition with tether attachment modification.

	$\mathbf{U_0}\,(m/s)$	$\mathbf{R_0}\,(m)$	$\varepsilon_\mathbf{t}\,(mm/m)$	$\mathbf{\Delta C_{L_a}}$	$\mathbf{\Delta C_{L_e}}$	$\mathbf{\Delta C_{L_r}}$
$\mathbf{X}_{0T}^B = [-0.3, 0, 0]$ m	41.1	144.7	0.51	-0.1016	0.0020	-0.1726
$\mathbf{X}_{0T}^B = [-0.25, 0, 0]$ m	40.99	144.1	0.509	-0.1023	0.0023	-0.1734
$\mathbf{X}_{0T}^B = [-0.2, 0, 0]$ m	40.96	143.4	0.509	-0.1038	0.0025	-0.0323
$\mathbf{X}_{0T}^B = [-0.15, 0, 0]$ m	40.89	142.5	0.507	-0.1060	0.0023	0.2313
$\mathbf{X}_{0T}^B = [-0.1, 0, 0]$ m	40.76	141.4	0.504	-0.1084	0.0021	0.5037
$\mathbf{X}_{0T}^B = [-0.05, 0, 0]$ m	40.51	140.3	0.499	-0.1109	0.0017	0.7905
$\mathbf{X}_{0T}^B = [0, 0, 0]$ m	40.19	139.1	0.493	-0.1136	0.0011	1.0982

Just like before, a graphical representation is reported herein for a better understanding of the behaviour of the parameters, followed by an explanation.

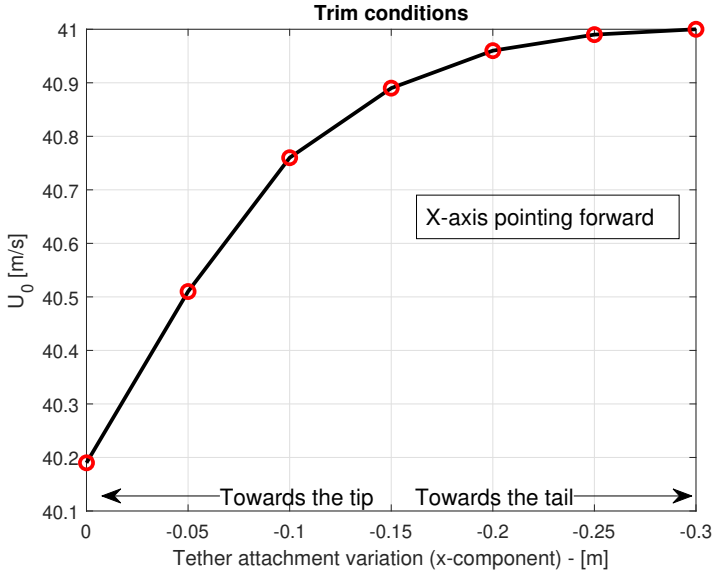
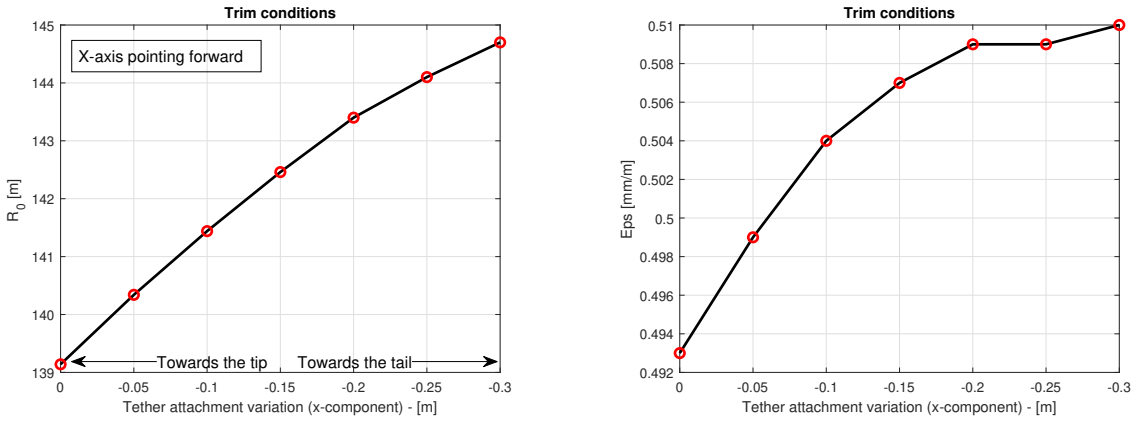


Figure 4.13: Airspeed behaviour with respect to tether attachment point translation along the fuselage, starting from zero and going towards the tail of the kite.

While moving the tether attachment point towards the tip of the aircraft, the computational time increases, with a proportional increase of the rudder lift coefficient offset (ΔC_L). This can be related to the excessive offset introduced to the tether attachment point, purposely introduced to analyse the effect of a drastic repositioning of the tether connection point. Similar results are obtained with a lower computational time when the

tether attachment point is moved towards the kite's tail. Moreover, the airspeed increase is reduced while the tether attachment point is moved towards the tail.



(a) Trajectory radius variation with respect to tether attachment point.

(b) Tether deformation parameter variation with respect to tether attachment point.

Figure 4.14: Trajectory radius and tether deformation variation with the the tether attachment point shift.

An increasing behaviour could also be seen when observing the radius and the tether deformation parameter. As visible from fig. 4.14a, the radius of the trajectory is increasing while moving away the tether attachment point from the centre of mass of the kite ($\mathbf{X}^b_{0CG} = [0, 0, 0]$). The tether attachment point is paramount when analysing stability; even if this is not investigated in the current thesis, the trim condition obtained is robust and adapts to modifying the tether attachment position.

4.1.8. Φ angle and glide ratio (G) investigation

In this subsection a comparison between the Φ angle of the tether and the Glide ratio analytic values and the corresponding results from the trim is performed. This analysis is very useful, if the Φ angle obtained at the trim condition is similar to the initial angle, then the the optimal angle is obtained.
On the other hand, a check on the system glide ratio is paramount. The glide ratio is analytically defined as a velocity ratio between kite linear velocity and wind velocity perpendicular to the rotor-plane. It is comprehensive of the drag acting on the tether:

$$\mathrm{G} = \frac{u_0}{V_w \cos(\beta)} \tag{4.12}$$

But it could be defined as:

$$\mathrm{G} = \frac{C_{L_w} A + C_{L_h} A_h}{C_{D_w} A + C_{D_h} A_h + C_{D_v} A_v + C_{\perp} \frac{\mathrm{d_t L_t^0}}{4}} \tag{4.13}$$

Where:

- C_{L_w} is the wing lift coefficient;
- C_{L_h} is the horizontal tail lift coefficient;
- C_{D_w} is the wing drag coefficient;
- C_{D_h} is the horizontal tail drag coefficient;
- C_{D_v} is the vertical tail drag coefficient;
- $C_{\perp}$ is the tether drag coefficient;
- A is the main wing surface, A_h is horizontal tail wings surface and A_v is the vertical tail wing surface;
- $\mathrm{d_t}$ is the tether diameter and $\mathrm{L_t^0}$ is the tether initial length.

The same is valid for the tether angle, that can be achieved analytically through eq. (4.11), but could it is also be computed at the trim as:

$$\Phi = \arcsin \frac{\mathrm{R_0}}{\mathrm{L_t^0}} \tag{4.14}$$

The results are reported in table 4.8:

Table 4.8: Comparison between analytic values and trim conditions.

	$\mathbf{\Phi_{analyt}}$ (°)	$\mathbf{\Phi_{trim}}$ (°)	$\mathbf{G_{analyt}}$	$\mathbf{G_{trim}}$
$\theta = -2$	21.7	21.4°	8.9	7.5
$\theta = -1$	21.0	20.8°	9.5	7.9
$\theta = 0$	20.4	20.2°	9.8	8.2
$\theta = 1$	19.8	19.6°	10.0	8.5
$\theta = 2$	19.2	19.1°	11.7	8.9
$\theta = 3$	18.6	18.5°	12.4	9.1

The behaviour of the parameters is presented in fig. 4.15 and fig. 4.16.

