

UNIVERSITY OF TRIESTE

Data Science and Scientific Computing Degree Course

Wind Energy Harvesting with Reinforcement Learning

Master Thesis in Data Science

Author:
Nicole Orzan

Supervisor:
Prof. Antonio Celani

Academic Year 2019 - 2020

Acknowledgements

I wish to say a big thank you to my supervisor, Prof. Antonio Celani, for the opportunity of working on this interesting and exciting thesis project, and for his continuous support and patient guidance. It has been a pleasure to work under your supervision.

A special thank goes to Claudio Leone for his smart suggestions for solving the problems encountered during this work, for the countless hours spent discussing, and for the help in the thesis proofreading. You have been a great ally in this work, thank you.

I wish to thank all the guys of the Quantitative Life Science research group at the ICTP, for being very friendly and making it a beautiful place where to work.

Last but not the least, a huge thanks to my family and friends for their constant encouragement and wonderful support during this journey. All of this would not have been possible without you.

Abstract

Airborne wind energy is a new technology which aims at extracting energy from high-altitude winds by using airborne devices connected to the ground, which need to be automatically controlled during their motion. In this thesis we face the problem of controlling the trajectory of an airborne wind energy system by means of reinforcement learning, which is a learning method based on trial-and-error. To perform the training process, we used the SARSA algorithm for control. The modeled airborne wind energy system consists of a kite of 5 m^2 connected to a block of 40 kg by means of a tether with a fixed length of 50 meters, and it is represented in figure 1. The kite exploits the aerodynamic forces to pull the block, which is bound to slide on a rail on the ground. The kite control system is the decision-making agent of the learning process, and takes actions with the goal of pulling the block so that it reaches the maximum possible distance in a given interval of time. In order to do that, the control system is given the opportunity to act by changing the attack and bank angles depending on the situation.

To start with, we have studied a simplified two-dimensional version of the system, where only the vertical and the longitudinal directions have been considered. In this setting the agent is able to vary the attack angle only. This model has been trained both in a uniform horizontal wind and in a spatially periodic flow.

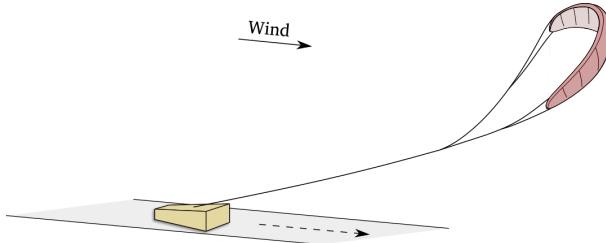


Figure 1: Example of a kite pulling a block along a rail on the ground exploiting the aerodynamic forces.

Next we focused on the full three-dimensional model, which was immersed in a constant wind of 10 m/s blowing along the direction of the rail. Upon allowing the agent to measure the angle between the horizontal and the component of the relative velocity in the vertical plane, a non-trivial maneuvering strategy has been found, where the kite moves in an helix-shaped trajectory, as shown in figure 2. We show that following this policy the distance traveled by the block increases six times with respect to using a policy in which the kite proceeds constantly in the direction of the wind. This trajectory leads to a better performance because it allows the kite to move almost crosswind,

a configuration which is known to be the most effective for power harvesting. To overcome the friction force and move the block, the constant flight configuration extracts a total of 48.53 W from the wind, while the helix-shape flight extracts a total of 953.84 W. If we wanted to use this system to collect power by substituting the block with a ground station, the helix-shape flight would allow to collect a maximum of 30.47 kW compared to 0.095 kW for the constant flight, an improvement by a factor of more than 300.

In short, we have shown how, by using reinforcement learning, we can find nearly-optimal policies and discover new and safe controls for airborne wind energy harvesting.

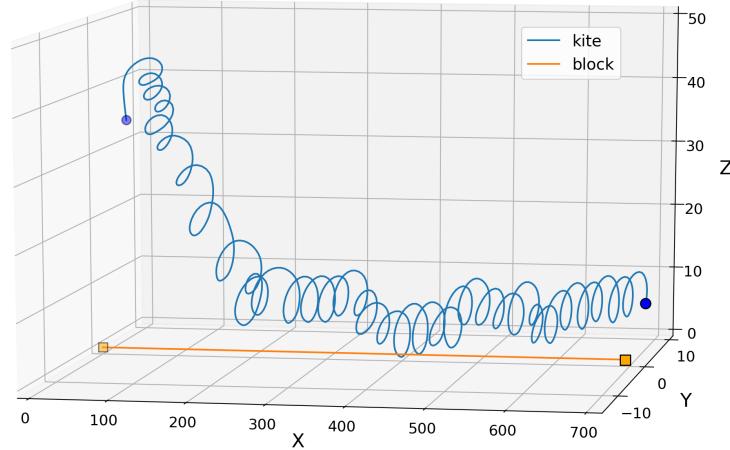


Figure 2: Trajectory followed by the kite and the block using the policy derived from the application of the SARSA learning algorithm, where the kite moves almost-crosswind in an helix-shaped trajectory.

Riassunto

L'eolico d'alta quota (airborne wind energy) è una nuova tecnologia che mira a estrarre energia dai venti di alta quota utilizzando dispositivi aerotrasportati collegati a terra, che devono essere controllati automaticamente durante il loro moto. In questa tesi affrontiamo il problema del controllo della traiettoria di un sistema eolico ad alta quota mediante reinforcement learning, un metodo di apprendimento per tentativi ed errori. Per eseguire il processo di addestramento è stato utilizzato l'algoritmo di controllo SARSA. Il sistema scelto è costituito da un aquilone di $5\ m^2$ collegato a un blocco di 40 kg per mezzo di un cavo con una lunghezza fissa di 50 metri, ed è rappresentato in figura 1. L'aquilone sfrutta le forze aerodinamiche per tirare il blocco, che è costretto a muoversi su una rotaia. Il sistema di controllo dell'aquilone è l'agente decisionale del processo di apprendimento, e compie azioni con l'obiettivo di trainare il blocco facendolo raggiungere la massima distanza possibile in un dato intervallo di tempo. Per fare ciò, al sistema di controllo viene data la possibilità di cambiare l'angolo di attacco e l'angolo di inclinazione laterale dell'aquilone, a seconda della situazione in cui esso si trova.

Per cominciare abbiamo studiato una versione bidimensionale semplificata del sistema, in cui sono state considerate solo le direzioni verticale e longitudinale. In questa impostazione l'agente è in grado di variare solo l'angolo di attacco. Questo modello è stato addestrato sia in un vento orizzontale uniforme che in un flusso periodico nello spazio.

Successivamente ci siamo concentrati sul modello tridimensionale completo, immergendolo in un vento costante di velocità 10 m/s che soffia lungo la direzione della rotaia. Permettendo all'agente di misurare l'angolo tra l'orizzontale e la componente della velocità relativa nel piano verticale, abbiamo trovato strategie di manovra non banale, in cui l'aquilone si muove quasi perpendicolarmente rispetto al vento in una traiettoria a forma di elica, come mostrato in figura 2. Abbiamo mostrato come, con questo comportamento, la distanza percorsa dal blocco misuri sei volte quella in cui l'aquilone procede con un assetto costante nella direzione del vento. Questa traiettoria porta a prestazioni migliori poiché consente all'aquilone di muoversi quasi trasversalmente rispetto al vento, in una configurazione che è nota per essere la più efficace per la raccolta di energia. Per superare la forza di attrito e muovere il blocco, la configurazione in volo costante estrae un totale di 48.53 W dal vento, mentre quella con traiettoria a forma di elica estrae un totale di 953.84 W. Se volessimo utilizzare questo sistema per raccogliere energia sostituendo il blocco con una stazione di terra, il volo con traiettoria a forma di elica consentirebbe di raccogliere un massimo di 30.47 kW rispetto ai 0.095 kW del caso stazionario, più di trecento volte tanto.

In breve, abbiamo dimostrato come, utilizzando l'apprendimento per rinforzo, possiamo trovare strategie quasi ottimali e scoprire metodi di controllo nuovi e sicuri per le tecnologie eoliche d'alta quota.

Contents

I State of the art	2
1 An introduction to Airborne Wind Energy	3
1.1 What is Airborne Wind Energy?	4
1.1.1 Airborne Wind Energy advantages and open problems	4
1.2 The physics of Airborne Wind Energy	6
1.2.1 Power extraction	7
1.2.2 Refinements to power extraction	9
1.3 Airborne Wind Energy technologies and devices	9
1.3.1 Power generation	10
1.3.2 Airborne object	12
1.4 Control techniques	12
1.4.1 Numerical control of the trajectory	13
2 A quick introduction to Reinforcement Learning	15
2.1 Introduction	15
2.2 Finite Markov Decision Process	16
2.3 Value Functions and Bellman Equations	17
2.4 Optimality	18
2.5 Generalised Policy Iteration	20
2.6 Temporal-Difference Learning	20
II Modeling and learning	23
3 Mathematical models	24
3.1 General characteristics of the models	24
3.2 Lift and Drag	26
3.3 Two-dimensional system	28
3.3.1 Polar description of the two-dimensional system	30
3.4 Cartesian model	34
3.4.1 Cartesian description of the two-dimensional model	36
3.4.2 Cartesian description of the three-dimensional model	38
3.5 Dynamics analysis	40
3.5.1 The lift and drag coefficients	41
3.5.2 Equilibrium motion	41

4 Application of Reinforcement learning to the model	46
4.1 General characteristics of the learning	46
4.2 Two-dimensional learning with constant wind	48
4.3 The streamfunction flow	53
4.3.1 Learning in the streamfunction flow	55
4.3.2 Augmenting two-dimensional learning	58
4.4 Three-dimensional learning with constant wind	63
4.4.1 Augmenting three-dimensional learning	64
5 Perspectives	72

At a glance

The work presented in this thesis aims to develop a model of an airborne wind energy system composed by a tethered kite connected to a ground-based block, showing how it is possible for the kite to pull the block using solely the energy extracted from the wind. The novel idea of this work is to control such a system using reinforcement learning techniques, in a way that the decision-making agent, namely the control system of the kite, is able to learn the best way to move into the environment using its own experience.

This work is divided into two main parts. Part I, titled “State of the art”, contains a review of the main theoretical concepts needed to understand the deployed analysis. This part comprises chapters 1 and 2. In Chapter 1 we explain the development of airborne wind energy, motivating its potential to become a great energy source. Follows a brief overview of the technologies and the prototypes already designed in the field of airborne wind energy, together with an explication of past achievements and open questions. Chapter 2 deals with the basics of reinforcement learning, the learning method used to tackle the problem, based on trial-and-error. All the main concepts necessary to explain the algorithm used in the simulations are described in this chapter.

Part II, titled “Modeling and Learning” contains the novel work developed during this thesis project. This part is divided into chapters 3 and 4. Chapter 3 contains the development of the dynamical models of the tethered kite connected to the block. A simplified two-dimensional model is first described, which is later extended to a more realistic three-dimensional model. Chapter 4 provides a detailed explanation of the application of reinforcement learning to the models and a discussion of the achieved results.

Finally, in the “Perspectives” we provide a brief description of the planned future developments.

Part I

State of the art

Chapter 1

An introduction to Airborne Wind Energy

Energy production and consumption and their consequences on the world are one of the biggest issues that today's generation has to face. The global energy market's reliance on fossil fuels, gas and oil needs a sharp change due to the inadequacy in the sustainability of those energy sources and the related carbon dioxide emissions that their combustion is generating. Fortunately things are gradually changing as the world's attention is moving towards renewable sources.

Renewable energy resources, such as wind, solar and hydro power, make up about 9% of the world's energy demand of 2018 [17]. Renewable energy production based on wind and solar sources is becoming cheaper with respect to fossil fuel power plants: in some countries it became even more advantageous to build new wind and solar power plants rather than keep running existing fossil fuel plants [26]. Despite this growing interest in renewable energy, fossil fuels still are the main source of energy required worldwide until 2018, covering the ~ 80% of the global demand. According to the World Energy Outlook of 2019 [17], following the current trends the demand would stay almost the same in 2040 (~ 78%), but by pursuing sustainable energy production it can be reduced to ~ 58%, cutting the CO₂ emissions by half.

Following current trends, energy demand is expected to increase by 1.3% per year until 2040 – while a peak of 2.3% already occurred in 2018 – and this will probably result in increased energy related-emissions [17]. The Paris Agreement committed to keeping the temperature rise below 2°C, and ideally below 1.5°C. To achieve this, hard cuts in emissions are needed, together with a marked improvement in energy efficiency. Population growth will also play a central role in the global trend of energy demands, especially because over half a billion people are expected to join the African urban population by 2040 [17]. The related expected growth in energy demand increases the importance of renewable energy sources.

In the last decades there has been an increased interest in wind harvesting thanks to the cost reductions recently achieved. Nowadays, wind source is one of the most likely energy sources to meet the energy demand for the next years. Thanks to the rapid growth of wind power generation, following today's policy ambitions, wind is set to become the first source of electricity within a decade, despite its intrinsic variability, while nuclear power and coal are both expected to decline [16]. Besides ordinary wind turbines, new promising ideas and technologies that go by the name of airborne wind energy (AWE) are being developed with the purpose of extracting energy from

high-altitude winds. These will be the subject of this work.

1.1 What is Airborne Wind Energy?

Airborne wind energy, as the name suggests, aims to find new and efficient ways to extract energy from the wind, overcoming the limitations of ground-based turbines by trying to reach higher altitudes while lowering the cost of building materials. Apart from a few articles and patents that date from a few decades ago, the greater interest in airborne wind technologies exploded after the end of the XX century. Thereafter airborne wind energy technologies have been and still are intensively studied by universities and companies, motivated by their great potential in becoming an energy source able to play its part in the challenge of responding to global needs.

The main idea at the core of airborne wind energy is to exploit airborne devices of various kind in order to reach high-altitude winds. Airborne energy systems are generally composed by one or more floating or flying devices – such as gliders, kites or buoyant turbines – connected to the ground with a cable, as showed in figure 1.1. In this set-up, power is produced by exploiting the action of the aerodynamic forces of lift and drag on the airborne devices. Power can be generated on the ground, by leveraging on the tether tension as an intermediary to transmit the aerodynamic forces to a generator, or it can be generated on board by means of a turbine, and then transmitted to the ground using a cable capable of carrying current. Those power production mechanisms will be discussed in more detail below.

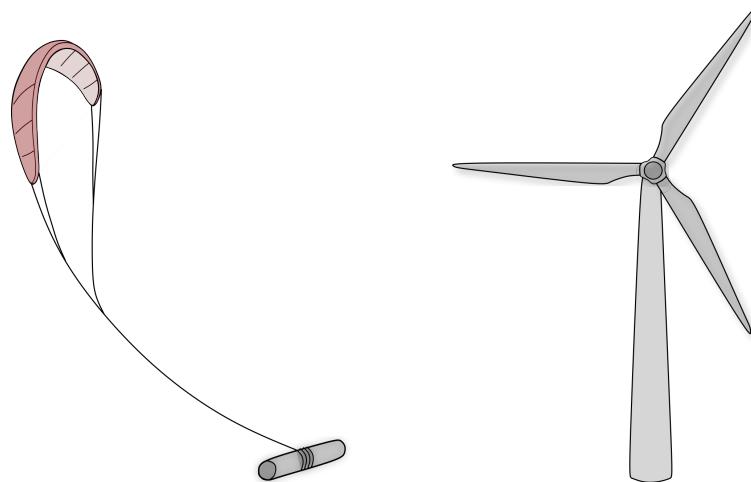


Figure 1.1: On the left, schematic representation of an airborne wind energy system composed by a tethered kite connected to a drum. On the right, schematic representation of regular wind turbine (not to scale).

1.1.1 Airborne Wind Energy advantages and open problems

There is a number of advantages in using an airborne device to harvest wind power rather than a classic wind turbine anchored to the ground [6]. First, an airborne device such as a kite or a

floating turbine can reach higher altitudes than a ground-based turbine, and higher altitudes are associated with higher wind speeds, and consequently more power to harvest [3], [2].