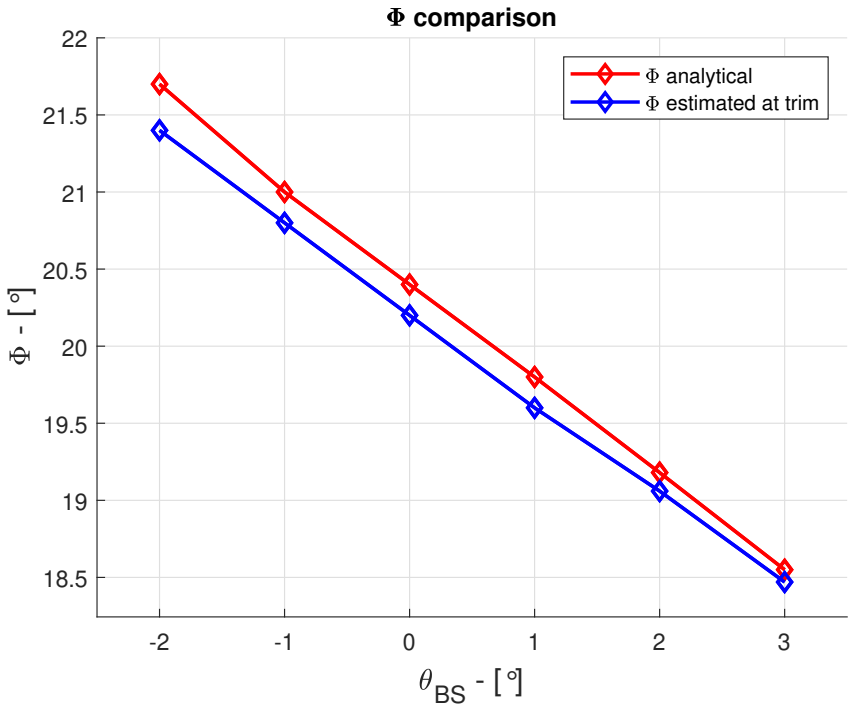


Figure 4.15: Comparison between Φ analytical and at trim as function of the pitch angle.

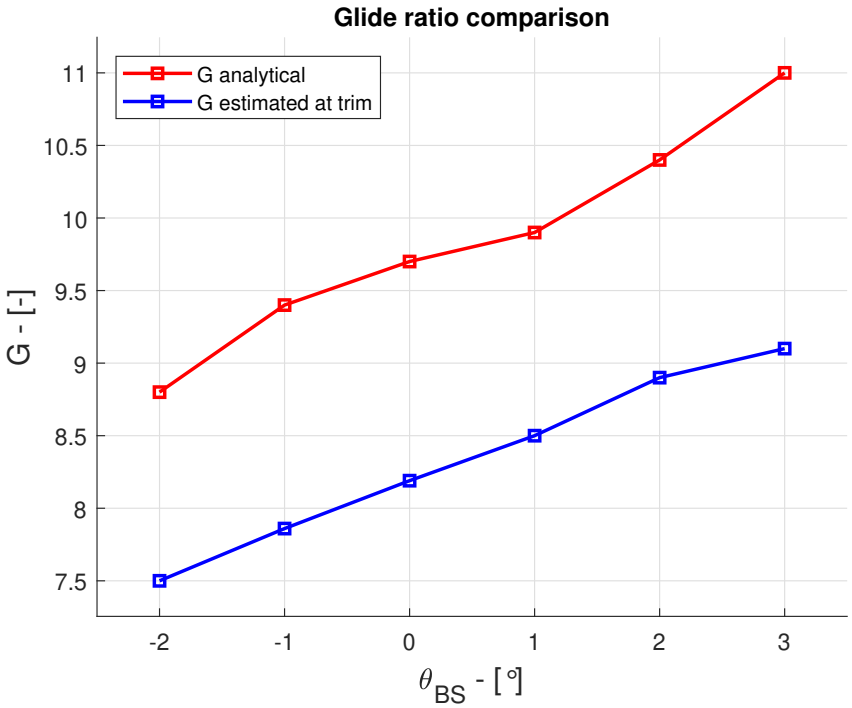


Figure 4.16: Comparison between the analytical Glide ratio and the Glide ratio at trim as function of the pitch angle.

A similar growing behaviour is visible from fig. 4.16, the difference could be traced back to the computation of the analytical glide ratio, in which drag coefficients at the denominator have been slightly underestimated.

4.1.9. Trim condition with the complete tether model

The last comparison is made between the trim condition obtained with the simplified version of the tether and the trim condition obtained with the complete tether model extracted from Delft MegAWEs simulator.
The purpose of this final analysis is to test if the forces obtained with a more complex and complete tether model are similar compared with the simple case tested before and to verify if the VSM model is able to interact with the Delft tether model.
Some assumptions must be made:

- The function is much more complex and involves a dogleg algorithm which can increase the computational time;
- The function requires as an input the tether initial state, the position of the kite, the kite speed and the wind speed in the Ground (Wind) reference frame;
- The wind vector becomes a matrix of 3xN elements, with N the number of the tether masses. In this iteration of the code, the wind is kept constant and uniform. Future developments could introduce a complex environment, including wind shear, and the wind will be modified for each tether mass according to their vertical position with respect to the ground station.

In order to perform a comparison, the trim is modified, introducing as the third variable of the initial guess Z_0 and not ε. The input needed for the function to work are the following:

- Position of the kite in the ground system: $\mathrm{X}_G = [0; \mathrm{R}_0; \mathrm{Z}_0]$, with R_0 and Z_0 that are variables of the trim process, being modified at each step.
- Airspeed of the kite in the ground system: $\mathrm{v}_G = [\mathrm{U}_0; 0; 0]$.
- Wind speed in the ground system, kept constant and uniform.
- Initial force applied on the ground station from the tether: $\mathrm{T}_{0_G} = \mathrm{T}_{mod}\ [0; \sin(\Phi); -\cos(\Phi)]$. Elevation angle ($\beta$) should be included too but due to the absence of gravity, the latter angle is null. Moreover, $\mathrm{T}_{mod} = \varepsilon_t\, \mathrm{E}\, \mathrm{A}$.

A summarize of the iterative process is presented herein, thanks to [48]. Also a figure of the tether quasi-static lumped mass representation is visible in fig. 4.17.

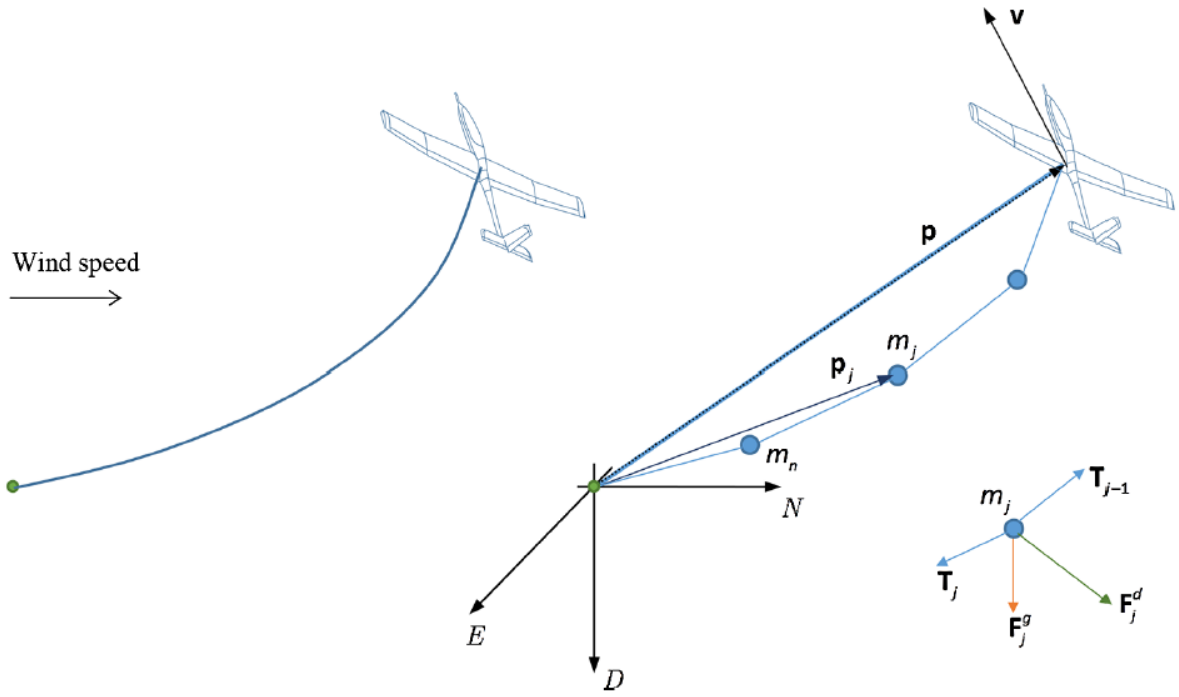


Figure 4.17: Lumped-mass tether model - Source: [48]