A measure of the amount of energy that can be harvested by a wind turbine can be given via the Wind Power Density δ , which is a function both of the modulus of the wind velocity v_w and of the air density ρ :

$$\delta = \frac{1}{2} \rho v_w^3 \quad (1.1)$$

This quantity is also useful for airborne wind energy, as it allows to take into account both the wind speed and the air density, two of the most important variables that influence the extractable power [2]. In reference [3], Archer and Caldeira calculated the trend of the wind power density profile with the height. They found out that between 500 and 2000 meters the wind power density is relatively constant, whereas above 2000 meters it increases linearly with the height. The highest values, however, are located between 8000 and 10000 meters above the ground: the wind power density at 10000 meters is on average 5 times higher than the one found at 1000 meters. Wind turbines can nowadays reach a maximum of about 220 meters. Their height increased from around 100 m in 2010 (3 MW turbine) to more than 200 m in 2016 (8 MW turbine) [17]. Today's highest and largest-capacity wind turbine is the Vestas V164, 220 meters high and capable of harvest 8 MW. Turbines of about 260 m and capable of harvesting 15-20 MW are now under development, and are expected by 2030. Unlike turbines, airborne devices permit, in principle, to reach higher altitudes with less effort since their only limitation is in the length of the cables. The maximum heights touched by the prototypes to the present day are different, but many companies have developed models of airborne devices that fly at around 300 meters of height, as the Californian start-up Makani Power [21] or the Hamburg-based company SkySails [27]. However, greater heights are being tested.

Another big advantage of airborne wind technology with respect to ordinary turbines is the saving in building materials. To increase the power generated by a turbine, indeed, it is needed to increase its height along with the size of the swept area, and since the diameter of the turbine is closely related with its cost, this causes the cost of construction to shoot through the roof. Airborne wind energy systems, on the other hand, are mostly composed of one or more lightweight flying devices connected to the ground via ropes or cables, which allow to keep construction costs lower [1].

Up to now airborne wind energy has not been widely deployed at an industrial level due to a number of open questions that still need a firm answer, even if in recent years various companies designed working prototypes which are currently on the way of commercialization. First, the optimal height for airborne devices is still under discussion [4], since greater heights are related to stronger wind speeds but also to structural problems, such as the elevation angle of the device that reduces the power output [10]. Moreover, even small wind speed fluctuations may influence the energy output [3], together with rising the need for strong automatic control. To get rid of the problems arising from energy fluctuations, Archer and Caldeira proposed the use of batteries to store the excess energy, the addition of more devices so as to cover a bigger area, or the interconnection of distant airborne stations by means of transmission lines [3]. The best building materials for airborne devices are still not known too, together with their optimal size: bigger kites, indeed, are directly linked to stronger lift and drag forces, but also to higher construction costs.

But perhaps the most important issue concerns the need for airborne devices to be strongly controlled during their motion: these systems are not intrinsically stable as ground-based turbines,

since they are usually moving or standing in ever-changing environments. Consequently the dangers of damage and destruction of the system due to falls have to be handled very carefully [1]. Automatic take off and landing are the hardest problems to solve in this context, but some companies and research groups have already developed some solutions [14], [12].

1.2 The physics of Airborne Wind Energy

This section illustrates the basic physics of airborne wind energy technology. First of all, we give an explanation about the aerodynamic forces that allow flight without the use of an engine, and the different ways in which those are used for the collection of energy. A description of the technologies that are most exploited by research groups and companies to extract power is then presented.

The total aerodynamic force acting on the airborne object can be split in two components, the *lift* and the *drag*. The lift force is defined as the component of the aerodynamic force which is perpendicular to the relative motion of the object in the fluid, while the drag is parallel and opposite to the relative motion. Thus, these forces are generated by the difference in velocity between the object and the fluid.

$$\mathbf{F}^{\text{aer}} = \mathbf{L} + \mathbf{D} \quad (1.2)$$

The drag force is mainly generated by the pressure exerted perpendicularly (pressure drag) and tangentially (friction drag) on the surface of the body by the fluid particles. The lift force is generated by the turning or deflection of the flow surrounding a body. This happens because of the shape of the body, or because of the body's movement into the flow itself. Indeed, these factors ensure that the relative velocity between the fluid and the body is locally changed and, since the fluid molecules remain in contact with the body, they generate a difference of pressure on its walls. The existence of lift force was discovered in 1878 by Lord Rayleigh, studying the flow around a circular cylinder [19].

Both lift and drag depend linearly on the fluid density ρ , the area A of the surface the force is acting on and the square of the relative velocity of the fluid moving around the body \mathbf{v}_{rel} . They depend also, in a more complicated way, on the shape of the body, the flow viscosity and the inclination between the reference line of the body and the effective velocity of the object in the flow, also called of attack angle α . This complex relationship is summarized in the lift and drag coefficients, C_L and C_D . From all the above, the expression of the lift and drag modules reads:

$$L = \frac{1}{2} \rho A C_L(\alpha) |\mathbf{v}_{\text{rel}}|^2 \quad (1.3)$$

$$D = \frac{1}{2} \rho A C_D(\alpha) |\mathbf{v}_{\text{rel}}|^2, \quad (1.4)$$

where the dependence of the lift and drag coefficients on the attack angle has been made explicit. The relative velocity \mathbf{v}_{rel} , or apparent velocity, is computed as the difference between the velocity of the flying device \mathbf{v}_k and the velocity of the wind \mathbf{v}_w :

$$\mathbf{v}_{\text{rel}} = \mathbf{v}_k - \mathbf{v}_w. \quad (1.5)$$

The dependence of the lift and drag forces on the square of the relative velocity has been one of the master key factors in the development of airborne wind energy.

Figure 1.2 gives a representation of the aerodynamic forces acting on a glider in motion, where the inclination between the relative velocity of the flow and the reference line of the aircraft is explicitly represented with the attack angle α .

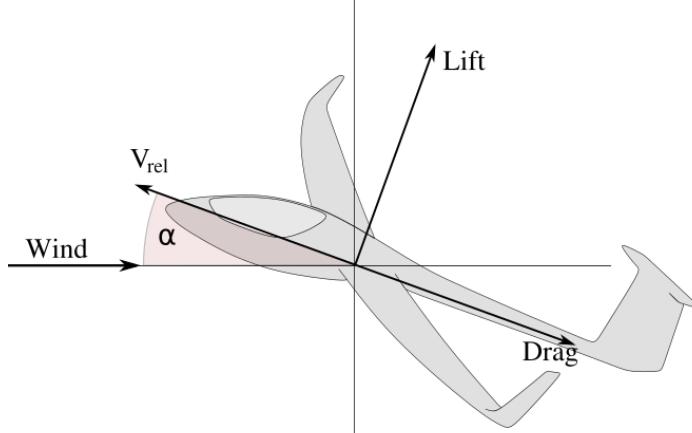


Figure 1.2: Lift and drag forces acting on a glider in motion. The inclination between the relative velocity of the flow and the reference line of the aircraft is represented by the attack angle α .

1.2.1 Power extraction

The instantaneous power that is provided by the aerodynamic forces to the airborne object moving into the fluid is defined as:

$$P_{aer} = \mathbf{v}_k \cdot \mathbf{F}^{aer} \quad (1.6)$$

Where \mathbf{v}_k is the velocity of the object. This power can be used to increase the mechanical energy of the airborne device, to pull a boat or a vehicle [29], or can be transmitted to a generator by means of a tether.

In “Crosswind Kite Power”, a pioneering work in airborne wind energy, Miles Loyd estimated the power that can be generated by a kite connected to the ground with a tether [20], developing a simplified analysis of three different models of power extraction, and estimating the maximum harvestable power from each of those. In this simplified analysis the kite is assumed as a point-mass object and characteristics such as the weight of the kite and the tether drag are neglected. Moreover, the tether is idealized as a straight and mass-less rod. The extractable power is computed assuming the kite to have constant velocity. In each of the three cases analysed, the power is computed as:

$$P = P_W A C_L F, \quad (1.7)$$

where A is the area of the wing, C_L the lift coefficient, $P_W = \frac{1}{2}\rho v_w^3$ is the power density of the wind, v_w is the module of the wind velocity, ρ the air density and F is a function which depends on the specific model analysed. Here the three models are listed briefly.

In the first model called “simple kite” power is generated by unrolling the cable from a pulley while the kite motion is collinear to the tether tension T and the tether extension velocity. In this case the power can be computed as $P = T v_k$.

The second model explains the groundbreaking concept of crosswind motion. In the ideal configuration in which a kite or glider flies crosswind, that is when its velocity is totally perpendicular with respect to the wind velocity, it happens that the total aerodynamic force acting on the device is parallel to the wind direction, as shown in figure 1.3.

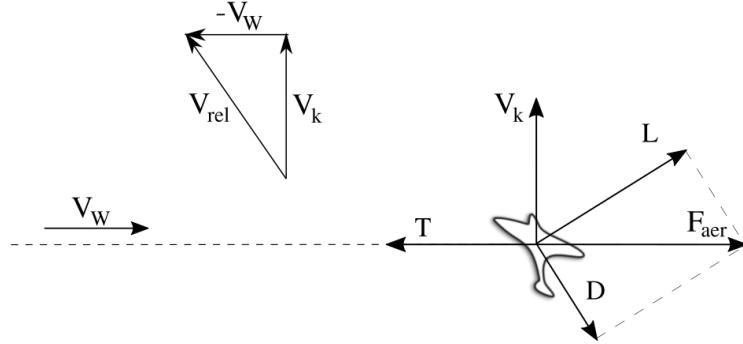


Figure 1.3: Example of a glider in crosswind motion.

In this configuration the relative velocity is larger with respect to the one of the static kite setting and therefore the amount of power that can be extracted from the aerodynamic forces is larger. As represented in figure 1.3, since in this configuration the tether and the aerodynamic forces are parallel, a large part of these forces can be turned into tether tension and thus transmitted to a generator.

In the third configuration analysed by Loyd, power is produced thanks to the presence of an on board turbine which generates additional drag. This paradigm has been called drag-mode to differentiate it from the crosswind model that uses mainly the lift force to generate tension, which is called lift-mode.

From his simplified analysis, Loyd finds out that the maximum power that can be extracted with the lift-mode equals the one of the drag-mode one. This occurs when:

$$v_k = \frac{1}{3} v_w. \quad (1.8)$$

In this configuration it turns out that:

$$F_{max} = \frac{4}{27} \left(\frac{C_L}{C_D} \right)^2 \quad (1.9)$$

Where C_L and C_D are the lift and drag coefficients of the kite. From this one can compute the maximum extractable power:

$$P = \frac{2}{27} \rho A v_w^3 C_L \left(\frac{C_L}{C_D} \right)^2 \quad (1.10)$$

The rate C_L/C_D is called gliding number, and explains how fast can a glider move horizontally rather than vertically [1]. Loyd showed that this power is bigger of a factor $\frac{1}{2}(C_L/C_D)^2$ than the maximum power that can be extracted from a “simple kite” system, namely a kite which is not flying in crosswind mode.

1.2.2 Refinements to power extraction

The work by Loyd may be considered as an upper bound for power extraction because of the simplifications that are made. One of these is the crosswind motion itself: unfortunately, it is hard to have a kite flying totally crosswind since it is usually attached to the ground with the tether.

Diehl refined the work of Loyd providing a better estimate of the upper bound to the power that can be extracted from the wind using an airborne wind energy system [1]. Diehl says that given a constant wind field with speed v_w , the maximum amount of power that can be extracted from it by a flying object is the scalar product of the velocity of the wind times the aerodynamic force to which the object is subjected:

$$P_w = v_w F^{aer} \cos(\chi) \quad (1.11)$$

Where χ is the angle between the wind velocity and the aerodynamic force. This means that whenever the aerodynamic force is not parallel to the wind direction, there is a loss in the extractable power. These have been called cosine losses, and are mostly due to two factors: the inclination needed by the tethered kite to reach a certain altitude, and the weight of the object itself, which causes the aerodynamic force to tilt even more with respect to the direction of the wind field. This second contribution, however, is usually quite small.

Given these considerations Diehl found the following limit for the extractable power [1]:

$$P_{max} = \frac{2}{27} \rho A v_w^3 C_R \left(\frac{C_R}{C_D} \right)^2, \quad (1.12)$$

where $C_R = \sqrt{C_L^2 + C_D^2}$.

An example of a crosswind motion limited by the cosine losses can be a kite or a glider which moves circularly or in a figure-eight trajectory on the surface of a sphere created by a fixed length tether, as shown in figure 1.4. Other sources that address the issues of energy and efficiency estimation of airborne wind energy systems can be found in references [11] and [5].

1.3 Airborne Wind Energy technologies and devices

Over the years a number of approaches and technologies have been used to study airborne wind energy systems, and many models have been developed [22], [18], [30], [8]. Companies and research groups around the world devised various ways to exploit the aerodynamic forces which consequently lead to implement very different prototypes of energy collection systems. In [10], Cherubini et al.

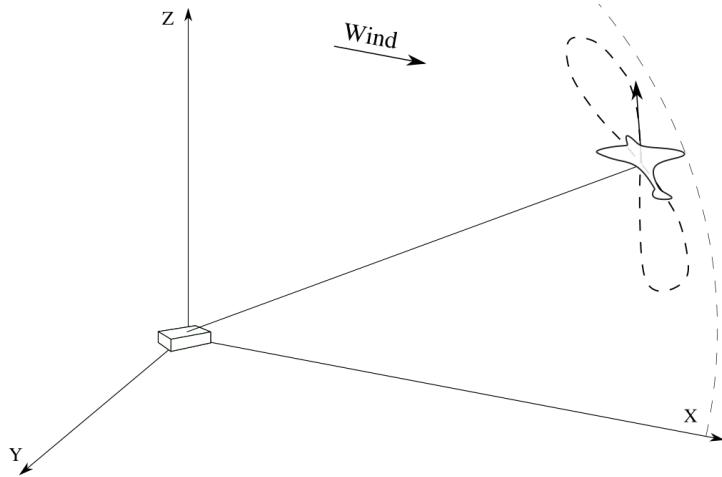


Figure 1.4: Example of a crosswind motion limited by the presence of a tether connected to the ground. Here a glider is moving with a figure-eight trajectory on the surface of the sphere defined by the fixed tether length.

gave a comprehensive survey of the technologies developed into airborne wind energy, which have been later summarized in the book “Airborne Wind Energy” by Ahrens et. al [1]. The differences between the realized airborne wind energy systems arise at many levels, ranging from the type and number of airborne object used, to the way in which wind energy is collected and consequently the exploited aerodynamic forces, to the site where energy transformation happens, and finally how the system is controlled in order to ensure a safe trajectory.

This section, which is based on references [10] and [1], presents a simplified review of the main airborne systems developed over the years, together with the physical concepts they are based on.

1.3.1 Power generation

Generally, airborne wind energy systems are made of two major parts: a ground based component and one or more flying objects, which are mechanically and sometimes electrically connected with tethers. In some systems power is generated on-board, in others on the ground, while a third system uses the aerodynamic forces for vehicle propulsion.

In the first power generation system, called on-board generation, the electrical generation happens on the aircraft-like object by means of an on-board turbine added to the system. Exploiting this mechanism there is the need to carry current from the object to the ground, so usually special ropes with integrated electrical cables are used. This kind of system is also called drag-mode, a name coined by Loyd, because the turbine adds some extra drag to the flying element. An useful feature of this system is that it allows to simplify the take-off and landing processes – which in other systems present control difficulties – since both can happen vertically using the generator as an engine. The company Makani Power developed its on-board generation systems by applying turbines on a tethered aircraft and exploiting crosswind motion for power production.

The second power generation system, called ground-based generation, generates the electrical energy on the ground by exploiting the tether tension. Therefore, in this system the cable must

only bear the tension induced by the aerodynamic forces. Among mechanisms of this type we find those that have an energy station fixed to the ground and those with a moving station placed above a vehicle or a ship.

The ones with a fixed ground station often exploit the tether tension by unwinding the cable from a coil through mechanical moment: the rotation of the coil is in this case responsible for the production of energy, which happens during the so called reel-out phase. The reel-out phase can happen with the kite traveling crosswind, as in the case of a figure-eight or circular trajectories, but also in a non-crosswind set-up where the airborne object is raised by the lift force while in a stationary position. Since the cable length is limited, the reel-out phase must be followed by a retraction phase called reel-in, in which a portion of the generated energy is used to rewind the cable on the coil moving the airborne object to its closest point to the ground station until the portion of unrolled cable reaches the minimum length (see figure 1.5). This task is done by changing the flight mode of the kite so as it produces less lift force; in this way the cable can be rewound wasting as little energy as possible. The full sequence then restarts again.

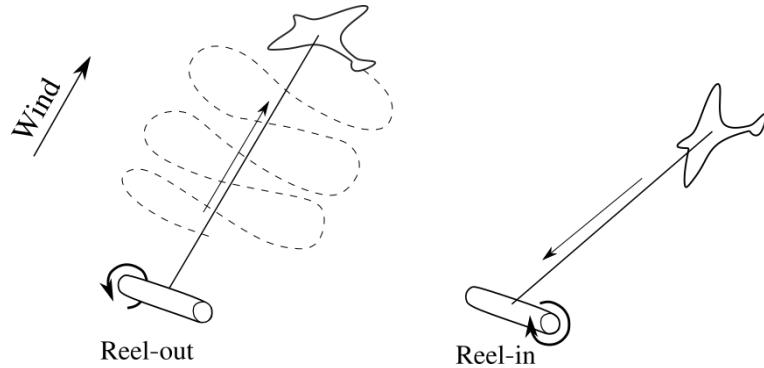


Figure 1.5: Reel-in and reel-out phases.

For systems in which ground-based generation happens while flying in crosswind motion, the term lift-mode, coined by Loyd, is used.

The ground-generation mechanisms which include a moving ground station are usually more complex, and some of them are designed in a way to have an always-positive amount of energy production. The advantage of these systems is that, having a continuous flow of positive power, they can be easily connected to electrical networks [10]. Two kinds of ground generation systems have been designed. The first presents one or more aircraft tied with ropes on a vertical axis fixed on a rotating base by means of a rotor [8], [7]. In this system the aircraft's motion is controlled in a way to turn the rotor which transmits mechanical moment to a generator. The second system uses horizontal rails and integrates the energy generating system on vehicles bounded to move along the rails, as it is schematically represented in figure 1.6. In this setting energy is generated by the moving vehicle. This kind of airborne system can be used not only to produce energy, but also to obtain vehicle propulsion. The SkySails company [27] exploited this mechanism deploying automatically controlled kites to achieve ship propulsion – as well as energy generation – with the goal of reducing fuel consumption and the related emissions. This is the first company that managed to use kites for an industrial application. In the present work a simplified version of this type of systems has been studied, where a kite is used to pull a block bounded to slide inside a rail,

with the goal of reaching the maximum velocity.

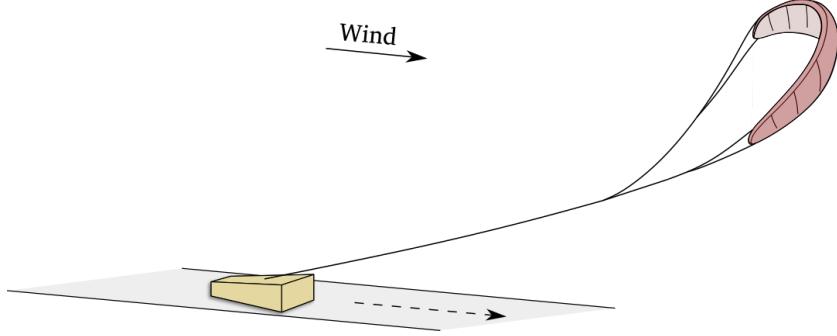


Figure 1.6: Example of a kite which exploits the aerodynamic forces to pull a block bounded to slide along a rail.

Some companies and research groups have gone beyond the simple idea of a single airborne object connected to the ground by a cable, proposing a connection of multiple devices to collect wind energy in a single system [8], [24], [25], [31].

1.3.2 Airborne object

Another subdivision between airborne technologies concerns the shape and the weight of the airborne object. Those characteristics are strongly linked to the purpose of the system and the physics behind power generation, which includes the mechanism adopted for the power conversion and the aerodynamic forces exploited.

Regarding the flying device, a main distinction between flexible and rigid wings can be done. Flexible wings are mostly lightweight, as leading-edge inflatable kites used for kite-surfing [10]. These keep their shape only thanks to the constant action of aerodynamic forces on them. Their characteristic lightness ensures that they not pose a risk for people, so they can even be used near populated areas, and for the same reason they are not easily destroyed by a fall. Rigid wings and gliders, on the other hand, are usually heavier and consequently can reach higher relative velocities. This means that stronger lift forces can be exploited and more power can be extracted. Both configurations need control in order to be exploited safely, however, because of the higher weights and speeds, rigid-wings items may represent a danger and therefore need stronger and safer control methods to avoid the destruction of the device and safety issues.

Some other systems do not rely on the presence of wings because their functioning is based on aerostatic lift rather than aerodynamic lift. These devices can float without consuming power even with little or no wind, because their fluctuating part is lighter than air. An example of this is the Buoyant Air Turbine (BAT) developed by Altaeros energies, which is schematically represented in figure 1.7, composed by a main giant volume filled by helium [23].

1.4 Control techniques

In order to harvest wind energy it is necessary to deal with the intrinsic instability of airborne wind energy systems, given that gliders and kites are located in an environment in continuous evolution.

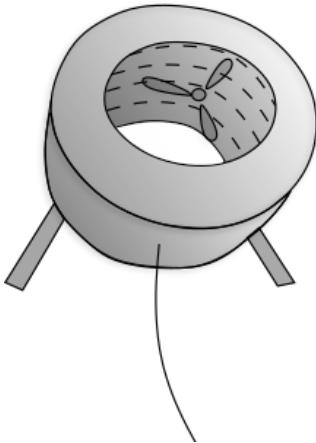


Figure 1.7: Schematic representation of the Buoyant Air Turbine (BAT) developed by Altaeros energies.

Moreover, as previously discussed, not all the airborne systems must remain static into the wind but most of them have an ideal trajectory to follow: an example are kites used for vehicle propulsion, that move in the so-called open-loop trajectory, or kites with fixed ground station moving in a closed-loop. Since the trajectory followed by the airborne device depends on the wind direction and intensity, two types of instruments are needed to control it. The first are sensors that allow to measure the physical quantities related to the environmental condition that can affect the aircraft trajectory, such as the speed and direction of the wind and the aircraft orientation. The seconds are actuators, which allow to modify some of the device control parameters influencing the trajectory and ensuring a safe and effective path.

The trajectory of flight can be monitored and controlled in different ways, depending on the airborne object employed and its desired flight path. Examples of such devices are on-board anemometers which can be used to assess the wind velocity and direction, while additional ropes or actuators can be used to control the angles with which the wing faces the wind, and therefore the aerodynamic force applied to it. In the case of pumping mode, the system can be controlled by regulating the speed of the rope that unwinds from the drum, and consequently the tension force.

The control itself can be carried out manually by a human operator or it can be automated through the use of numerical control systems. This option is certainly the most interesting, and several research groups are working to find the best way to make the task of control completely automatic.

1.4.1 Numerical control of the trajectory

In order to understand the best way to control an airborne wind energy apparatus to achieve the targets set, numerical simulation is used. In a computational simulation, the problem of controlling nonlinear and time-varying dynamics is addressed by optimal control theory, which is a particular case of reinforcement learning. Computational control is exploited in the field of airborne wind energy in order to find the best values for some parameters of the system – that represent the

actuators – in order to achieve the main objectives to which these technologies aim: maximize the extracted power or maximize the road travelled.

The problem of optimal control of airborne systems has been addressed by several authors over the years [9], [12]. Houska and Diehl in reference [15], for example, studied the problem of optimizing the trajectory of a kite moving crosswind in periodic eight-figure in order to maximize power production. In reference [29], Williams et al. studied the model of a kite tied with a tether in two different settings: in the first the kite was employed in crosswind motion to pull a vehicle with the aim of maximizing the distance covered, while in the second the goal was to generate power by connecting the kite to a ground-based generator. In both these works the kite was controlled by manipulating the roll and attack angles.

To address the vehicle towing issue, Williams et al. solved a periodic optimal control problem, where the inputs were the control angles derivatives and the acceleration of cable elongation, while the cost function aimed at maximizing the amount of road traveled by the vehicle while keeping at minimum the control effort. A problem linked to the vehicle towing will be covered in this thesis work, but the control mechanism that we use has not yet been employed to solve control problems for airborne wind energy systems so far. In this work the issue of trajectory control has been addressed through reinforcement learning, whose basic concepts and guidelines will be introduced in the next chapter.

Chapter 2

A quick introduction to Reinforcement Learning

In this work we address the problem of optimizing the flight of a kite connected to a block, with the goal of maximizing the speed of the block. This problem can be formulated as an optimal control problem, where the task is to maximize a cost function or a signal that depends on some variables of the dynamics. Reinforcement learning is one of the possible approaches to face optimal control problems. In this chapter we cover the basics of reinforcement learning. First we provide an intuitive explanation of what reinforcement learning is, together with its classification among other machine learning algorithms. Next we move to its core concepts and its mathematical formulation, walking through all the notions needed to understand the procedures used in Chapter 4 to perform the learning processes. This chapter is based on Sutton and Barto’s “Introduction to Reinforcement Learning” [28].

2.1 Introduction

The core idea of reinforcement learning is to learn how to behave in every situation in order to reach a stated goal. This target is usually translated into the form of a numerical signal. In reinforcement learning the decision-maker of the process – called agent – has the task of understanding which actions to take in each situation to maximize this signal over time, with the particularity that the actions the agent takes at a certain time can affect his entire future. Furthermore, the agent is never told what is best to do to achieve the goal, but has to discover it through trial-and-error.

The fact that the agent’s actions influence its future together with the trial and error learning procedure are the central characteristics of reinforcement learning, which differentiates from the other two branches of machine learning, namely supervised and unsupervised learning. Supervised learning, indeed, aims to realise a model extracting knowledge from a set of labeled data, and therefore bases its operation on some preexisting knowledge. The final goal of supervised learning it is to comprehend at best new unlabeled data when presented. This framework can be imagined as an omniscient teacher who is handing to his student all the knowledge he needs. Unsupervised learning, on the other hand, has the goal of discovering hidden patterns in the data without the use of preexisting knowledge. In this setting there is still a dataset where to learn from, but there

is no teacher, that is, no labels from which to extract knowledge: the learner has to work on it autonomously. Reinforcement learning differentiates from both those paradigms because of its objective, namely to maximize a numerical signal, and its learning framework, where no instruction is provided to the learning agent.

2.2 Finite Markov Decision Process

The reinforcement learning framework can be formalised as a finite Markov decision process (MDP), a mathematical framework widely used to describe sequential decision-making processes that presents stochasticity. Markov decision processes have been introduced around 1950 by Richard Bellmann as an extension of Markov Chains. In this framework the main characters of the decision-making process are the agent – the decision-maker – and the environment, which represents everything he interacts with. The decision-making process is discretized into a list of subsequent time steps $t = 0, 1, 2, \dots$. At each time step the agent knows the state $S_t \in \mathcal{S}$ in which stands and takes an action $A_t \in \mathcal{A}(S_t)$ on the basis of this knowledge. Taking this action the agent interacts with the environment, which modifies the agent's state in the next time step S_{t+1} , and gives to the agent a numerical reward $R_{t+1} \in \mathcal{R} \subset \mathbb{R}$. This one-step process of interaction between the agent and the environment is schematized in figure 2.1. The process then repeats giving rise to a sequence of states, actions and rewards until a terminal condition is reached. The term “finite” in the Markov decision process comes from the fact that states, actions and rewards all belongs to finite size sets. Moreover, it is important to notice that the above defined time intervals do not refer to real intervals of time, but rather to sequential steps of choice of actions, and can be totally independent of time.

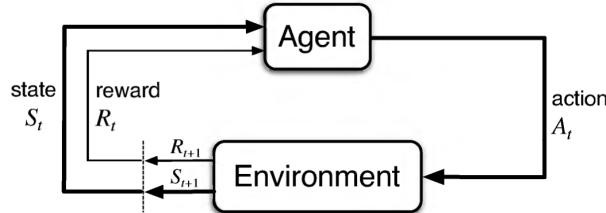


Figure 2.1: The agent-environment interaction in a Markov decision process [28].

Markov decision processes are stochastic processes. This means that, given a state and an action, the next state and the reward are not necessarily deterministic: there is indeed a discrete probability distribution for the random variables R_{t+1} and S_{t+1} given the current state S_t and the selected action A_t . This probability distribution is defined by the dynamic function p , which maps each state action couple to the following state and reward:

$$p(s', r | s, a) = P(S_{t+1} = s', R_{t+1} = r | S_t = s, A_t = a) \quad (2.1)$$

As can be noticed the probability of obtaining a state s' and reward r depends only on the previously visited state and on the action selected in that state. This peculiarity is called Markovian or memoryless property of a stochastic decision-making process, and it is necessary for the correct

formal definition of the same. All this, however, must not make one think that the agent is limited in his actions, since the information needed to perform a move must always be specified as observables belonging to the state itself.

As anticipated, in this framework the target of the agent is modeled via a signal called reward, which is provided to the agent by the environment at each time step as a number $r \in \mathbb{R}$. The agent's goal is therefore to maximize the total reward cumulated in the long run. Since the sequence of rewards the agent will get during the trajectory is not known at the beginning, one can maximize the return that the agent expects to get from the current time step t on. The expected return G_t is defined as the sum of future discounted rewards:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \quad (2.2)$$

Where γ is called discount rate, and is defined so that $0 \leq \gamma < 1$. One can immediately notice that the infinite sum in 2.2 has a finite value if $\gamma < 1$. The utility of γ is to value differently a reward obtained at the present time instant ($k = 0$) with respect to one obtained k time steps in the future: the latter is weighted γ^{k-1} times less than if it was received at the present time instant. The value of γ changes the way the agent looks at long-time rewards. If $\gamma = 0$, indeed, the agent cares only about maximizing the reward received immediately after the action, while, as γ increases, it gives more and more values to rewards obtained in the future.

Returns at different time steps can be linked via the following relation:

$$G_t = R_{t+1} + \gamma G_{t+1} \quad (2.3)$$

Understand how to make choices is the central task of the agent. The actions the agent selects at each time step, in fact, have an impact along its entire future path, that is, all the states and the rewards that he will encounter. In the Markov decision process the agent's behavior in each state is totally defined by the policy π , a function which maps states to actions. The simplest policy is the deterministic one that associates to each state a single possible action, but a more general policy can be a stochastic one that assigns to each available action in each state a probability of being selected. The probability of selecting the action a in the state s is denoted by $\pi(a|s)$. Since a policy is a probability distribution, the following conditions apply:

$$\begin{cases} \pi(a|s) \geq 0 \\ \sum_{a \in A} \pi(a|s) = 1 \end{cases}$$

Given this definition of policy as a strategy adopted by the agent to behave in the environment, the goal of the learning process can be reformulated as finding the best policy for a specific task, also called optimal policy, which is the one that allows to get the maximum possible return in time. This is the task one needs to complete in order to solve the problem of optimal control in the formulation of reinforcement learning.