1. Choose the number of cable segments, this is performed in the main script of the trim analysis;
2. Set the cable tension at the fixed end, where the winch should be present;
3. Obtain the position (X_G) and the velocity (v_G) of the kite from initial conditions or aircraft dynamics;
4. Divide the cable into N_S elements, and compute their masses ($m_j = \rho_c L_S$), with ρ_c being the tether density, presented in table 4.1;
5. On the first time step, set the initial guess of the unknown parameter vector using the position as the guess of the tether direction, and set the initial tension based on the initial elongation of the cable;
6. On the following time steps, set the initial guess to the solution from the previous step;
7. Compute the angular velocity of the cable ($\omega_t = \frac{\mathbf{p}}{\|\mathbf{p}\|^2} \times \mathbf{v}$);
8. Using the current value of the state (depending on the step), compute the position of the end of the cable by looping over the cable elements ($\mathbf{p}_{j-1} = \mathbf{p}_j + l_{j-1} \frac{\mathbf{T_{j-1}}}{\|\mathbf{T_{j-1}}\|}$);
9. If the difference $\|\mathbf{p} - \mathbf{p}_0\|$ is greater than a set tolerance, update the parameter vector

using a Newton method (or similar) and go back to the previous step, otherwise, termine the iteration.

Differently from the method used in [20], the potentiality of MATLAB are exploited, using fsolve for the iterative process instead of a personalized dogleg algorithm. The results are reported herein, starting with the trim point obtained with a null pitch and with the new trim problem, including the parameter Z_0 and computing the tether force exploiting the iterative process (table 4.9).
Later on, a comparison of the forces obtained from the simplified tether model and from the iterative model is presented (table 4.10).

Table 4.9: Trim condition using the new tether model.

	$\mathbf{U_0}\,(m/s)$	$\mathbf{R_0}\,(m)$	$\mathbf{Z_0}\,(m)$	$\mathbf{\Delta C_{L_e}}$	$\mathbf{\Delta C_{L_a}}$	$\mathbf{\Delta C_{L_r}}$
@Trim Exact	40.8	132.73	-338	-0.1021	-0.4940	0.1888
@Trim Approx	40.9	143.56	-389	-0.0976	-0.4908	0.1747
Error [%]	0.25	8	15	4	0.65	7

Except for the new parameter Z_0, the others are similar to the results obtained with the initial trim condition with the simplified tether model.
The forces are similar, the results are as expected, and further modifications and improvement of the tether model could help speed up the iterative process and obtain better results.

Table 4.10: Tether force comparison.

[kN]	F_{Tx}	F_{Ty}	F_{Tz}
Simplified model	-1.66	-5.75	1.71
Williams model	-1.67	-6.63	1.66

5 | Conclusions, Recommendations and future developments

5.1. Conclusions

This thesis analyses the possibility of exploiting an engineering model of low-fidelity aerodynamics for the conceptual design of AWEs. The introduction and implementation of the VSM can support the analysis of local aerodynamic phenomena. It can be exploited with different geometry configurations, a necessary condition for the conceptual design.

The introduction of an aerodynamic engineering model such as the VLM, located in the low-mid fidelity domain, makes it possible to estimate the aerodynamic characteristics of AWEs. The VSM is computationally fast and can include non-linearity via airfoil polars with results similar to the Tornado VLM. Such a model can be helpful for AWE system's preliminary and conceptual design stages.

This thesis presents a computationally efficient low-fidelity aerodynamic method and an initial trim condition that will be used as an initial guess to integrate non-linear equations through ODEs. The selection of the VSM is due to the outstanding results obtained for LEI kites in [12], and since a three-filament VSM is already implemented in KiteFast. According to the analysis performed in chapter 2, the latter is one of the most promising low-fidelity aerodynamic model available, being able to model key behaviours of fixed-wing kites.

Code validation was necessary due to the new environment in which the aerodynamic model is implemented (MATLAB). The results obtained during the validation followed an incremental approach, from a single elliptic wing to a wing plus horizontal tail. Regarding the elliptic wing:

- The LLT implemented in the code can estimate the lift coefficient with a low percentage error compared to the analytic LLT from the theory; Moreover, it can compute the drag coefficient with an initial high error that is reduced while increasing the

discretisation.

- The VSM implemented in the code can estimate the lift coefficient with a low percentage error; nevertheless, the drag coefficient estimate differs from the analytic LLT. This could be related to the analytical LLT overestimating the lift and drag coefficient, especially at lower aspect ratios. An analysis increasing the aspect ratio confirms that the VLM tends to the LLT analytic value while increasing the aspect ratio.

A dihedral and sweep angle has been added on the same elliptic wing, leading to verification if the code can capture an input modification and modify the geometry accordingly. The results were satisfying, with the code able to estimate the aerodynamic stability derivative with respect to the sideslip, returning values similar to the ones predicted by the VLM. Finally, the horizontal tail is introduced, leading to a double confirmation. A secondary lifting surface can be easily added to the code, and the aerodynamic forces and moments are modified accordingly. Moreover, it can estimate the downwash angle, even if the result estimated from the code is lower than the analytic one, but with a similar trend.

The results of the comparison with the Tornado VLM have shown the great potential of the VSM in predicting the aerodynamic performance of both the Zefiro and Delft kites, which have completely different geometries. Several advantages of the code derived from the previous analyses are summarized herein:

- The non-linearity can be considered thanks to the airfoil's polar interpolation, leading to the possibility of capturing local aerodynamic phenomena and introducing control surfaces.
- It is possible to introduce different lifting surface geometries by simply creating a function due to the versatility of the code.
- Different airfoils can be introduced for each lifting surface, leading to the possibility of modifying the airfoil along the surface.
- It has a computational time similar to the Tornado VLM, highly utilised for aircraft conceptual design while remaining a vortex filament method belonging to the low mid-fidelity domain.
- It is versatile:
  - Changing a simple input string makes it possible to pass from the LLT to the VSM, modifying the control point in which the boundary condition is applied.

- It is possible to pass from the three-filaments horseshoe vortex version to the five-filaments horseshoe vortex version by changing input settings.
- It is possible to consider more lifting surfaces and modify the induction between the latter lifting surfaces by changing an input vector.

Moreover, the Delft MegAWEs simulator has been tested with the direct substitution of the aerodynamic method, introducing the already validated VLM. This resulted in several expected problems, primarily related to the control module used in the framework.
The latter analysis was very useful for understanding how the simulator works and how crucial the control module-aerodynamic module interaction is.

Immediately after, the trim analysis was performed. Introducing a vertical tail and a straightforward control system is paramount to trimming the kite since the rudder is fundamental.
A generic trim problem has been created starting from an existing model, defining the constraint equations together with an initial guess. Several parameters were varied to test the trim's ruggedness, verifying if a proper computational time was obtained. The physics of the trim solution was verified, the forces and moments acting on the kite were coherent with the reference systems introduced, and the kite behaved as expected. While testing the robustness of the trim, several primary values were varied, and the behaviour of the solution at each trim point showed the correctness of the trim problem and related state variable and constraint equations.

Since the beginning of the work, the idea has been to estimate the aerodynamic characteristic of fixed-wing kites (FG or GG AWEs) by introducing an efficient aerodynamic method, starting from understanding the AWEs world and delving deeper into the computational aspect by studying AWEs simulation tools available. A modification of the aerodynamic module was necessary to analyse complex and general cases while remaining in the low-fidelity domain. The introduction of the VSM and the implementation of a MATLAB version were selected after evaluating various candidates, leading to the creation of an effective and practical aerodynamic model that could be used for the conceptual design of AWEs. The last step was to create a specific trim problem to test the aerodynamic method and to achieve a suitable starting point for future real-time dynamic simulation of kites.
Future research could lead to general improvements and modification of the method by introducing more complex aerodynamic concepts.

5.2. Recommendations and future developments

This section presents several ideas for further research and developments of this model or related studies.

5.2.1. Aerodynamic model issues and future developments

Implementing new MATLAB functions could introduce new lifting surfaces with new geometries as long as the coordinate vector is correctly written. While developing the thesis, just an elliptic wing type is used to stick with the geometry of the Zefiro kite.