2.3 Value Functions and Bellman Equations

In order to choose the best policy for a given problem, we need to measure the effectiveness of one policy compared to another: the value functions serve this purpose. A value function gives a

measure of the goodness of a state on the basis of the return the agent expects to gather in his future if starting from that state and acting according to a given policy. Formally, the value function of the state s under the policy π is defined as the expected return if the agent starts from that state and moves following policy π :

$$v_\pi(s) = E_\pi[G_t | S_t = s], \quad \text{for all } s \in S. \quad (2.4)$$

Similarly one can define the action-value function, which is the return expected by the agent when it starts from the state s , performs the action a , and follows the policy π thereafter:

$$q_\pi(s, a) = E_\pi[G_t | S_t = s, A_t = a] \quad (2.5)$$

Those functions give a summary of how good a certain policy is by means of an expectation of the return obtained by averaging on all the paths that the agent could take if following a certain policy. Using the given definition of value functions together with relation 2.3 and writing explicitly the expectation one can work out an important recursive relation between the value of a state and the value of all its possible successor states:

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)(r + \gamma E[G_{t+1} | S_{t+1} = s']) \quad (2.6)$$

$$= \sum_a \pi(a|s) \sum_{s', r} p(s', r|s, a)(r + \gamma v_\pi(s')) \quad (2.7)$$

The same can be done for the action-value function:

$$q_\pi(s, a) = \sum_{s'} \sum_r p(s', r|s, a)(r + \gamma E[G_{t+1} | S_{t+1} = s']) \quad (2.8)$$

$$= \sum_{s', r} p(s', r|s, a)(r + \gamma \sum_{a'} \pi(a'|s') q(s', a')) \quad (2.9)$$

Those results are called Bellman equations and can be used to compute the value of each state or state-action couple of the Markov decision process, once fixed the policy, the value of the discount factor and the dynamics function. Indeed, the application of the Bellman equation to a finite Markov decision process produces a system of linear equations with a unique solution. The solution of the system can be worked out simply for small-sized problems, but becomes harder when the number of states and actions increases. In these cases different algorithms are needed, with the ability to deal with more complex and larger problems.

2.4 Optimality

As already mentioned, to behave at best the agent needs to find an optimal policy. In a formal definition, a policy π is optimal if $v_\pi(s) \geq v_{\pi'}(s)$ for each $s \in S$, for each π' . In this case one uses the expression $\pi \geq \pi'$, or π is better or equal to π' . In all finite MDP there is always at least

one deterministic policy that is as good as or better than all the other ones. This can be easily understood: if the policy π_1 is the best in most of the states, but another policy π_2 does better than π_1 in a small and different subset of states, all that the agent needs to do is to follow π_1 in certain states and π_2 in others, using the best one in each case. In this way the agent is actually using a new policy π_3 composed by the two starting policies, which is as good as or better than the other two. Sometimes there may be more optimal policies, and in this case the value function computed for the different optimal policies in each state must be the same.

It is important to notice that the measure of the goodness of a policy is given by the value function computed under that specific policy, but the value function does depend also on other parameters, among which the discount factor γ . The value of γ , indeed, can heavily change the result of the value function computed in each state for a fixed policy.

Since the optimal policy $v_*(s)$ is the one with the highest value function, one can write:

$$v_*(s) = \max_{\pi} v_{\pi}(s) \text{ for each } s \in S, \quad (2.10)$$

and

$$q_*(s, a) = \max_{\pi} q_{\pi}(s, a) \text{ for each } s \in S \text{ and } a \in A(s). \quad (2.11)$$

These results allows to write the version of the Bellman equation for the optimal value functions. Indeed, since the optimal policy is the deterministic one that chooses in each state the action that maximizes the future reward, one can found that:

$$v_*(s) = \max_a \sum_{s', r} p(s', r | s, a) (r + \gamma v_*(s')) \quad (2.12)$$

$$q_*(s, a) = \sum_{s', r} p(s', r | s, a) (r + \gamma \max_{a'} q_*(s', a')) \quad (2.13)$$

Those are called Bellman optimality equations. If the dynamic function $p(s', r | s, a)$ is provided, the Bellman equations can be used to find out the optimal policy. If the equation for $v_*(s)$ is provided, one can find the optimal policy $\pi_*(s)$ by computing the right hand side of the respective Bellman equation for each possible action, and then searching for the maximum value.

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s', r} p(s', r | s, a) (r + \gamma v_*(s')) \quad (2.14)$$

By using $q_*(s, a)$, computing the optimal policy $\pi_*(s|a)$ is even easier because it is not necessary to search for the best action in each state:

$$\pi_*(s|a) = \operatorname{argmax}_{a'} q_*(s', a') \quad (2.15)$$

However, even if the dynamic function is available, it is extremely expensive from a computational point of view to find the optimal policy using only the Bellmann optimality equations. On the other hand, if the dynamic function is not available, these equations cannot be used, and it is necessary to find a way to approximate them. In the next two sections we describe a method widely used to approximate the Bellmann optimality equations, on which the algorithm that was used in the learning phase of this work is based.

2.5 Generalised Policy Iteration

To reach the objective of learning the optimal policy the agent needs to improve his choices. The task of refining a policy is formally called policy improvement or control. Since to improve a policy one needs a way to asses how good or bad it is, often to hit the target of control it is necessary to evaluate a given policy by computing the state value function or state-action value function. This task is called policy evaluation or prediction.

The optimal policy can be found by using an iterative process where evaluation and improvement are repeatedly applied to a starting policy. Given a policy, indeed, it is quite easy to find a better one by acting greedily with respect to the value function or the state-action value function of the starting policy, namely using a deterministic policy where the only available action in each state is the one for which the value function finds its maximum. Repeating this procedure, one would get a sequence of improved policies, until both the policy and its value do not change anymore. This process is summarized under the name of policy iteration, and can be represented as a bouncing between a moment in which an accurate value function is found, and one in which one acts greedily with respect to it to obtain a better policy. It is not necessary for one of these two processes to be completed in order to start the next one: computing a partial policy evaluation and a partial policy improvement will still lead to optimality. This generalized framework is called Generalised Policy Iteration. A lot of reinforcement learning methods are based on this description, included the one that has been used in this work.

2.6 Temporal-Difference Learning

In this section we finally describe the learning algorithm used in this thesis project, bringing together all the concepts developed within the chapter. Temporal-difference learning is one of the core ideas of reinforcement learning, and it is a framework widely applied to solve learning problems when the dynamic function is not available. For this reason, this learning technique has been used in the present work. Temporal difference learning allows to obtain an estimate of the value function and, if used into the Generalised Policy Iteration setting, can be exploited to perform policy improvement with the aim of finding the optimal policy.

The strength behind temporal difference learning is given by two important factors:

- it allows to learn directly by interacting with the environment without the need to know the dynamics function p ,
- it updates the estimates for each element of the value function by using the other estimates, namely it bootstraps from the current estimated values.

In order to use temporal difference learning together with Generalized Policy Iteration to find improved policies it is easier to work with the estimate of the state-action value function, since in this case the updates to the estimate are performed each time an action is taken. We refer to the estimate of the state-action value function for a couple made by the state S_t and the action A_t with $Q(S_t, A_t)$. To update the estimate one can write:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \eta[G_t - Q(S_t, A_t)], \quad (2.16)$$

which means that the current estimated value $Q(S_t, A_t)$ is being corrected by the amount $G_t - Q(S_t, A_t)$. η is the learning rate, and is defined so that $0 \leq \eta < 1$.

Recalling that $G_t = R_{t+1} + \gamma G_{t+1}$, equation 2.16 becomes:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \eta[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)]. \quad (2.17)$$

In these expressions the quantity inside the square bracket is called temporal difference error or TD-error, while $R_{t+1} + \gamma Q(S_{t+1}, A_{t+1})$ is called the TD-target, since when $Q(S_t, A_t)$ reaches this value it does not change anymore. This update is performed each time the agent moves in a non terminal state. If S_{t+1} is a terminal state, one can define $Q(S_{t+1}, A_{t+1}) = 0$. At this point it is easy to have a complete control algorithm, since to perform policy improvement one can just change the policy greedily with respect to Q , namely by choosing the action which leads to its maximum value.

All of this together makes up the SARSA algorithm for control. The sequence of data needed to perform one step of the algorithm gives rise to its name: to perform one step the SARSA algorithm needs the current state S_t , the current action A_t , the reward gained from the environment R_{t+1} , the next state S_{t+1} , and the next action A_{t+1} . Figure 2.2 represent schematically this process.

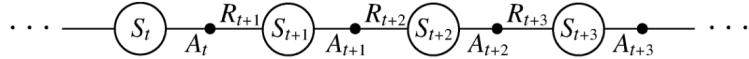


Figure 2.2: State-action-reward schema (Sutton and Barto [28]).

SARSA is an on-policy algorithm, meaning that it aims to improve the same policy that it uses to make decisions. In on-policy methods usually the employed policy is "soft", meaning that $\pi(a|s) > 0, \forall s \in S, \forall a \in A(s)$, but while proceeding with the learning this policy must move closer to the optimal deterministic one. In order to do this the ϵ -greedy policy often is used, where the agent the most of the times acts by choosing the action which leads to the best reward, the so-called greedy action, but with probability ϵ takes random actions. By varying the amount of randomness ϵ during the learning one can slowly move the policy to the deterministic one, where $\epsilon = 0$.

The complete algorithm for SARSA is written below in pseudo-code:

Algorithm 1: Sarsa algorithm [28]

Algorithm parameters: learning rate $\eta \in (0, 1]$, small $\epsilon > 0$

Initialize $Q(s, a)$, for all s, a except $Q(\text{terminal}, .) = 0$

while episode **do**

 Initialize s

 Choose a with ϵ -greedy policy

while step **do**

 Take action a , get r', s'

 Choose a' with ϵ -greedy policy

$Q(s, a) \leftarrow Q(s, a) + \eta[r' + \gamma Q(s', a') - Q(s, a)]$

$s \leftarrow s'$

$a \leftarrow a'$

end

end

This algorithm is used in Chapter 4 to perform the learning process of the modeled airborne wind energy system to find the nominally optimal policy for the kite's control system, with the aim of pulling the block reaching the maximum possible speed.

Part II

Modeling and learning

Chapter 3

Mathematical models

In this chapter we describe the mathematical model of the system and, given its complexity, we develop a simplified two-dimensional model before tackling the more realistic three-dimensional one. In both cases the system consists of a block connected to a kite through a straight cable of fixed length. The kite exploits the aerodynamic forces to pull the cable and move the block, which is fixed on a rail on the ground and can only move along it.

The two-dimensional model has been simplified considering the existence of a single horizontal dimension identified by x , and a vertical dimension perpendicular to it, identified by z . In this setting the rail along which the block moves coincides with the x axis, while the kite is free to move on the circumference defined by the fixed rope length on the x/z plane. This model has been described both in Cartesian and in polar coordinates.

The three-dimensional model has been developed only in Cartesian coordinates for the simplicity of that depiction. In this setting we used the right handed system consisting of the x , y and z axes, where x and y represent the horizontal plane. Here the block can move only along the rail aligned with the x axis, while the kite is able to travel on the whole $z > 0$ half of the surface of the sphere drawn by the cable.

3.1 General characteristics of the models

The model of the system has been simplified by means of physical approximations for the purpose of analyzing its general functioning and to apply the learning algorithm on it. Both the kite and the block in this framework are modeled as mass points and are bounded to each other by means of an inextensible tether with fixed length, which sets the reciprocal distance of the two during their motion. This is an approximation, since in reality the tether would slightly bend because of the action of the gravity and the drag on it, shortening the distance between the two objects.

The orientation of aircrafts and kites in space is usually controlled by turning the airborne object in the three dimensions with respect to its center of gravity, which is the average location of its mass. From this point one can define a triad of axes perpendicular to each other, about which the aerodynamic object can be turned. These are known as principal axes and are represented for a kite in figure 3.1. We will refer to the axis going from the left to the right side of the kite's body as the transverse axis e_t ; this is the axis around which the kite should be turned in order to change the attack angle α (or pitch angle), performing an up or down movement with the “nose” of the object.

The axis going from the tail to the head of the kite is called longitudinal axis e_l , and turning on this axis allows to vary the bank angle ψ (or roll) by moving up or down the tips of the wing. In figure 3.2 the effect of rotating the kite around the longitudinal axis e_l is represented. The third axis goes from the bottom to the top of the kite orthogonally to the other two, and turning on this axis allows to vary the yaw angle by moving the “nose” of the object to the left or to the right [13].

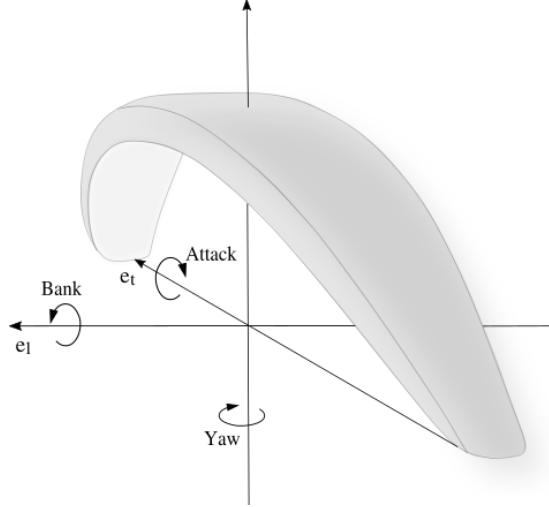


Figure 3.1: Representation of the control angles of attack, bank and yaw as rotations around the three principal axes of the kite.

In the presented models we always assume that the longitudinal axis coincides with the opposite of the relative velocity axis when the attack angle of the kite is set to $\alpha = 0$, while it is tilted with an angle α with respect to it when $\alpha \neq 0$. This means that the transverse axis e_t is always perpendicular to the relative velocity of the wind [18], therefore the yaw angle is always set to zero. This concept is represented in figure 3.3.

In the two-dimensional model we allow the kite controller to vary only the attack angle α , while the roll angle ψ is not included in this description, meaning that its value is always set to zero. Figure 3.3 provides both a depiction of how the two-dimensional model of the kite should be thought in this setting, and a projection of the three-dimensional model on the two-dimensional x/z plane. On the other hand, in the three-dimensional model both angles can be changed. We assume that both the attack and the bank angle can be manipulated using the ropes fixed on the sides of the kite, as suggested by Houska in [18].

The information about the shape and the aerodynamic features of the kite is all contained in the dimensionless lift and drag coefficients C_L and C_D , whose values have to be measured experimentally for a given shape and usually depend strictly on the attack angle. In this work the values of the lift and drag coefficients for a given angle of attack were taken from reference [7], and refer to a rigid curved kite wing with an airfoil profile of Clark-Y type, which is a commonly used profile designed in 1922. The Clark-Y kite profile is schematically represented in figures 3.4 and 3.5.

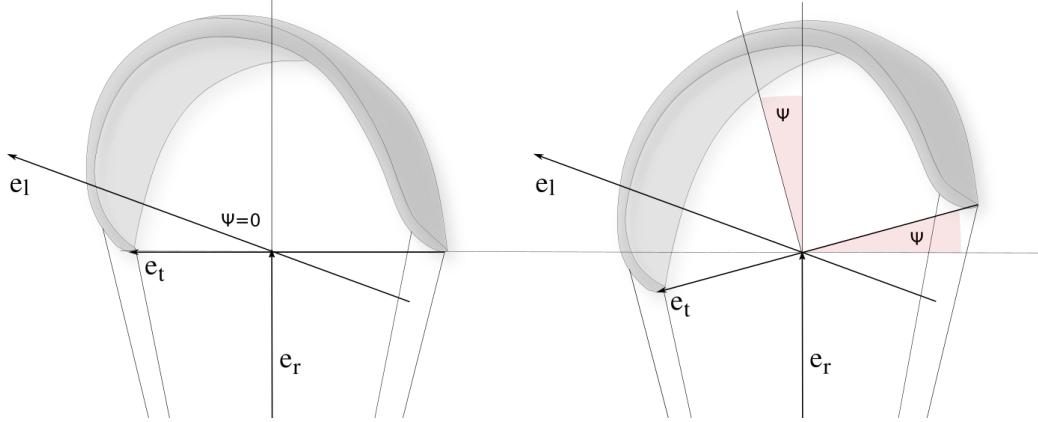


Figure 3.2: Rotation of a kite about the longitudinal axis e_l , which goes from the tail to the head of the kite. Turning on this axis allows to change the bank angle ψ . The control over the bank and the attack angles can be kept by pulling the ropes on the sides of the kite.

3.2 Lift and Drag

To develop a complete dynamic of the models it is necessary to compute the lift and drag forces on the kite, and their components on the axes. Since to work out the lift force it is necessary to perform some vector products, the three-dimensional computation of the aerodynamic forces will be carried out first, and only afterwards the simplifications for the two-dimensional model will be drawn. The lift computation is based both on references [18] and [29].

As anticipated, the modulus of the lift and drag forces depends on the difference between the speed of the kite \mathbf{v}_k and the wind speed \mathbf{v}_w , namely the relative velocity $\mathbf{v}_{rel} = \mathbf{v}_k - \mathbf{v}_w$. The module and direction of the drag are relatively easy to find since its orientation is equal and opposite with respect to the relative velocity vector. Once computed the relative velocity, the expression for the drag force reads:

$$\mathbf{D} = -\frac{1}{2}\rho A C_D(\alpha) |\mathbf{v}_{rel}| \mathbf{v}_{rel}, \quad (3.1)$$

where ρ is the density of the air, A is the proportion of the kite surface perpendicular to the wind and $C_D(\alpha)$ is value of the drag coefficient that depends on the attack angle. The drag components on the axes can be computed by decomposing the relative velocity vector on the axes:

$$D_i = -\frac{1}{2}\rho A C_D(\alpha) |\mathbf{v}_{rel}| v_{rel,i}, \quad (3.2)$$

where i is the index of the axis.

Working out the components of the lift force is a bit more tricky. The lift is oriented perpendicularly with respect to the relative velocity vector, and tilted by the roll angle ψ with respect to the perpendicular to the plane formed by the head-tail and right-left wing axes of the kite. In reference [29] the lift direction is computed by searching first the plane containing both the drag force and the velocity vector:

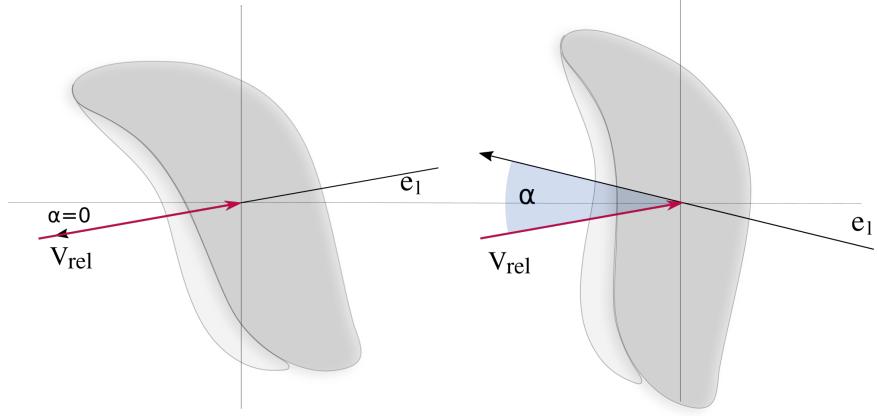


Figure 3.3: Representation of a kite with view from the side. In the left figure the attack angle is set to zero, while in the right figure $\alpha > 0$. The attack angle represents the inclination between the kite longitudinal axis e_l and the relative wind direction.

$$\mathbf{t}_2 = \frac{\mathbf{e}_r \times \mathbf{v}_{rel}}{|\mathbf{e}_r \times \mathbf{v}_{rel}|}, \quad (3.3)$$

where \mathbf{t}_2 is the unit vector perpendicular to that plane, and \mathbf{e}_r is the vector identifying the kite position. Then one has to compute the vector parallel to the lift force when the roll angle is zero:

$$\mathbf{t}_3 = \frac{\mathbf{v}_{rel} \times \mathbf{t}_2}{|\mathbf{v}_{rel} \times \mathbf{t}_2|}. \quad (3.4)$$

Because of its perpendicularity with respect to the relative velocity, the lift finds itself on the plane generated by the unit vectors \mathbf{t}_2 and \mathbf{t}_3 , which are both perpendicular to \mathbf{v}_{rel} . Therefore one can write:

$$\mathbf{L} = \frac{1}{2} \rho A C_L(\alpha) |\mathbf{v}_{rel}|^2 (\mathbf{t}_2 \sin(\psi) + \mathbf{t}_3 \cos(\psi)), \quad (3.5)$$

where ψ is the roll or bank angle. To get the components on the axes we just need to express \mathbf{t}_2 and \mathbf{t}_3 on the basis of them:

$$L_i = \frac{1}{2} \rho A C_L(\alpha) |\mathbf{v}_{rel}|^2 (t_{2,i} \sin(\psi) + t_{3,i} \cos(\psi)), \quad (3.6)$$

where i is the index of the axis. Figures 3.4 and 3.5 give a vector representation of the lift and drag forces acting on the kite, together with the unit vectors used in [29] to compute the lift.

A different method to compute the lift direction is used by Houska in [18], based on the knowledge of the kite's transverse axis \mathbf{e}_t . Knowing that the lift is perpendicular both to the transverse axis and to the relative velocity direction, one can write:

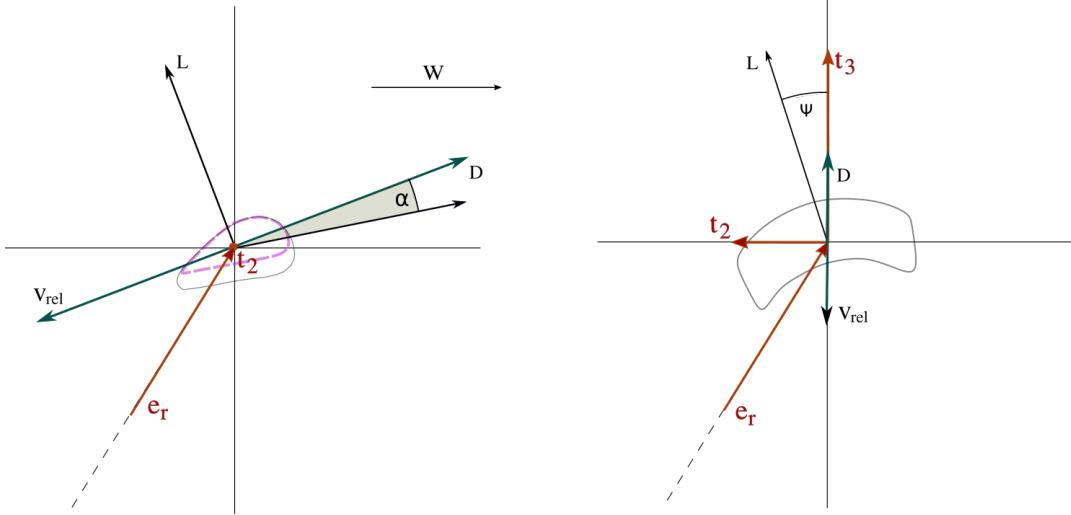


Figure 3.4: Two-dimensional projections of the lift and drag forces acting on a kite. The basis vectors used in reference [29] are represented. The purple dotted line represents the Clark-Y profile of the kite.

$$\mathbf{L} = \frac{1}{2} \rho A C_L(\alpha) |\mathbf{v}_{rel}|^2 (\mathbf{w}_e \times \mathbf{e}_t), \quad (3.7)$$

where \mathbf{w}_e is the unit vector identifying the effective wind velocity, that is $\mathbf{w}_e = \mathbf{v}_w - \mathbf{v}_k = -\mathbf{v}_{rel}$. In this setting the roll angle is defined as $\sin(\psi) = \mathbf{e}_t \cdot \mathbf{e}_r$. It can be noticed that, when the roll angle is zero, $\mathbf{e}_t = -\mathbf{t}_2$.

Given that in two dimensions the roll angle is always zero, the expression of the lift force simplifies. Indeed, expression 3.5 reduces to:

$$\mathbf{L} = \frac{1}{2} \rho A C_L(\alpha) |\mathbf{v}_{rel}|^2 \mathbf{t}_3 \quad (3.8)$$

Which means that the kite's transverse axis \mathbf{e}_t is fixed perpendicularly to the two-dimensional plane in which it can move. Therefore, in two dimensions, the orientation of \mathbf{e}_t can be set simply by knowing the wind speed vector and heading the longitudinal axis in the opposite direction.

3.3 Two-dimensional system

In the simplified two-dimensional framework we indicate with $\hat{\mathbf{x}}$ the unit vector oriented along the horizontal ground line and with $\hat{\mathbf{z}}$ the unit vector perpendicular to $\hat{\mathbf{x}}$ pointing to the sky. The block position in this framework is given by the following vector:

$$\mathbf{r}_b = x_b \hat{\mathbf{x}} + z_b \hat{\mathbf{z}}, \quad (3.9)$$

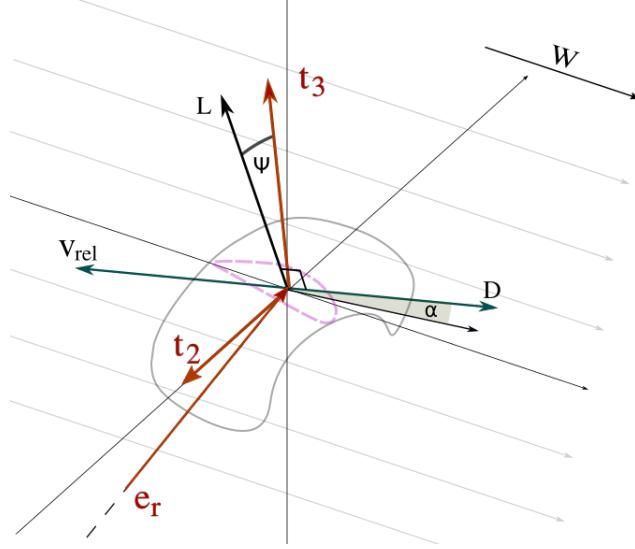


Figure 3.5: Three-dimensional representation of the lift and drag forces acting on a kite, with depiction of the basis vectors used to compute the lift in reference [29]. The purple dotted line represents the Clark-Y profile of the kite.

where x_b and z_b represent the distance of the block with respect to the origin of the axes. Since the block is forced to stay on the ground, z_b is always equal to zero. The position of the kite with respect to the origin of the axes is obtained by introducing θ , the angle made by the tether and the positive direction of the \hat{x} axis:

$$\begin{aligned} \mathbf{r}_k &= (x_b + R \cos(\theta))\hat{x} + (z_b + R \sin(\theta))\hat{z} \\ &= (x_b + R \cos(\theta))\hat{x} + R \sin(\theta)\hat{z}, \end{aligned} \quad (3.10)$$

where R is the fixed length of the tether. Deriving this equation we can find the velocity and the acceleration of the kite:

$$\mathbf{v}_k = (\dot{x}_b - R \sin(\theta)\dot{\theta})\hat{x} + R \cos(\theta)\dot{\theta}\hat{z} \quad (3.11)$$

$$\mathbf{a}_k = (\ddot{x}_b - R \cos(\theta)\dot{\theta}^2 - R \sin(\theta)\ddot{\theta})\hat{x} + (-R \sin(\theta)\dot{\theta}^2 + R \cos(\theta)\ddot{\theta})\hat{z} \quad (3.12)$$

In this model the wind can blow only on the \hat{x} and \hat{z} axes. The wind velocity is then identified by the vector:

$$\mathbf{v}_w = v_{w_x}\hat{x} + v_{w_z}\hat{z}. \quad (3.13)$$

And the relative velocity between the kite and the wind is computed as:

$$\mathbf{v}_{rel} = \mathbf{v}_k - \mathbf{v}_w. \quad (3.14)$$

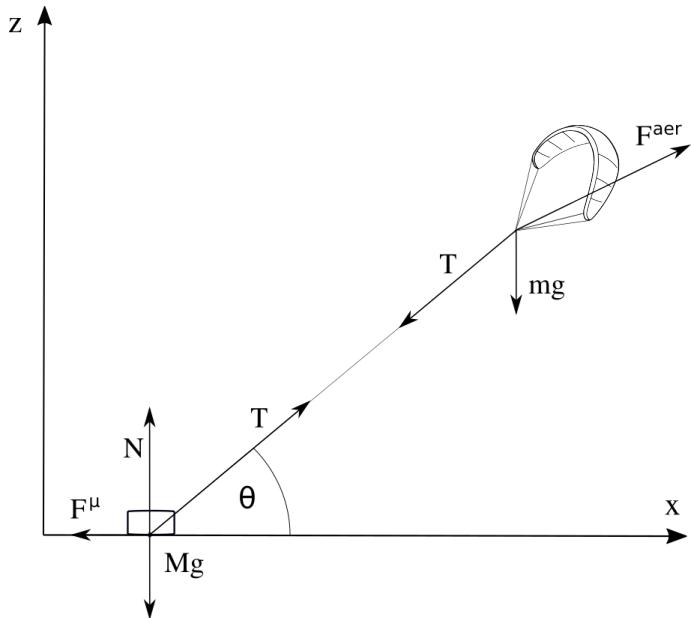


Figure 3.6: Schematic representation of the two-dimensional model with the forces acting on the kite and the block. \mathbf{F}^{aer} indicates the sum of the lift and drag forces.

3.3.1 Polar description of the two-dimensional system

To solve the system in polar coordinates, a transformation to the polar unit vectors is necessary. For the polar coordinates, the following relation holds:

$$\begin{cases} \hat{\mathbf{e}}_r = \cos(\theta)\hat{\mathbf{x}} + \sin(\theta)\hat{\mathbf{z}} \\ \hat{\mathbf{e}}_\theta = -\sin(\theta)\hat{\mathbf{x}} + \cos(\theta)\hat{\mathbf{z}} \end{cases}$$

Where $\hat{\mathbf{e}}_r$ and $\hat{\mathbf{e}}_\theta$ are respectively the radial and angular polar unit vectors.

Since the cable is not extensible the radial component of the kite velocity is always zero, while the tangential component can be expressed using equation 3.11. In the same way the radial component of the kite acceleration is zero while the angular one can be expressed via 3.12.

The equations of motion of the system are found by applying Newton's law to the kite and to the block, taking into account the constraints of the fixed length of the tether and the block forced to stay on the ground. The forces acting on a kite of mass m_k are the lift \mathbf{L} , the drag \mathbf{D} , the tension of the cable \mathbf{T} and the gravitational force $m_k\mathbf{g}$. The forces acting on a block of mass M are the tension of the cable \mathbf{T} , the normal reaction of the plane (or of the rail) \mathbf{N} , the ground friction \mathbf{F}^μ and the gravitational force $M\mathbf{g}$. A depiction of the two-dimensional system with the forces acting on the two items is given in figure 3.6, where \mathbf{F}^{aer} indicates the sum of the lift and drag forces.

For the kite one can write:

$$m_k \mathbf{a}_k = -\mathbf{T} + \mathbf{L} + \mathbf{D} + m_k \mathbf{g}, \quad (3.15)$$

where the tension \mathbf{T} is defined with a negative sign for the kite. Expressing this result separately on the x and z axes and applying polar transformations, the equation 3.15 becomes:

$$\begin{cases} m[\ddot{x}_b - R\ddot{\theta}\sin(\theta) - R\dot{\theta}^2\cos(\theta)] = -T\cos(\theta) + L_x + D_x \\ m[R\ddot{\theta}\cos(\theta) - R\dot{\theta}^2\sin(\theta)] = -T\sin(\theta) - mg + L_z + D_z \end{cases} \quad (3.16)$$

where $T\cos(\theta)$ and $T\sin(\theta)$ are the projections of the rope tension on the x and z axes, meaning that the tension force can be written as: $\mathbf{T} = T(\cos(\theta)\hat{x} + \sin(\theta)\hat{z})$.

For the block one can write:

$$M\ddot{x}_b = \mathbf{T} + \mathbf{N} + \mathbf{F}^\mu, \quad (3.17)$$

where \ddot{x}_b is the block acceleration. Expressing the constraint of the block forced to stay on the ground by using $z_b = 0$, the z component of the 3.17 can be written as:

$$N = T\sin(\theta) - Mg, \quad (3.18)$$

where N is the module of the normal reaction of the plane, $\mathbf{N} = N\hat{z}$.