A different solver than fsolve could be introduced alongside a computational time study and improvements to the code's general stability and computational efficiency.

A vortex core correction is applied in [12] and [14]. It was beyond the purpose of this thesis to investigate the introduction of the core correction. However, the function used to find the induced velocities is created to introduce the vortex core correction during future modifications.

The most relevant issue to be investigated is the numerical error presented while trimming the kite exploiting the VSM. The error is undoubtedly related to a numerical error. It was visible since the VSM had a high computational time during the trim, and the related results highlighted a high value for the control surfaces lift coefficient variation at trim. This is traced back to the aircraft polar interpolation, which depends on the induced velocities. These, in turn, are obtained from the AIC matrices, in which higher values than expected are present. Before concluding the thesis, a severe analysis was performed trying to isolate the error, leading to two possible areas of investigation:

- A problem related to how the code detects the induction between surfaces, especially vertical tail - horizontal tail induction and the vertical tail / horizontal tail induction on itself;
- A problem in how the code detects the geometric discretisation since an unexpected jump is presented during specific steps while passing from one surface to the other, leading to a higher value than expected as visible in fig. 4.10.

Airfoil data extraction The power of the VSM lies in the interpolation of the airfoil polars, this part could be heavily modified:

- Possibility of modifying the airfoil along the lifting surfaces span creating a sort of aerodynamic twist;

- Possibility of extracting the data directly from XFOIL or XFLR5 using a communication block between Matlab and those program;
- Possibility of introducing higher fidelity airfoil data through CFD database, if needed.

5.2.2. Helicoidal wake modification

The modelling of the aerodynamic wake of AWEs is crucial for estimating their performance and design. The VSM can model the induced velocities at the AWEs in a consistent way. In this first version of the code, a straight wake is used, but according to Trevisi et al. [45], the introduction of a vortex model for the wake, exploiting helicoidal filaments, is possible. The modelling of the near wake is performed in the paper by exploiting a lifting line method. A future modification of the code could be focused on introducing this type of wake; according to [45], the model is suitable for time-marching aero-servo-dynamic-elastic simulations.

5.2.3. Trim condition issues and future developments

The trim condition exploited in this thesis is based on six constraint equations, and it is not the usual trim problem obtained with twelve equations, including the Euler angles.
The first issue with the investigations during trim is related to certain values of the prescribed pitch angle. Using these particular values, certain iterations' computational time greatly increases.
The trim problem is stiff, but the computational time has been kept low thanks to the assumptions introduced and the oculate choice of the initial guessing.
The biggest issue was the implementation of the vertical tail and control surfaces, which created a minor numerical bug, necessitating exploiting the LLT for the trim analysis results. Once the problem is solved, a trim analysis with the VSM could be performed, comparing the results with the LLT analysis. This trim condition could be used as a starting point once a dynamic simulation for the kite is needed.
Finally, having the possibility of relying on an efficient and general aerodynamic method, a physically feasible trim condition and an oculate ODE simulation, the kite could be exploited as a **sensor**.

5.2.4. Airborne Wind Energy systems exploited as Sensors

Once the simulation framework is complete (MATLAB or Simulink), it should be possible to take advantage of it and analyse a particular aspect of AWEs.
They give access to high-altitude wind energy extraction, and in the future, they will play a big role in the energy sector, greatly supporting actual wind turbines.
As it is now, while many factories and researchers are working to obtain a greater energy extraction from Kite systems, many different possibilities can be explored.
Namely, the possibility to exploit them as sensors to be put in front of wind turbine farms to extract relevant wind information, in particular **wind intensity, wind direction** and **wind shear**.
Several theoretical developments are present in the literature, like the one visible in [41], in which the filtering approach is fed with measurements available at the ground station, namely, the line angles and their rates, the traction force on the tether, and the wind speed and direction a few meters above the ground. The brilliant work presented in the paper shows that the algorithm can effectively estimate both the state and the aerodynamic parameters of the wings, as well as the wind speed and direction at the wing's altitude. The real time estimation of these parameters is useful, in fact:

1. Wind characteristics at wing's altitude and position are known with a limited accuracy;
2. Even if the aerodynamics characteristics were initially known, they may change significantly during operations due to the environmental conditions, command inputs or disturbances;
3. They could be used, as previously stated, to help in data analysis and prevention inside a wind farm, by putting the kite in a oculate position in front of the wind turbines.
4. Monitor the system conditions for the sake of fault detection and predictive maintenance.
5. Perform a study campaign of a suitable zone for a wind farm installation. Analysing the wind conditions during a certain amount of time to find the suitable orientation and height of the tower.

Bibliography

[1] Awesco site:, 2022. URL http://awesco.eu/awe-explained/.

[2] Enerkite website:, 2022. URL https://www.enerkite.de/en.

[3] Kitegen website:, 2022. URL http://www.kitegen.com/en/s.

[4] Kitenergy website:, 2022. URL https://kitenrg.com/.

[5] Minesto project:, 2022. URL https://minesto.com/projects/faroe-islands.

[6] Skypull website:, 2022. URL https://www.skypull.technology/solution.

[7] C. L. Archer. An introduction to meteorology for airborne wind energy. In *Airborne wind energy*, pages 81–94. Springer, 2013. doi: 10.1007/978-3-642-39965-7_5.

[8] M. J. Bhagwat and J. G. Leishman. Generalized viscous vortex model for application to free-vortex wake and aeroacoustic calculations. In *Annual forum proceedings-American helicopter society*, volume 58, pages 2042–2057. American Helicopter Society, Inc, 2002.

[9] S. Bouckaert, A. F. Pales, C. McGlade, U. Remme, B. Wanner, L. Varro, D. D'Ambrosio, and T. Spencer. Net zero by 2050: A roadmap for the global energy sector. 2021. URL https://iea.blob.core.windows.net/assets/063ae08a-7114-4b58-a34e-39db2112d0a2/NetZeroby2050-ARoadmapfortheGlobalEnergySector.pdf.

[10] M. Canale, L. Fagiano, and M. Milanese. Power kites for wind energy generation [applications of control]. *IEEE Control Systems Magazine*, 27(6):25–38, 2007. URL https://doi.org/10.1109/MCS.2007.909465.

[11] M. Canale, L. Fagiano, and M. Milanese. Kitegen: A revolution in wind energy generation. *Energy*, 34(3):355–361, 2009. URL https://doi.org/10.1016/j.energy.2008.10.003.

[12] O. Cayon, M. Gaunaa, and R. Schmehl. Fast aero-structural model of a leading-edge

inflatable kite. *Energies*, 16(7), 2023. ISSN 1996-1073. doi: 10.3390/en16073061. URL https://www.mdpi.com/1996-1073/16/7/3061.

[13] A. Cherubini, A. Papini, R. Vertechy, and M. Fontana. Airborne wind energy systems: A review of the technologies. *Renewable and Sustainable Energy Reviews*, 51:1461–1476, 2015. URL https://doi.org/10.1016/j.rser.2015.07.053.

[14] R. Damiani, F. F. Wendt, J. M. Jonkman, and J. Sicard. A vortex step method for nonlinear airfoil polar data as implemented in kiteaerodyn. In *AIAA Scitech 2019 Forum*, page 0804, 2019. URL https://doi.org/10.2514/6.2019-0804.

[15] M. De Lellis, A. Mendonça, R. Saraiva, A. Trofino, and A. Lezana. Electric power generation in wind farms with pumping kites: An economical analysis. *Renewable energy*, 86:163–172, 2016. URL https://doi.org/10.1016/j.renene.2015.08.002.

[16] M. De Lellis, R. Reginatto, R. Saraiva, and A. Trofino. The betz limit applied to airborne wind energy. *Renewable Energy*, 127:32–40, 2018. URL https://doi.org/10.1016/j.renene.2018.04.034.

[17] J. R. F. Diógenes, J. Claro, J. C. Rodrigues, and M. V. Loureiro. Barriers to onshore wind energy implementation: A systematic review. *Energy Research & Social Science*, 60:101337, 2020. URL https://doi.org/10.1016/j.erss.2019.101337.

[18] M. Drela. Xfoil: An analysis and design system for low reynolds number airfoils. In *Low Reynolds Number Aerodynamics: Proceedings of the Conference Notre Dame, Indiana, USA, 5–7 June 1989*, pages 1–12. Springer, 1989. doi: 10.1007/978-3-642-84010-4_1.

[19] D. Eijkelhof. Design and optimisation framework of a multi-mw airborne wind energy reference system. 2019. URL http://resolver.tudelft.nl/uuid:e759f9ad-ab67-43b3-97e0-75558ecf222d.

[20] D. Eijkelhof and R. Schmehl. Six-degrees-of-freedom simulation model for future multi-megawatt airborne wind energy systems. *Available at SSRN 4003237*, 2022. URL https://doi.org/10.1016/j.renene.2022.06.094.

[21] J. Heilmann and C. Houle. Economics of pumping kite generators. In *Airborne Wind Energy*, pages 271–284. Springer, 2013. doi: 10.1007/978-3-642-39965-7_15.

[22] J. Jonkman. Google/makani energy kite modeling: Cooperative research and development final report, crada number crd-17-00569. Technical report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2021. URL https://doi.org/10.2172/1813012.

[23] J. Jonkman, G. Hayman, R. Mudafort, R. Damiani, F. Wendt, G. Incorporated, and USDOE. Kitefast, 8 2018.

[24] J. Katz and A. Plotkin. *Low-speed aerodynamics*, volume 13. Cambridge university press, 2001.

[25] P. Masarati. Mbdyn input file format version 1.7. 3. *Politecnico di Milano*, 2017.

[26] B. W. McCormick. *Aerodynamics, aeronautics, and flight mechanics*. John Wiley & Sons, 1994.

[27] T. Melin. User's guide and reference manual for tornado. *Royal Inst. of Technology (KTH), Stockholm, Sweden*, 2000.

[28] R. Mukherjee, A. Gopalarathnam, and S. Kim. An iterative decambering approach for post-stall prediction of wing characteristics from known section data. In *41st aerospace sciences meeting and exhibit*, page 1097, 2003. URL `https://doi.org/10.2514/6.2003-1097`.

[29] M. M. Musk. *The minimum induced drag of aerofoils*. US Government Printing Office, 1921.

[30] M. D. L. C. d. Oliveira et al. Airborne wind energy with tethered wings: modeling, analysis and control. 2016. URL `https://repositorio.ufsc.br/xmlui/handle/123456789/173661`.

[31] A. Pastor-Rodríguez, G. Sanchez-Arriaga, and M. Sanjurjo-Rivo. Modeling and stability analysis of tethered kites at high altitudes. *Journal of Guidance, Control, and Dynamics*, 40(8):1892–1901, 2017. URL `https://doi.org/10.2514/1.G002550`.

[32] W. Phillips, E. Anderson, J. Jenkins, and S. Sunouchi. Estimating the low-speed downwash angle on an aft tail. *Journal of aircraft*, 39(4):600–608, 2002. URL `https://doi.org/10.2514/2.2998`.

[33] M. Ranneberg. Direct wing design and inverse airfoil identification with the nonlinear weissinger method. *arXiv preprint arXiv:1501.04983*, 2015. URL `https://doi.org/10.48550/arXiv.1501.04983`.

[34] S. Rapp, R. Schmehl, E. Oland, and T. Haas. Cascaded pumping cycle control for rigid wing airborne wind energy systems. *Journal of Guidance, Control, and Dynamics*, 42(11):2456–2473, 2019. URL `https://doi.org/10.2514/1.G004246`.

[35] J. Roskam. *Airplane Design: Preliminary Calculation of Aerodynamic, Thrust and Power Characteristics*. Airplane Design. DARcorporation. ISBN

9781884885525. URL https://books.google.it/books?id=L60uqKL2q10C&hl=it&source=gbs_similarbooks.

[36] P. Sadorsky. Wind energy for sustainable development: Driving factors and future outlook. *Journal of Cleaner Production*, 289:125779, 2021. URL https://doi.org/10.1016/j.jclepro.2020.125779.

[37] M. H. Sadraey. *Aircraft design: A systems engineering approach*. John Wiley & Sons, 2012.

[38] G. Sánchez-Arriaga, M. García-Villalba, and R. Schmehl. Modeling and dynamics of a two-line kite. *Applied Mathematical Modelling*, 47:473–486, 2017. URL https://doi.org/10.1016/j.apm.2017.03.030.

[39] G. Sanchez-Arriaga, A. Pastor-Rodríguez, R. Borobia-Moreno, and R. Schmehl. A constraint-free flight simulator package for airborne wind energy systems. In *Journal of Physics: Conference Series*, volume 1037, page 062018. IOP Publishing, 2018. doi: 10.1088/1742-6596/1037/6/062018.

[40] G. Sánchez-Arriaga, A. Pastor-Rodríguez, M. Sanjurjo-Rivo, and R. Schmehl. A lagrangian flight simulator for airborne wind energy systems. *Applied Mathematical Modelling*, 69:665–684, 2019. URL https://doi.org/10.1016/j.apm.2018.12.016.

[41] E. Schmidt, M. D. L. C. de Oliveira, R. S. da Silva, L. Fagiano, and A. T. Neto. In-flight estimation of the aerodynamics of tethered wings for airborne wind energy. *IEEE Transactions on Control Systems Technology*, 28(4):1309–1322, 2019. URL https://doi.org/10.1109/TCST.2019.2907663.

[42] I. Tani. A simple method of calculating the induced velocity of a monoplane wing. *title Report of Aeronautical Research Institute, Tokyo Imperial University* , 9(111): 64, 1934.

[43] F. Trevisi. Awes and wind turbine comparison presentation. Presentation for a private meeting inside Politecnico, 12 2021.

[44] F. Trevisi, A. Croce, and C. E. Riboldi. Flight stability of rigid wing airborne wind energy systems. *Energies*, 14(22):7704, 2021. URL https://doi.org/10.3390/en14227704.

[45] F. Trevisi, C. E. D. Riboldi, and A. Croce. Vortex model of the airborne wind energy systems aerodynamic wake. *Wind Energy Science Discussions*, pages 1–31, 2023. URL https://doi.org/10.5194/wes-2023-25.

[46] C. Vermillion, M. Cobb, L. Fagiano, R. Leuthold, M. Diehl, R. S. Smith, T. A. Wood, S. Rapp, R. Schmehl, D. Olinger, et al. Electricity in the air: Insights from two decades of advanced control research and experimental flight testing of airborne wind energy systems. *Annual Reviews in Control*, 52:330–357, 2021. URL `https://doi.org/10.1016/j.arcontrol.2021.03.002`.

[47] J. Weissinger. The lift distribution of swept-back wings. Technical report, 1947.

[48] P. Williams. Cable modeling approximations for rapid simulation. *Journal of guidance, control, and dynamics*, 40(7):1779–1788, 2017. URL `https://doi.org/10.2514/1.G002354`.