The x component of equation 3.17 can be expressed as:

$$M\ddot{x}_b = T\cos(\theta) + F^\mu. \quad (3.19)$$

where F^μ is the module of the friction force, $\mathbf{F}^\mu = F^\mu\hat{x}$. The kite and the block are tied to each other due to the presence of the cable, therefore their equations of motion are coupled. The system we need to solve is:

$$\begin{cases} m[\ddot{x}_b - R\ddot{\theta}\sin(\theta) - R\dot{\theta}^2\cos(\theta)] = -T\cos(\theta) + L_x + D_x \\ m[R\ddot{\theta}\cos(\theta) - R\dot{\theta}^2\sin(\theta)] = -T\sin(\theta) - mg + L_z + D_z \\ M\ddot{x}_b = T\cos(\theta) + F^\mu \end{cases} \quad (3.20)$$

Cramer's rule has been used to work out the solution of this system composed by three equations and three unknowns, to avoid problems at the poles. To do so one needs to write the system in the form $Ax = b$, where A is a 3×3 matrix with non-zero determinant, $x = (\ddot{x}_b, \ddot{\theta}, T)$ is the column vector of the unknown variables and b is another column vector of values. With this hypothesis the system has a unique solution where the unknown variables can be found by:

$$x_i = \frac{\det(A_i)}{\det(A)} \quad i = 1, \dots, n \quad (3.21)$$

where A_i is the matrix formed by replacing the i -th column of A by the column vector b .

In section 3.2 we already discussed how to find the components of the lift and drag forces on the axes, both in two and three dimensions, but we still need to find the expression for the friction force \mathbf{F}^μ , which is a function of the state of motion of the block and the cable tension. The friction force in this system is all directed along the \hat{x} axis, and it is headed in the opposite direction with respect to the block velocity if the block is moving, or to the horizontal component of the cable tension if the block is still. Let us now deal with the two cases separately.

If the block is moving, namely $|\dot{x}_b| \neq 0$, the friction force is proportional to the normal reaction of the ground on the block N , and its direction is opposite to the direction of the motion of the block. The module of the friction is then equal to its maximum value. In this setting we can write the friction force as:

$$\begin{aligned}\mathbf{F}^\mu &= \mathbf{F}_{Max}^\mu = \\ &- \mu |N| \frac{\dot{\mathbf{x}}_b}{|\dot{x}_b|} \\ &= -\mu |Mg - T \sin(\theta)| \hat{\mathbf{v}}_b,\end{aligned}\tag{3.22}$$

where μ is the dynamic friction coefficient and $\hat{\mathbf{v}}_b$ is the unit vector pointing in the block velocity direction. In equation 3.22 we are considering the absolute value of the normal force N because the block is bounded on the ground by a rail which contains it. Consequently, when there is a positive tension on the block, if the weight force of the block is greater than the vertical component of the tension, the normal reaction force is exerted by the plane on the block and it is directed upwards, and in this case the friction force is exerted between the block and the plane. In the opposite case the normal reaction force it is exerted by the upper part of the rail on the block and it is directed downwards, and in the same way the friction force is exerted between the block and the upper part of the rail containing it. Using the absolute value of N we are considering both cases. Therefore, to write the friction force explicitly we have to deal with the modulus of the normal reaction force $|N|$. Since we do not know T , to solve equation 3.22 we can first assume that $Mg > T \sin(\theta)$. In this case $\mathbf{F}^\mu = -\mu(Mg - T \sin(\theta))\hat{\mathbf{v}}_b$ and equation 3.19 becomes:

$$M\ddot{x}_b = T \cos(\theta) - \mu(Mg - T \sin(\theta)) \frac{\dot{x}_b}{|\dot{x}_b|}\tag{3.23}$$

Inserting this expression into the 3.20 and writing all of it into the form $Ax = b$ we get:

$$A = \begin{bmatrix} m & -mR \sin(\theta) & \cos(\theta) \\ 0 & mR \cos(\theta) & \sin(\theta) \\ M & 0 & -\cos(\theta) - \mu \sin(\theta) \frac{\dot{x}_b}{|\dot{x}_b|} \end{bmatrix}\tag{3.24}$$

$$x = \begin{bmatrix} \ddot{x}_b \\ \ddot{\theta} \\ T \end{bmatrix} \quad b = \begin{bmatrix} L_x + D_x + mR\dot{\theta}^2 \cos(\theta) \\ L_z + D_z - mg + mR\dot{\theta}^2 \sin(\theta) \\ -\mu Mg \frac{\dot{x}_b}{|\dot{x}_b|} \end{bmatrix}\tag{3.25}$$

On the other hand, if we assume that $Mg < T \sin(\theta)$, the friction force would be $\mathbf{F}^\mu = -\mu(T \sin(\theta) - Mg)\hat{\mathbf{v}}_b$. In this case equation 3.19 becomes:

$$M\ddot{x}_b = T \cos(\theta) - \mu(T \sin(\theta) - Mg) \frac{\dot{x}_b}{|\dot{x}_b|},\tag{3.26}$$

from which:

$$A = \begin{bmatrix} m & -mR\sin(\theta) & \cos(\theta) \\ 0 & mR\cos(\theta) & \sin(\theta) \\ M & 0 & -\cos(\theta) + \mu \sin(\theta) \frac{\dot{x}_b}{|\dot{x}_b|} \end{bmatrix} \quad (3.27)$$

$$x = \begin{bmatrix} \ddot{x}_b \\ \ddot{\theta} \\ T \end{bmatrix} \quad b = \begin{bmatrix} L_x + D_x + mR\dot{\theta}^2 \cos(\theta) \\ L_z + D_z - mg + mR\dot{\theta}^2 \sin(\theta) \\ +\mu Mg \frac{\dot{x}_b}{|\dot{x}_b|} \end{bmatrix} \quad (3.28)$$

The only difference between the two formulations is the presence of two negative signs in equations 3.24 and 3.25, which become positive in equations 3.27 and 3.28. Once we have the matrix A and the column vector b we can solve the unknowns using the Cramer's rule as explained.

Let us now deal with the case in which the block is still, namely $\dot{x}_b = 0$. In this circumstance solving the system is more tricky because the direction of the friction force depends on the projection of the tension on the x axis: indeed, it is the tension force that allows the block to overcome the ground friction and start moving on the x axis. Moreover, it is not necessarily true that the projection of the tension on the horizontal is able to overcome the friction force allowing the block to move, meaning that the same friction force is not necessarily equal to its maximum possible value. For the friction force in this setting we can only write:

$$\mathbf{F}^\mu \leq \mathbf{F}_{Max}^\mu \text{ with } \mathbf{F}_{Max}^\mu = -\mu|Mg - T \sin(\theta)| \frac{\mathbf{T}_x}{|\mathbf{T}|} = -\mu|Mg - T \sin(\theta)| \cos(\theta) \hat{\mathbf{x}}, \quad (3.29)$$

and $\mathbf{F}^\mu = \mathbf{F}_{Max}^\mu$ only when the x component of the tension force is capable to overcome the maximum value of the friction. If this does not happen, it is true that $\mathbf{F}^\mu = -T \cos(\theta) \hat{\mathbf{x}}$ and $M\ddot{x}_b = 0$.

Since we do not know both the tension and the friction force, we can first solve the system assuming that the projection of the tension force on the horizontal is bigger or equal to the maximum friction force. In this particular case we can write $\mathbf{F}^\mu = \mathbf{F}_{Max}^\mu$ and the system needs again to be solved dealing with the modulus first.

If $Mg > T \sin(\theta)$ we end up with:

$$A = \begin{bmatrix} m & -mR\sin(\theta) & \cos(\theta) \\ 0 & mR\cos(\theta) & \sin(\theta) \\ M & 0 & -\cos(\theta) - \mu \sin(\theta) \cos(\theta) \end{bmatrix} \quad (3.30)$$

$$x = \begin{bmatrix} \ddot{x}_b \\ \ddot{\theta} \\ T \end{bmatrix} \quad b = \begin{bmatrix} L_x + D_x + mR\dot{\theta}^2 \cos(\theta) \\ L_z + D_z - mg + mR\dot{\theta}^2 \sin(\theta) \\ -\mu Mg \cos(\theta) \end{bmatrix}, \quad (3.31)$$

while if $Mg < T \sin(\theta)$:

$$A = \begin{bmatrix} m & -mR\sin(\theta) & \cos(\theta) \\ 0 & mR\cos(\theta) & \sin(\theta) \\ M & 0 & -\cos(\theta) + \mu\sin(\theta)\cos(\theta) \end{bmatrix} \quad (3.32)$$

$$x = \begin{bmatrix} \ddot{x}_b \\ \ddot{\theta} \\ T \end{bmatrix} \quad b = \begin{bmatrix} L_x + D_x + mR\dot{\theta}^2\cos(\theta) \\ L_z + D_z - mg + mR\dot{\theta}^2\sin(\theta) \\ +\mu Mg\cos(\theta) \end{bmatrix} \quad (3.33)$$

Where the only difference between equations 3.24 - 3.25 and 3.27 - 3.28 is the replacement of $\frac{\dot{x}_b}{|\ddot{x}_b|}$ with $\frac{T_x}{T} = \cos(\theta)$.

Now \ddot{x}_b , $\ddot{\theta}$ and T can be found using the Cramer's rule, but the initial guess has to be verified: once found T , one has to find F^μ and check if $|T_x| > |F^\mu|$. If this is true we can keep our result, while if it is not it means that the horizontal component of the tension force is not capable to overcome the friction and tow the block; the acceleration of the block in this case is zero $\ddot{x}_b = 0$. This means that $F^\mu < F_{Max}^\mu$ and the friction force is exactly equal to the horizontal component of the tension force and opposite in sign: $F^\mu = -T_x = -T\cos(\theta)$. In this case the system in the 3.20 reduces to:

$$\begin{cases} m[\ddot{x}_b - R\ddot{\theta}\sin(\theta) - R\dot{\theta}^2\cos(\theta)] = -T\cos(\theta) + L_x + D_x \\ m[R\ddot{\theta}\cos(\theta) - R\dot{\theta}^2\sin(\theta)] = -T\sin(\theta) - mg + L_z + D_z \\ M\ddot{x}_b = 0 \end{cases} \quad (3.34)$$

This system has one single solution:

$$A = \begin{bmatrix} m & -mR\sin(\theta) & \cos(\theta) \\ 0 & mR\cos(\theta) & \sin(\theta) \\ M & 0 & 0 \end{bmatrix} \quad (3.35)$$

$$x = \begin{bmatrix} \ddot{x}_b \\ \ddot{\theta} \\ T \end{bmatrix} \quad b = \begin{bmatrix} L_x + D_x + mR\dot{\theta}^2\cos(\theta) \\ L_z + D_z - mg + mR\dot{\theta}^2\sin(\theta) \\ 0 \end{bmatrix} \quad (3.36)$$

From here we can finally find the values of the acceleration $\ddot{\theta}$ and the tension force modulus T .

Once the system has been solved and the block acceleration, the angular acceleration and the tension force values have been found, one can proceed by numerically integrating the equations of motion to find the velocity and the position of both the block and the kite in that simulation step.

3.4 Cartesian model

The dynamics of the kite-block system has been expressed and solved in Cartesian coordinates for both for the two-dimensional and the three-dimensional models. In this section we present the equations of dynamics in vector form, avoiding for now to decompose the vectors on the axes. All that is written in this section is therefore valid for both the two-dimensional and the three-dimensional models. We remind that the forces acting on the kite are the lift \mathbf{L} , the drag \mathbf{D} ,

the gravitational force $m_k \mathbf{g}$ and the cable tension \mathbf{T} , while the forces acting on the block are the gravitational force $M \mathbf{g}$, the cable tension \mathbf{T} , the friction \mathbf{F}^μ and the normal reaction of the plane (or of the rail) \mathbf{N} . By explicitly writing the two constraints of the fixed rope length and the block forced to stay on the ground, the system of equations for the dynamics can be written as follows:

$$\begin{cases} m\ddot{\mathbf{x}}_k = \mathbf{F}^{aer} + m\mathbf{g} - \mathbf{T} \\ M\ddot{\mathbf{x}}_b = \mathbf{F}^\mu + \mathbf{N} + M\mathbf{g} + \mathbf{T} \\ z_b = 0 \\ |\mathbf{r}(t)| = R \quad \text{where } \mathbf{r} = \mathbf{x}_k - \mathbf{x}_b \end{cases} \quad (3.37)$$

Where \mathbf{F}^{aer} represents the total aerodynamic force acting on the kite, given by the sum of the lift and the drag. This systems contains a total of 6 equations and 6 unknown for the two-dimensional setting and 8 equations and 8 unknown for the three-dimensional one. The solution is found by assuming that the initial positions and velocities are given. After computing the relative velocity \mathbf{v}_{rel} , the total aerodynamic force \mathbf{F}^{aer} can be easily computed as shown in section 3.2. To solve the dynamics one needs to find out both the acceleration of the kite $\ddot{\mathbf{x}}_k$ and the acceleration of the block $\ddot{\mathbf{x}}_b$, which depend on the tension force and the normal reaction force on the block, whose magnitudes are unknown.

First of all, one can easily see that $\ddot{z}_b = 0$. From this, the equation for the equilibrium of the forces on the z axis for the block reads:

$$N = Mg - T_z. \quad (3.38)$$

The modulus of the friction force – which is exerted between the block and the ground or the block and the upper side of the rail – is proportional to the modulus of the sum of the forces acting on the block on the vertical direction:

$$|\mathbf{F}^\mu| = \mu|N| \quad (3.39)$$

$$= \mu|Mg - T_z| \quad (3.40)$$

Let us consider now the relationship $\mathbf{r} = \mathbf{x}_k - \mathbf{x}_b$, from which $\dot{\mathbf{r}} = \ddot{\mathbf{x}}_k - \ddot{\mathbf{x}}_b$. Making the ratio of the first two equations of system 3.37 by the respective masses and subtracting the results we can write:

$$\ddot{\mathbf{r}} = \frac{\mathbf{F}^{aer} - \mathbf{T}}{m} - \frac{\mathbf{F}^\mu + \mathbf{N} + \mathbf{T}}{M}, \quad (3.41)$$

where the gravitational force has disappeared.

Considering now the second constraint of the system 3.37, namely $|\mathbf{r}(t)| = R$, we can compute the derivative of $|\mathbf{r}(t)|$ with respect to the time (the time dependence in the following passages is suppressed for clarity):

$$\frac{1}{2} \frac{d}{dt} |\mathbf{r}|^2 = |\mathbf{r}| \hat{\mathbf{r}} \cdot \dot{\mathbf{r}} = \mathbf{r} \cdot \dot{\mathbf{r}} = 0 \quad (3.42)$$

Deriving a second time we can write:

$$\frac{d}{dt} \mathbf{r} \dot{\mathbf{r}} = 0 = \dot{\mathbf{r}}^2 + \mathbf{r} \ddot{\mathbf{r}}, \quad (3.43)$$

from which:

$$\ddot{\mathbf{r}} \mathbf{r} = -\dot{\mathbf{r}}^2 \quad (3.44)$$

Given this relation, we can now equate $-\dot{\mathbf{r}}^2$ to the product of equation 3.41 with \mathbf{r} , which means inserting the constraint of fixed rope length indirectly in the system:

$$\ddot{\mathbf{r}} \cdot \mathbf{r} = -|\mathbf{T}|R(m^{-1} + M^{-1}) + \frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} - \frac{(\mathbf{F}^\mu + \mathbf{N}) \cdot \mathbf{r}}{M} = -\dot{\mathbf{r}}^2, \quad (3.45)$$

where \mathbf{T} is all directed along \mathbf{r} , therefore $\mathbf{T} \cdot \mathbf{r} = |\mathbf{T}|R$. The second equality of the 3.45 can now be used to find out the modulus of tension force along the rope direction, $|\mathbf{T}|$, by writing explicitly both \mathbf{N} and \mathbf{F}^μ as functions of $|\mathbf{T}|$:

$$|\mathbf{T}|R(m^{-1} + M^{-1}) = \frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} - \frac{(\mathbf{F}^\mu + \mathbf{N}) \cdot \mathbf{r}}{M} + \dot{\mathbf{r}}^2 \quad (3.46)$$

We are now going to solve the above equation separately for the two and three-dimensional models, writing explicitly the vectors. Once found the expressions for the friction and tension forces, the velocities and positions of the kite and the block can be obtained by integrating the accelerations.