A | Appendix A - Simulink blocks, algorithms and programming

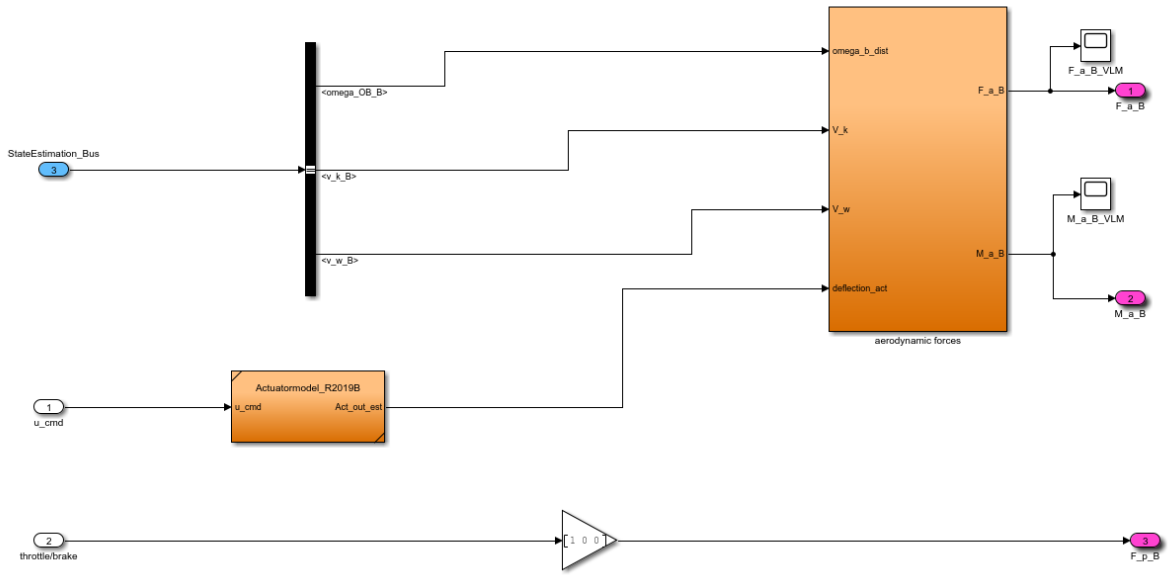


Figure A.1: Aerodynamic model pt.1

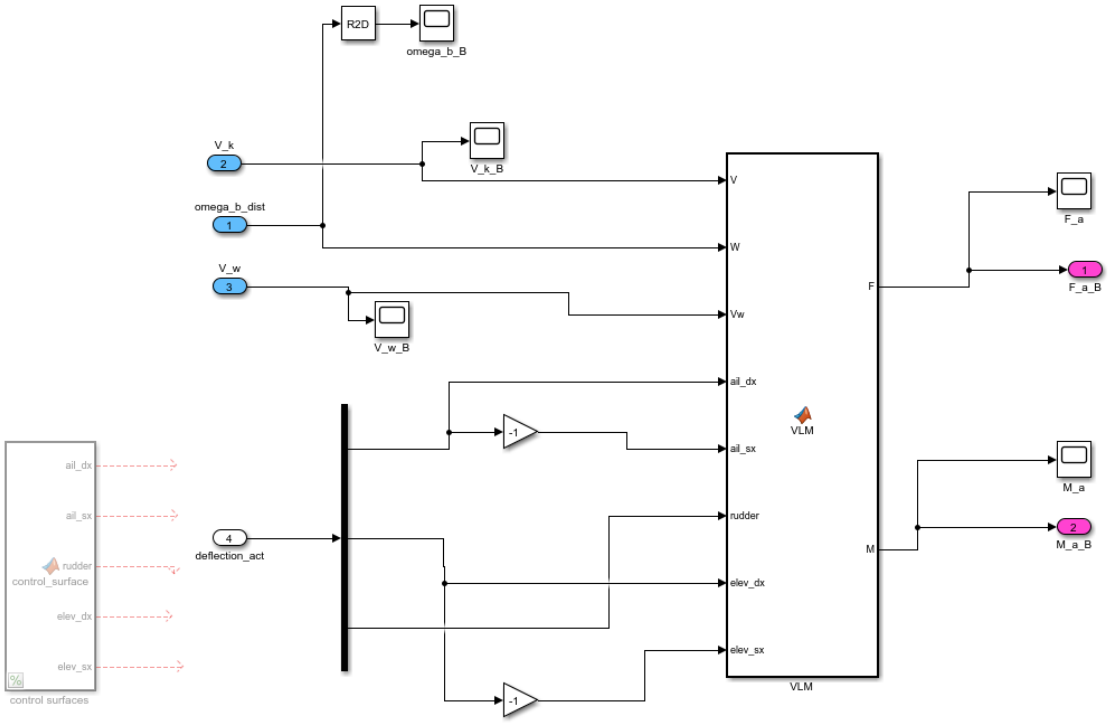


Figure A.2: Aerodynamic model pt.2

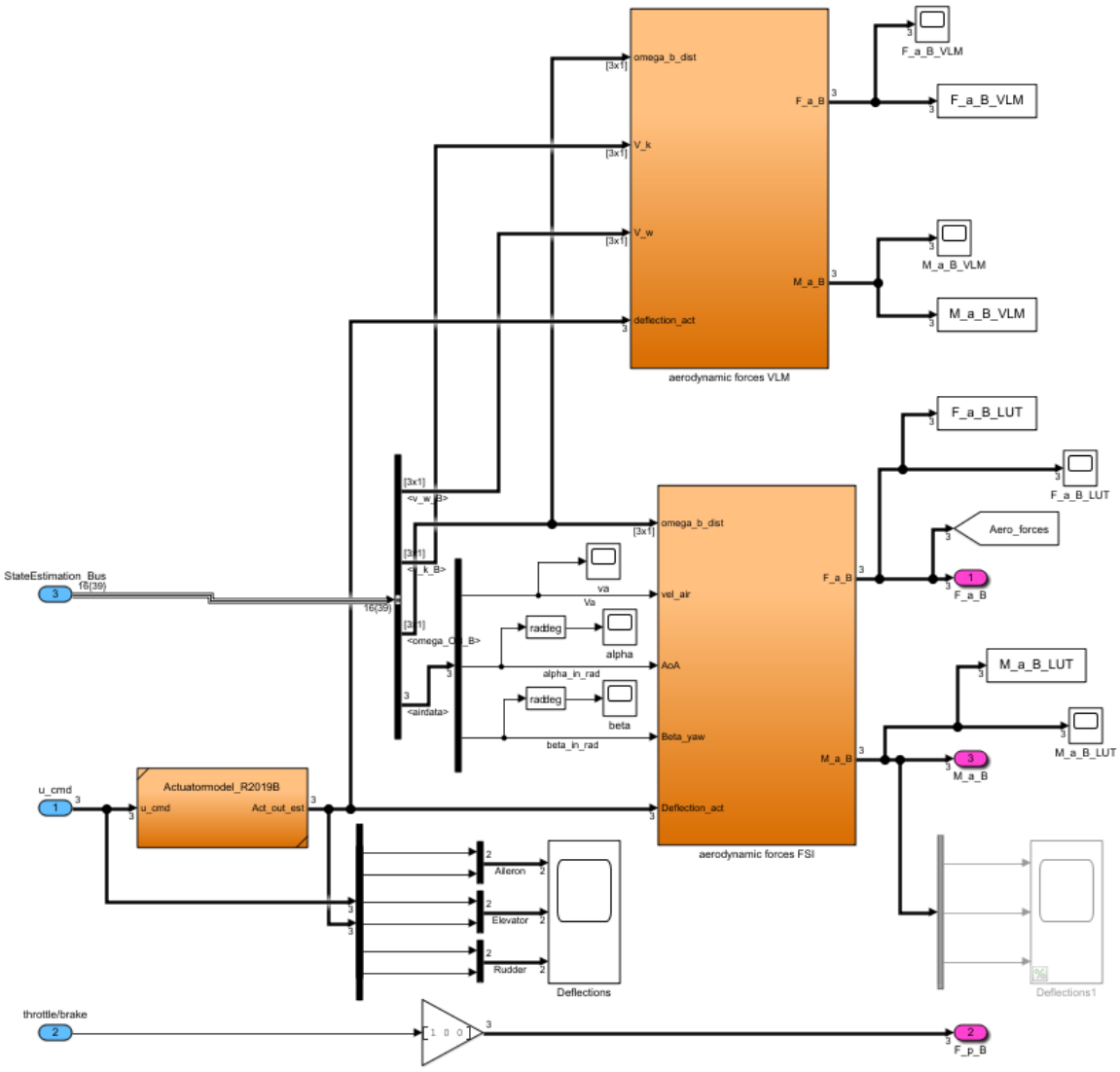


Figure A.3: Aerodynamic models in parallel pt.2

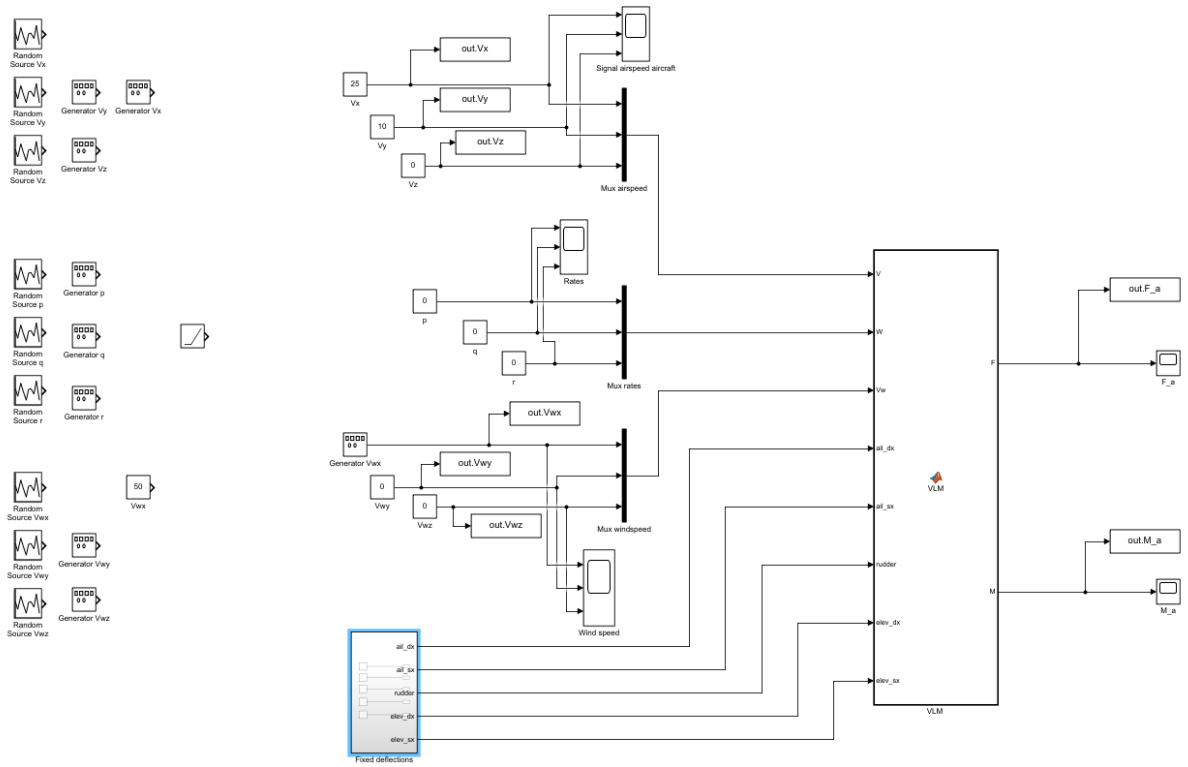


Figure A.4: VLM scheme for prescribed input

Algorithm A.1 VLM main function

```
function [F,M] = VLM(V,W,Vw,ail_dx,ail_sx,rudder,elev_dx,elev_sx,geo_in,DE2019)
geo = geo_in
flap = [ail_dx; ail_sx; rudder; rudder; elev_dx; elev_sx];
flap_vector = diag(flap) * geo.partition_matrix_flap;
state=struct('U',V(1),'V',V(2),'W',V(3),'P',W(1),'Q',W(2),'R',W(3),
'Vwx',Vw(1),'Vwy',Vw(2),'Vwz',Vw(3),'phi',0,'theta',0,'psi',0, 'AS',0,'rho',1.225);
state.AS = norm([(state.Vwx-state.U) ; (state.Vwy-state.V) ; (state.Vwz-state.W)]);
[lattice]=setup_lattice(geo, state, flap_vector);
[a, ~, ~]=size(lattice.VORTEX);
res = struct('F',zeros(a,3),'FORCE',zeros(3,1),'M',zeros(a,3),
'MOMENTS', zeros(3,1),'gamma',zeros(a,1));
[res] = compute_forces(res,state,geo,lattice);
F = res.FORCE;
M = res.MOMENTS + cross(DE2019.cog', res.FORCE);
```

The functions related to the simplified version of the VLM are the following:

- An input structure for the geometry of the kite: *geo_in*;
- A function that generates the lattice: *setup_lattice.m*;
- A function to setup the geometry: *setup_geometry.m*;
- A function to fill the vortex structures: *setup_vortex*;
- A function to create and set the wake: *setup_wake*;
- A function needed for the resolution of the system of equations to obtain the circulation distribution: *fastdw*;
- A function for the computation of the forces: *compute_forces.m*.

Regarding the VSM, the model and program specifics are presented in fig. A.5, followed by a brief explanation of each variable.

```
global settings T ENVM

%% Model and program specifics for VSM
settings.ring_geo = '5fil'; % it can be '3fil' or '5fil', meaning each horseshoe vortex has 3 filaments or 5.
settings.model = 'LLT'; % 'VSM' - Vortex Step Method, 'LLT' - Lifting Line Theory
settings.infl_surf = {'w','h','v'};  % Only wing {'w'} - wing+horizontal tail {'w','h'} - wing+h.tail + v.tail {'w','h','v'}
settings.infl_blocks =  [1,1,1,2,2,2,3,3,3 ; 1,2,3,1,2,3,1,2,3]; % Only wing [1 ; 1] - wing + horizontal tail [1,1,2,2 ; 1,2,1,2] - wing + com
settings.nocore = false; % must remain false, to shut down core correction change to true (to be developed)
settings.iter = "init";
settings.id = 'WingTail'; % It can also be 'Wing' only
```

Figure A.5: VLM model and program specifics

Where:

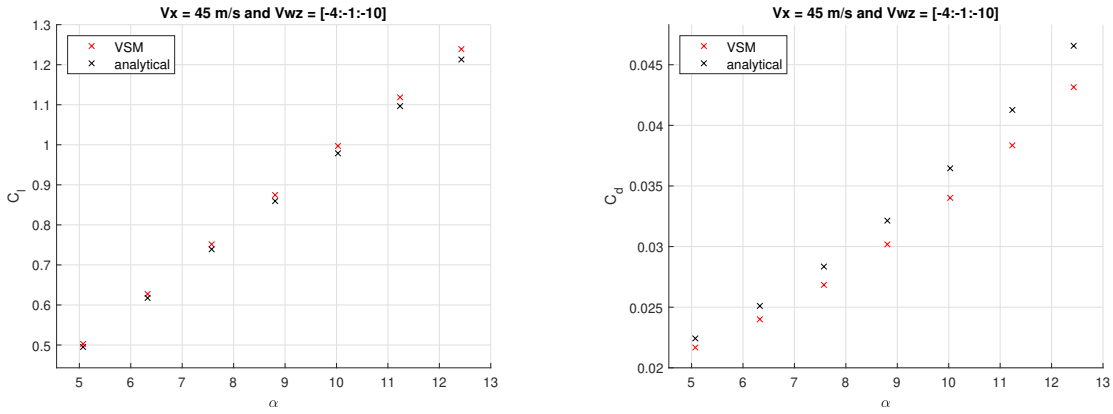
- *settings* is a global structure;
- *settings.model* is a string defining which model is used (LLT or VSM);
- *settings.infl_surf* is a cell defining how many lifting surfaces are considered in the program;
- *settings.infl_blocks* is a two-row vector defining the interaction between surfaces: 1 represents the wing, 2 represents the horizontal surface and 3 represents the vertical surface for example:
  - $[1, 1, 1; 1, 2, 3]$ means the wing horseshoe vortices induce on the wing itself, on the horizontal tail control points and on the vertical tail control points, with the AIC matrices being populated accordingly.
  - $[1, 1, 1, 2, 2, 2, 3, 3, 3; 1, 2, 3, 1, 2, 3, 1, 2, 3]$ is the complete case with all the cross inductions between the surfaces.

- *settings.nocore* is the string for the core correction, not implemented in the current version.
- *settings.iter* is a string used to switch the initialisation of the circulation during the fsolve algorithm from a null value at the first iteration to the value of the previous iteration from the second iteration onward.
- *settings.id* is a string used in the first iteration of the code to switch from the 'OnlyWing' to the 'WingTail' case, it should be removed accordingly.

The MATLAB program is just the tool, while the aerodynamic engineering model is the aim of the thesis; therefore, all the related functions will not be presented in the appendix. The code will be open-source, and it will be available online once all the minor bugs and issues are solved.

B | Appendix B - Graphs and data

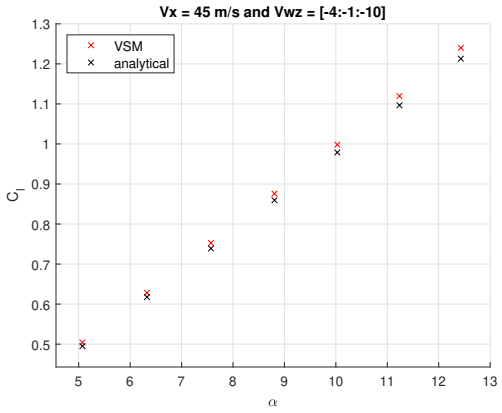
All the additional graphs and results are presented in this appendix.