Notice that, since the constraint $|\mathbf{r}(t)| = R$ is inserted indirectly in the dynamics, the distance between block and kite must be fixed after the integration. To do this, since the block is fixed at $z_b = 0$, the kite position is adjusted according to:

$$\mathbf{x}'_k = \mathbf{x}_k + \frac{\mathbf{x}_k - \mathbf{x}_b}{|\mathbf{x}_k - \mathbf{x}_b|} R \quad (3.47)$$

3.4.1 Cartesian description of the two-dimensional model

In this section we solve the Cartesian model of the system for the two-dimensional setting. Let us recall that, in polar coordinates, θ is defined as the angle made by the cable and the positive direction of the x axis. Knowing this one can write:

$$N = Mg - T \sin(\theta), \quad (3.48)$$

where $\mathbf{N} = N\hat{\mathbf{z}}$. Moreover $\mathbf{F}^\mu = F^\mu \hat{\mathbf{x}}$. From this we can write the friction force as a function of the block status of motion as:

$$\begin{cases} F^\mu = -\mu|N|\frac{\dot{x}_b}{|\dot{x}_b|} & \text{if } \dot{x}_b \neq 0 \\ F^\mu = -\mu|N|\frac{T_x}{|T_x|} & \text{if } \dot{x}_b = 0 \text{ and } |T_x| \geq |F_{Max}^\mu| \\ F^\mu = -T_x = -T \cos(\theta) & \text{if } \dot{x}_b = 0 \text{ and } |T_x| < |F_{Max}^\mu|. \end{cases} \quad (3.49)$$

To find the tension force one has to know which of the three expressions for the friction force to use. If the block velocity is nonzero, namely $|\dot{x}_b| \neq 0$, the friction force is always equal to its maximum value and it is headed in the opposite direction with respect to the block velocity. In this configuration for the tension we can write, using equations 3.48 and 3.49:

$$-TR(m^{-1} + M^{-1}) + \frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} - \frac{(-\mu|Mg - T \sin(\theta)|\frac{\dot{x}_b}{|\dot{x}_b|} + (Mg - T \sin(\theta))\hat{z})}{M} \cdot \mathbf{r} = -\dot{\mathbf{r}}^2 \quad (3.50)$$

Where $T = |\mathbf{T}|$. To solve for T, one has first to deal with the modulus. If $Mg > T \sin(\theta)$:

$$T = \frac{\frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - g \left(r_z - \mu r_x \frac{\dot{x}_b}{|\dot{x}_b|} \right)}{R \frac{m+M}{mM} - \frac{\sin(\theta)}{M} \left(r_z - \mu r_x \frac{\dot{x}_b}{|\dot{x}_b|} \right)}, \quad (3.51)$$

while if $Mg < T \sin(\theta)$:

$$T = \frac{\frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - g \left(r_z + \mu r_x \frac{\dot{x}_b}{|\dot{x}_b|} \right)}{R \frac{m+M}{mM} - \frac{\sin(\theta)}{M} \left(r_z + \mu r_x \frac{\dot{x}_b}{|\dot{x}_b|} \right)}, \quad (3.52)$$

where $r_x = x_k - x_b$ and $r_z = z_k - z_b$.

If the block is still, namely $|\dot{x}_b| = 0$, we first have to solve the equation assuming that the tension force is bigger or equal to the maximum friction force, whose formula is given by the second equation of the 3.49. Again, if $Mg > T \sin(\theta)$ we obtain:

$$T = \frac{\frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - g \left(r_z - \mu r_x \cos(\theta) \right)}{R \frac{m+M}{mM} - \frac{\sin(\theta)}{M} \left(r_z - \mu r_x \cos(\theta) \right)}, \quad (3.53)$$

while if $Mg < T \sin(\theta)$:

$$T = \frac{\frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - g \left(r_z + \mu r_x \cos(\theta) \right)}{R \frac{m+M}{mM} - \frac{\sin(\theta)}{M} \left(r_z + \mu r_x \cos(\theta) \right)} \quad (3.54)$$

Once found the values for T and F^μ , one has to check if the assumption $|T_x| \geq |F_{Max}^\mu|$ is true. If it is the results can be kept, while if it is not, the tension has to be recomputed by imposing $F^\mu = -T \cos(\theta)$, from which:

$$T = \frac{\frac{F^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - gr_z}{R \frac{m+M}{mM} - \frac{\sin(\theta)}{M} r_z - \frac{r_x \cos(\theta)}{M}}. \quad (3.55)$$

In this last case the acceleration of the block on the x axis is zero because the tension force is not capable to overcome the friction force.

Once found T we can compute N and F^μ , and finally $\ddot{\mathbf{x}}$ and $\ddot{\mathbf{x}}_b$. The dynamic system can be finally expressed as:

$$\begin{cases} m\ddot{x}_k = F_x^{aer} - T_x \\ m\ddot{z}_k = F_z^{aer} - T_z - mg \\ M\ddot{x}_b = F_x^\mu + T_x \\ M\ddot{z}_b = 0 \end{cases} \quad (3.56)$$

This system can be integrated to obtain the speed and the position of both the kite and the block.

3.4.2 Cartesian description of the three-dimensional model

In this section we develop a complete solution of the three-dimensional Cartesian model presented in section 3.4. To develop the model in three dimension we need to define two angles: θ , which is the angle made by the cable and the z axis, and ϕ , the angle made by the projection of the cable in the xy plane and the x axis. Notice that in this setting we are employing a different definition of θ with respect to the two-dimensional case. Figure 3.7 shows a depiction of the three-dimensional system with a representation of the forces acting on it.

In three dimensions the components of the tension force with respect to the block reference system read:

$$\begin{cases} T_x = T \sin(\theta) \cos(\phi) \\ T_y = T \sin(\theta) \sin(\phi) \\ T_z = T \cos(\theta), \end{cases} \quad (3.57)$$

and the normal reaction force on the block becomes:

$$N = Mg - T \cos(\theta). \quad (3.58)$$

For the friction force we can write:

$$\begin{cases} \mathbf{F}^\mu = -\mu|N|\frac{\mathbf{v}_b}{|\mathbf{v}_b|} = -\mu\frac{|N|}{|\mathbf{v}_b|}(v_{b,x}\hat{\mathbf{x}} + v_{b,y}\hat{\mathbf{y}}) & \text{if } v_b \neq 0 \\ \mathbf{F}^\mu = -\mu|N|\frac{\mathbf{T}_{xy}}{|\mathbf{T}_{xy}|} = -\mu\frac{|N|}{|\mathbf{T}_{xy}|}(T_x\hat{\mathbf{x}} + T_y\hat{\mathbf{y}}) & \text{if } v_b = 0 \text{ and } |\mathbf{T}_{xy}| \geq |\mathbf{F}_{Max}^\mu| \\ \mathbf{F}^\mu = -\mathbf{T}_{xy} = -T(\sin(\theta) \cos(\phi)\hat{\mathbf{x}} + \sin(\theta) \sin(\phi)\hat{\mathbf{y}}) & \text{if } v_b = 0 \text{ and } |\mathbf{T}_{xy}| < |\mathbf{F}_{Max}^\mu|, \end{cases} \quad (3.59)$$

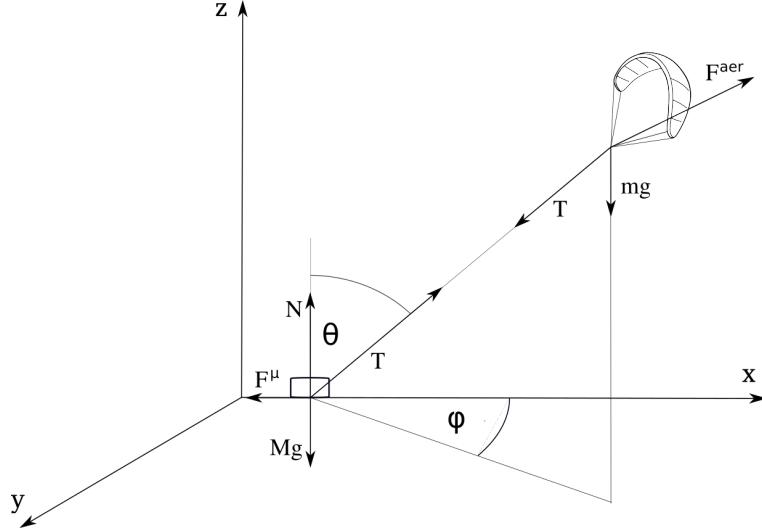


Figure 3.7: Schematic representation of the three-dimensional model with the forces acting on the kite and the block. \mathbf{F}^{aer} indicates the sum of the lift and drag forces.

where \mathbf{v}_b is the block velocity on the x and y axes and \mathbf{T}_{xy} is the projection of the tension force on the horizontal plane. Notice that, even if the block can move only on the x axis, here we are considering the block velocity, the tension force and the friction force on both x and y axes. This is not strictly needed in this case, but allows to generalize easily the system, for example in the case of absence of the rail.

Let us solve again for the friction and tension forces and the status of motion of the block. We know that if the block is moving, namely $|\mathbf{v}_b| \neq 0$, the friction force is always equal to its maximum value and it is headed in the opposite direction with respect to the block velocity, so we use the first expression of system 3.59. Suppose that $Mg > T \cos(\theta)$. In this case the expression for the tension T (equation 3.46) becomes:

$$T = \frac{\frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - g \left[z_k - \frac{\mu}{|\mathbf{v}_b|} (v_x r_x + v_y r_y) \right]}{R \frac{m+M}{mM} - \frac{\cos(\theta)}{M} \left[z_k - \frac{\mu}{|\mathbf{v}_b|} (v_x r_x + v_y r_y) \right]}, \quad (3.60)$$

where $T = |\mathbf{T}|$, $r_x = r_{x,k} - r_{b,x}$, $r_y = r_{y,k} - r_{b,y}$, $v_x = v_{x,k} - v_{b,x}$ and $v_y = v_{y,k} - v_{b,y}$. If, on the other hand, $Mg < T \cos(\theta)$:

$$T = \frac{\frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - g \left[z_k + \frac{\mu}{|\mathbf{v}_b|} (v_x r_x + v_y r_y) \right]}{R \frac{m+M}{mM} - \frac{\cos(\theta)}{M} \left[z_k + \frac{\mu}{|\mathbf{v}_b|} (v_x r_x + v_y r_y) \right]} \quad (3.61)$$

If the block is still, namely $\mathbf{v}_b = 0$, we start the resolution by assuming that the tension force is

bigger or equal to the maximum friction force $\mathbf{F}_{Max}^\mu = -\mu|N|\frac{\mathbf{T}_{xy}}{|\mathbf{T}_{xy}|}$. In this case, if $Mg > T \cos(\theta)$, we get:

$$T = \frac{\frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - g \left[z_k - \mu(\cos(\phi)r_x + \sin(\phi)r_y) \right]}{R \frac{m+M}{mM} - \frac{\cos(\theta)}{M} \left[z_k - \mu(\cos(\phi)r_x + \sin(\phi)r_y) \right]}, \quad (3.62)$$

while if $Mg < T \cos(\theta)$:

$$T = \frac{\frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - g \left[z_k + \mu(\cos(\phi)r_x + \sin(\phi)r_y) \right]}{R \frac{m+M}{mM} - \frac{\cos(\theta)}{M} \left[z_k + \mu(\cos(\phi)r_x + \sin(\phi)r_y) \right]} \quad (3.63)$$

Once found T one can compute \mathbf{F}^μ . At this point one has to check if the assumption that $|\mathbf{T}_{xy}| \geq |\mathbf{F}_{Max}^\mu|$ is true. If it is, the computed values can be kept, while if it is not, the tension has to be recomputed by imposing $\mathbf{F}^\mu = -\mathbf{T}_{xy}$. The modulus of the tension in this case becomes:

$$T = \frac{\frac{\mathbf{F}^{aer} \cdot \mathbf{r}}{m} + \dot{\mathbf{r}}^2 - gz_k}{R \frac{m+M}{mM} - \frac{\sin(\theta)}{M} (\cos(\phi)r_x + \sin(\phi)r_y) - \frac{\cos(\theta)}{M} z_k} \quad (3.64)$$

We are now able to find \mathbf{F}^μ , \mathbf{T} , and finally $\ddot{\mathbf{x}}_k$ and $\ddot{\mathbf{x}}_b$ in each different case. Expressing the components for the accelerations of kite and block in the three dimensions, one can write:

$$\begin{cases} m\ddot{x}_k = F_x^{aer} - T_x \\ m\ddot{y}_k = F_y^{aer} - T_y \\ m\ddot{z}_k = F_z^{aer} - T_z - mg \\ M\ddot{x}_b = F_x^\mu + T_x \\ M\ddot{y}_b = F_y^\mu + T_y \\ M\ddot{z}_b = 0. \end{cases} \quad (3.65)$$

Those equations can be integrated to find the velocities and the positions of the kite and the block at the current time step.

3.5 Dynamics analysis

In this section a simple analysis of the behavior of the two and three-dimensional systems is made. For the analysis, and for the subsequent applications of the reinforcement learning algorithm, a kite of mass 1 kg and area $A = 5 m^2$, and a block of mass 40 kg were used. The cable connecting the two was fixed at the length of $R = 50 m$. The air density was set to $\rho = 1.225 \text{ kg/m}^3$ and the friction coefficient to $\mu = 0.4$.

3.5.1 The lift and drag coefficients

As anticipated, both in the two and three-dimensional models the control angle between the kite longitudinal axis e_l and the relative wind speed \mathbf{v}_{rel} , namely the attack angle α , is modeled. Both the lift and drag coefficients $C_L(\alpha)$ and $C_D(\alpha)$ depend on the angle of attack. As mentioned before, those coefficients are also functions of other parameters such as the body shape, size and orientation, and the density, viscosity and compressibility of the fluid. Usually, their values are computed from the measure of the aerodynamic force acting on the airborne object; the two coefficients are in fact defined as:

$$C_L(\alpha) = \frac{2L}{\rho A v_{rel}^2} \quad (3.66)$$

$$C_D(\alpha) = \frac{2D}{\rho A v_{rel}^2} \quad (3.67)$$

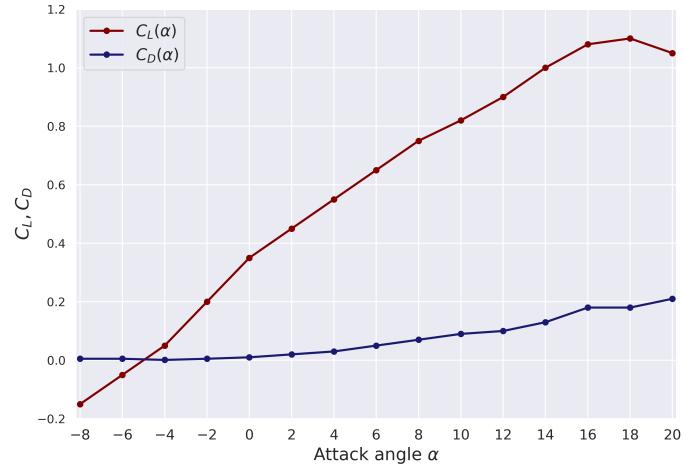
Where ρ is the mass density of the fluid surrounding the body, v_{rel} is the modulus of the speed of the flow relative to the body, A is the reference area, namely the surface area of the body that faces the flow, and L and D are the lift and drag forces.