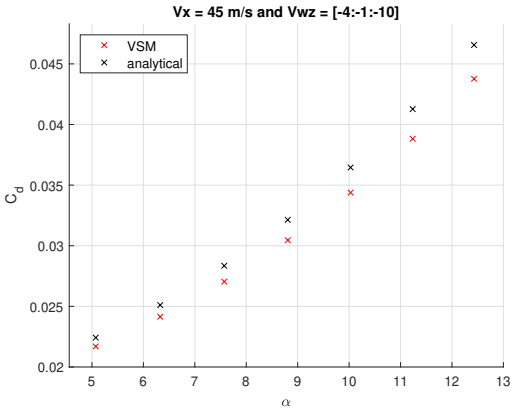
(a) $C_l - \alpha$ Elliptic wing.

(b) $C_d - \alpha$ Elliptic wing.

Figure B.1: Elliptic wing - case with Vx = 45 m/s , variable wind speed along z-axis , 30 aerodynamic sections along the wing and five filaments

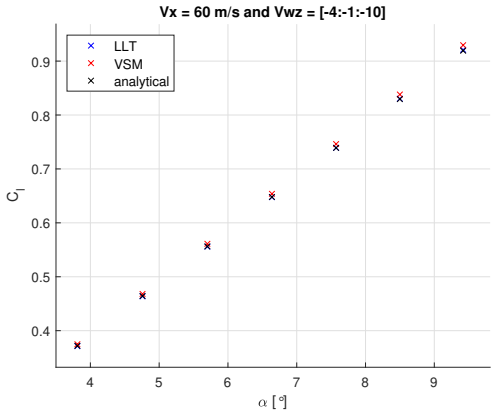


(a) $C_l - \alpha$ Elliptic wing.

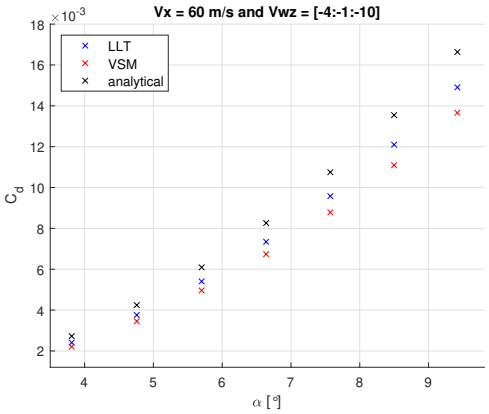


(b) $C_d - \alpha$ Elliptic wing.

Figure B.2: Elliptic wing - case with Vx = 45 m/s , variable wind speed along z-axis , 50 aerodynamic sections along the wing and five filaments



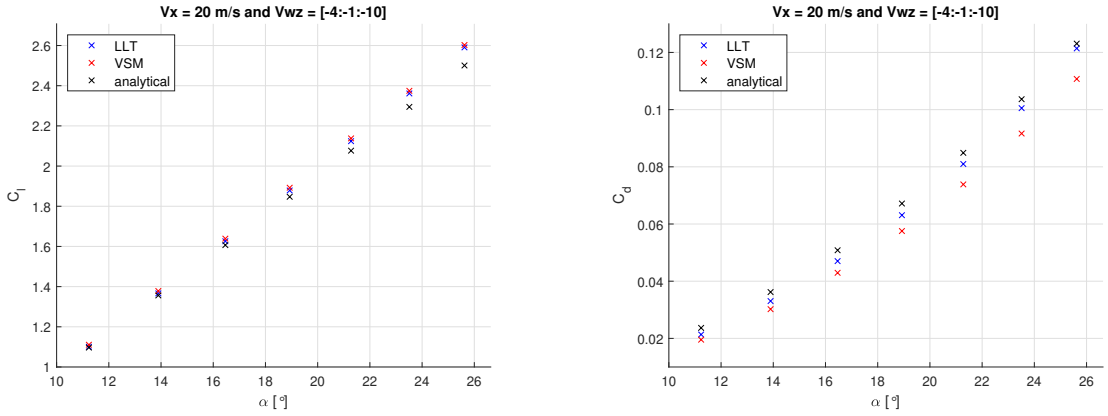
(a) $C_l - \alpha$ Elliptic wing.



(b) $C_d - \alpha$ Elliptic wing.

Figure B.3: Elliptic wing - case with Vx = 60 m/s , variable wind speed along z-axis and 11 aerodynamic sections along the wing.

In fig. B.3a a higher value for Vx have been used. The percentage error is equal to 1% for $\alpha = 9.4°$ for the VSM and equal to 0.18% for the same α but computed exploiting the LLT.



(a) $C_l - \alpha$ Elliptic wing.

(b) $C_d - \alpha$ Elliptic wing.

Figure B.4: Elliptic wing - case with Vx = 20 m/s , variable wind speed along z-axis and 11 aerodynamic sections along the wing.

List of Figures

List of Tables

List of Symbols

Variable	Description	SI unit
I_{xx}	Inertia around x-axis	$kg\,m^2$
I_{yy}	Inertia around y-axis	$kg\,m^2$
I_{zz}	Inertia around z-axis	$kg\,m^2$
α	Angle of attack	◦
α_{0_L}	Zero-lift line angle of attack	◦
c	Chord	m
b	Span	m
AR	Aspect ratio	-
S	Surface	m^2
e	Efficiency factor	-
C_D	Drag coefficient	-
C_L	Lift coefficient	-
C_{l_w}	Wing lift coefficient	-
C_m	Moment coefficient	-
C_N	Normal drag coefficient	-
$C_\perp$	Tether drag coefficient	-
ΔC_L	Delta in Lift coefficient due to control surfaces	-
Γ	Dihedral angle	◦
Λ	Sweep angle	◦
θ	Pitch angle	◦
β	Elevation angle	◦
m	Mass	kg
ρ_t	Linear density	-
Γ_i	Circulation	-
ε_1 ε_2	Core radius	m
ε	Downwash angle	◦
ν	Air kinematic viscosity	Sk
E	Axial elastic modulus	-
G	Glide ratio	-
U_{rel}	Relative airspeed	m/s
n	Normal versor	-
t	Tangential versor	-
δ_f	Airfoil's flap or aileron deflection	◦
k_p	Position factor	-
T_n	Tether tension	-
U_0	Kite speed in $\mathcal{RF}_S$	m/s
R_0	Radius of gyration of the trajectory	m
ε_t	Tether deformation parameter	m/s
d_t	Tether diameter	cm
L_0	Tether initial length	cm

List of acronym

Acronym	Description
AIC	Aerodynamic influence coefficient matrix
AWEs	Airborne wind energy system
CFD	Computational fluid dynamics
DoF	Degree of freedom
EKF	Extended kalman filter
FEG	Flying electric generator
FG	Flight generation
FSI	Fluid Structure interaction
GG	Ground generation
GS	Ground station
HAWT	Horizontal axis wind turbine
KF	Kalman filter
KJ	Kutta-Jouwkosky
LASKA	LAgrangian KiteSimulators
LCOE	Levelized cost of energy
LLT	Lifting line theory
LUT	Look up table
ODE	Ordinary Differential Equation
VAWT	Vertical axis wind turbine
VLM	Vortex lattice method
VSM	Vortex step method
WE	Wind energy

Acknowledgements

I want to acknowledge the Politecnico di Milano, it has been a path of ups and downs, but such is life. Politecnico di Milano taught me how to face the most difficult challenges, recognize and deal with problems, and manage complex projects. Thanks to all my colleagues who accompanied me on this university journey, with whom I shared exhausting labours during exams and assignments. Especially Riccardo Falsetti, who has been with me since the first day of my Bachelor's until the last exam of the Master's performed as a team of two. I owe you a lot, and I hope you will find satisfaction in life by doing the thing at your best as you have always done during these years, with a smile on your face and sharing happiness with everyone. Thanks to my roommates, with whom I laughed and joked at the end of heavy days.

I want to acknowledge my Prof. Alessandro Croce and co-advisor Carlo E.D. Riboldi for their beautiful courses and for being my advisors during the thesis. Thanks to Filippo Trevisi, who followed and supported me even during the delusions. Thanks to them, I was able to learn about the AWE world, attend an industry conference and meet so many wonderful people.

A huge thank you to my sweetheart, Bianca, for helping me, calming me down and supporting me whenever I was under pressure or demoralized, for the joys shared during this journey that I have also completed thanks to you. I will always be there for you and hope to help you at least half as much as you did.

Thanks also to my friends in Mantova, who eased my weekend. Thanks for your patience and outings, especially our bond, which never cracked during university.

Lastly, thanks to my parents, who supported me morally and financially, spurring me to give my best without asking for anything in return. This is the end of my university journey but the beginning of a new one. I will repay you for everything and carry you in my heart forever, even if the future will take me far away.