Canale, Fagiano and Milanese in [7] estimated numerically the values of the lift and drag coefficients as functions of the angle of attack, for a Clark-Y kite. In all the simulations, a set of 15 values for the attack angle and the related coefficients of the cited work have been used. The values are listed in the table in figure 3.8, together with the curve showing their dependence on the angle of attack. Figure 3.8 shows that the drag coefficient increases only a little from about zero at negative attack angles to a maximum of 0.21 at an angle of 20°. On the other hand, the lift coefficient varies strongly with the growth of the attack angle shifting from a slightly negative value at negative angles of attack, to values slightly bigger than 1 at angles bigger than 12°, reaching a peak at 18° and decreasing after. This is the typical shape for the lift coefficient for kites or airplane wings, as a function of the attack angle. The decline of the lift coefficient after a certain angle is due to the separation of the airflow over the surface of the body that happens by tilting the wing over a certain value: from that angle, the more the wing is tilted, the more the flow separates from the body. The positive lift generated at some negative attack angles is due to the manufactured shape of the wing: a wing with above and below symmetry would have zero lift coefficient at 0 attack angle, since in this case the flow is perfectly parallel to the body reference line.

3.5.2 Equilibrium motion

Let us observe the behavior of the three models – two models for the two-dimensional setting and one for the three-dimensional one – when the kite is moving with a fixed attack angle in a fixed wind setting. The wind blows parallel to the positive direction of the x axis. In the chosen initial condition the block is still at the origin of the axes and the kite is situated in air at an angle of $\theta_0 = 45^\circ$ with the x axis and zero groundspeed. For the models to be in the same conditions, the bank angle ψ of the three-dimensional model was kept equal to zero, and the same choice was made for ϕ_0 . To analyse the dynamics in these conditions, the kite is given an initial attack angle, which is kept fixed until the kite falls or the single simulation ends. Each simulation runs up to a maximum of 200 seconds, if the kite does not fall prematurely.

Figure 3.9 shows the instantaneous speed reached by the block after 200 seconds, for different values of the wind speed on the x axis and different values of the attack angles α . The upper image



Attack angle α	$C_L(\alpha)$	$C_D(\alpha)$
-8	-0.15	0.005
-6	-0.05	0.005
-4	0.05	0.001
-2	0.2	0.005
0	0.35	0.01
2	0.45	0.02
4	0.55	0.03
6	0.65	0.05
8	0.75	0.07
10	0.82	0.09
12	0.9	0.1
14	1.0	0.13
16	1.08	0.18
18	1.1	0.18
20	1.05	0.21

Figure 3.8: Lift coefficient C_L (red line) and drag coefficient C_D (blue line) as functions of the attack angle α . The employed values are listed in the table, and are taken from reference [7].

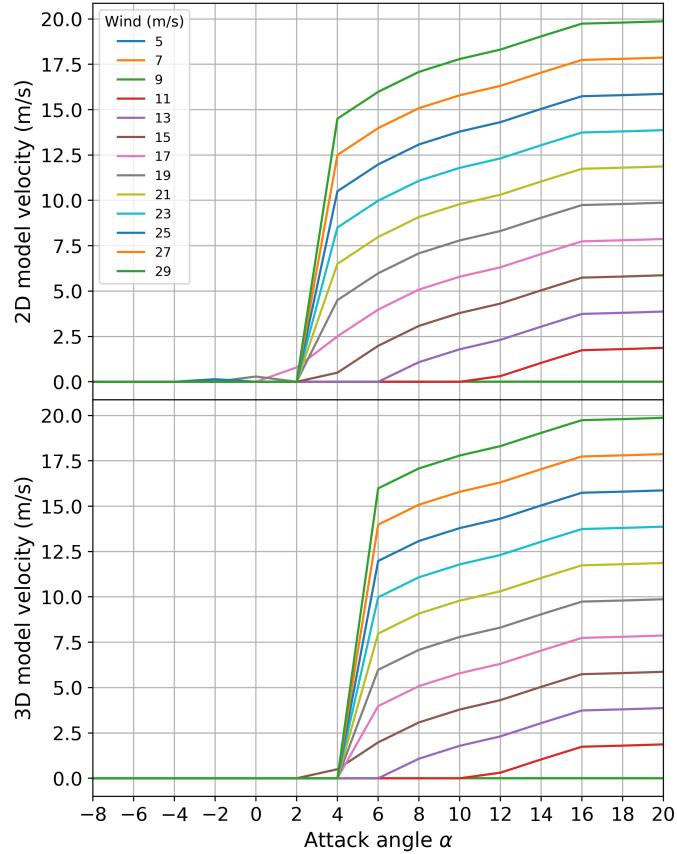


Figure 3.9: Relative velocity of the kite as a function of the wind velocity for different values of the attack angle, for the two-dimensional Cartesian model (upper image) and three-dimensional Cartesian model (lower image).

refers to the behavior of the two-dimensional models, which is exactly the same for the polar and the Cartesian systems. The lower image refers to the three-dimensional Cartesian model.

One can easily notice that the higher is the wind velocity, the higher the velocity that can be reached by the block, for a fixed value of the attack angle. Moreover, for wind velocities of 9 m/s and weaker the block velocity is always zero. This happens because the aerodynamic force acting on the kite is not strong enough to overcome the block friction, therefore the kite gets stuck in the air or falls to the ground. The same happens for attack angles equal and smaller than 2° for the two-dimensional setting and 4° for the three-dimensional one: aside some small exceptions we can see that for these values the block velocity at the end of the simulation is always zero. We just noticed a discrepancy between the two figures: while in the two-dimensional model the kite is able to move the block with an attack angle of $\alpha = 4^\circ$, the same does not happen in the three-dimensional model, where the minimum beneficial attack angle that leads to move the block is $\alpha = 6^\circ$. This difference is due to a computational problem of which only the three-dimensional model is suffering:

whenever the cable direction is getting parallel to the relative velocity vector, the computation of the transverse axis of the kite becomes indeterminate. This leads to an incorrect computation of the lift force, which causes the system to behave in an unpredictable way, as the lift direction depends on the computational precision with which the calculation is carried out. This problem, which has not yet been solved to date, appears when computing the lift with both models studied in section 3.2, namely references [18] and [29], and it is never mentioned in such works.

It has been noticed that in case of a simple fixed wind directed along the x axis, namely $\mathbf{v}_w = v_w \hat{\mathbf{x}}$, depending on the initial condition and the selected attack angle, the kite-block system often reaches an equilibrium condition in which both the kite and the block move with a constant velocity on the horizontal axis. In this condition the forces acting on the kite and the block on the x and z axes must add up to zero. Considering the forces acting on the kite, namely lift, drag, gravity and tether tension, and computing the equilibrium on the two axis, one can find for the x axis:

$$D = T \cos(\theta^{eq}), \quad (3.68)$$

and for the z axis:

$$L - m_k g = T \sin(\theta^{eq}). \quad (3.69)$$

For the block, considering the forces acting on the horizontal one can write:

$$F^\mu = \mu |Mg - T \sin(\theta^{eq})| = T \cos(\theta^{eq}) \quad (3.70)$$

The value of θ at the equilibrium is then given by:

$$\theta^{eq} = \arctan \frac{L - m_k g}{D}. \quad (3.71)$$

Inserting 3.68 and 3.69 in equation 3.71 one can find:

$$D = \mu |(M + m)g - L|, \quad (3.72)$$

and writing explicitly the form of the lift and the drag:

$$\frac{1}{2} \rho A C_D(\alpha) v_{rel}^2 = \mu \left| (M + m)g - \frac{1}{2} \rho A C_L(\alpha) v_{rel}^2 \right| \quad (3.73)$$

From here one can obtain a relation that connects the relative velocity with some parameters of the system:

$$v_{rel}^2 = \frac{\mu(M + m)g}{\frac{1}{2}\rho A(C_D(\alpha) + \mu C_L(\alpha))}, \quad (3.74)$$

or

$$v_{rel} = \pm \sqrt{\frac{\mu(M + m)g}{\frac{1}{2}\rho A(C_D(\alpha) + \mu C_L(\alpha))}}. \quad (3.75)$$

$$(3.76)$$

This means that, at the equilibrium, the relative velocity between the kite and the wind does not depend on the wind velocity, but on other parameters of the system, included the attack angle. This result has been found in the simulations: figure 3.10 shows the relative velocity obtained in the analysis of the two-dimensional Cartesian model, which is constant when varying the wind velocity value for almost every value of the attack angle. When it is not constant, the kite gets stuck in the air or falls to the ground.

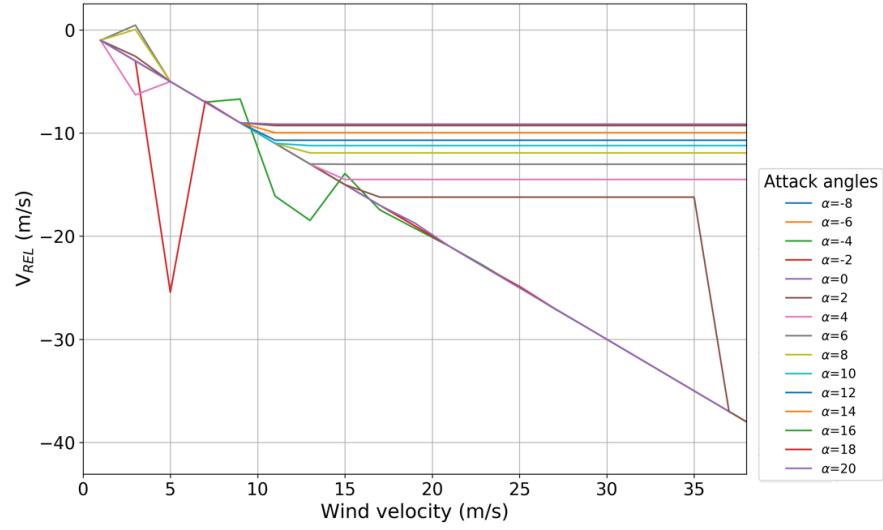


Figure 3.10: Relative velocity of between the kite and the wind as a function of the wind velocity on the x axis for different values of the attack angle α , obtained with the two-dimensional Cartesian code.

Chapter 4

Application of Reinforcement learning to the model

In this chapter we describe how we applied reinforcement learning to the models developed in the previous chapter, and we present the results obtained from the simulations.

The purpose of the use of reinforcement learning is to ensure that the control system of the kite, which is the decision-making agent of the process, learns the best way to maneuver the controllable variables while moving in the environment so as to maximize the speed of the block along the rail. To do this the agent can exclusively act on the two angles of bank and attack.

In this chapter we show that using reinforcement learning we are able not only to obtain some theoretically expected results, but also to learn non trivial policies which allow the agent to move in optimal and somewhat unexpected ways.

4.1 General characteristics of the learning

In this section we describe some general characteristics of the learning processes that are relevant for all the simulations described in the chapter.

The model parameters employed for the simulations are the same as those used for the simplified analysis of section 3.5, and are listed in table 4.1.

Parameter	Value
Kite mass m_k	1kg
Kite area A	5 m ²
Block mass M	40kg
Tether length R	50m
Air density ρ	1.225 kg/m ³
Friction coefficient μ	0.4

Table 4.1: Model parameters employed for all the simulations.

As mentioned above, the purpose of applying reinforcement learning to the models is to make

the kite control system use its experience – through trials and errors – to find the optimal behavior that allows it to pull the block with the goal of maximizing its speed along the rail, and consequently the traveled distance in a fixed amount of time. In order to achieve this goal the SARSA algorithm for control was employed. As we described in Chapter 2, SARSA allows to search for the optimal policy by a process that alternates the evaluation of the current policy and its improvement by acting (almost) greedily on the estimate of the state-action value function $Q(s, a)$ computed with the previous policy.

The actions allowed to the agent in this setting are defined as changes of both the attack angle α and the bank angle ψ (in the three-dimensional model), or the attack angle only (in the two-dimensional model). In order to act on the control angles the agent needs to know their value at each step of the learning simulation, therefore these angles constitute a fundamental part of the state-space, and will belong to a discrete set of selected values. The actions are at most three: the agent can increase or decrease the value of both angles, bringing them to the next or previous one in the set, or keep them the same. This is true for each angle except for the smallest one in the set, where the only possible actions are to keep or increase it, and for the biggest one, where the possible actions are to keep or decrease it. As explained in section 3.5.1, we choose 15 values for the attack angle, whose respective lift and drag coefficients were taken from reference [7]. Concerning the bank angle, a set of discrete values was manually selected.

The duration of the learning episodes varies among the simulations depending on the complexity of the problem at hand, but all of them last at least 200 seconds of flight, unless the kite falls prematurely. During each episode the agent is allowed to act on the values of both angles every 0.1 seconds. This means that, in the shorter simulations, the agent can take at maximum 2000 actions without the kite falling to the ground. The duration of each episode is limited, but the effective time horizon seen by the agent can be set longer than the maximum length of the episodes. This is done by regulating the discount factor γ present in equations 4.1 and 4.2, whose value belongs to the range $[0, 1]$. The bigger the discount factor, the more importance it is given to long-term rewards. In the present case the value of γ was fixed to 0.9999999999, which corresponds to an effective time horizon of $1/(1 - \gamma) = 10^{10}$ integration steps, or 10^6 seconds.

As already mentioned, the reward given to the agent at each learning step is the instantaneous speed of the block on the x axis, since this is exactly the signal that we want to maximize. This means that the faster the block travels, the more the agent is rewarded. Because of the definition of the reward, the return gained at the end of each episode coincides with the amount of road traveled by the block during the episode, if the kite does not fall. A penalty is given to the agent each time the kite falls to the ground, adding a negative number to the instantaneous reward, and terminating the episode. Consequently the updates performed on the Q matrix are different if the kite falls or not. The general SARSA update for the estimate of the state-action value function reads:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \eta[R_t + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)] \quad (4.1)$$

This update is performed every time a new action is taken. In the last step of each episode, since there is no next state S_{t+1} nor next action A_{t+1} , one has $Q(S_{t+1}, A_{t+1}) = 0$. If the kite falls, there are no next state S_{t+1} nor next action A_{t+1} again, and a negative penalty is added to the instantaneous reward obtained in that time step. The effective update of the Q matrix in this case is then given by:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \eta[R_t + \text{penalty} - Q(S_t, A_t)] \quad (4.2)$$

In each analyzed setting, the learning penalty value was set to -300. The reflection behind this choice is that this value should be large enough to influence the value of the return collected by the kite throughout its journey towards the fall. The penalty provides a negative feedback for the latest state-action pair that led to this end, which spreads through the previous states and actions during subsequent updates of the Q matrix.

In order to encourage the exploration of the environment at the beginning of each learning simulation, two techniques were employed. The first one is the optimistic initialization of the Q matrix, where the entries of the matrix were initialized to values greater than those that could actually be achieved by pulling the block at the maximum velocity. In this way, for any initial action chosen by the agent, the reward obtained is less than the expected value and this encourages the agent to change action the next time it ends up in that same state [28]. Another way to encourage exploration is to use of the ϵ -greedy approach. As explained in Chapter 2, ϵ is a quantity that ranges from 0 to 1 and regulates the amount of exploration granted to the agent: if $\epsilon = 0.1$ the agent takes one random action every 9 greedy actions, where the greedy ones are taken by choosing the action which leads to the larger available value for the current estimate of the Q matrix. To ensure initial exploration, the value of ϵ was taken larger at the beginning of each learning simulation and slowly reduced to 0 as the number of episodes increased.

After a certain number of episodes, when the estimate of the state-action value function no longer changes, the nominally optimal policy can be found by acting greedily with respect to Q , namely, by selecting in each state s the action that leads to the highest value of the Q matrix:

$$\pi(a|s) = \arg \max_{a'} Q(s, a') \quad (4.3)$$

In this way we obtain the optimal deterministic policy, that is, the one that optimizes the choices of the control angles in order to maximize the speed of the block along the rail.

4.2 Two-dimensional learning with constant wind

Let us first remind that the two-dimensional model was designed in such a way that the whole system lies in the Cartesian x/z plane. Here, the block can only travel on the horizontal rail aligned with the x axis, while the kite is forced to stay on the circumference at fixed distance R from the block. In this setting only the attack angle α was considered, while the bank angle ψ is always set to zero, therefore all the actions allowed to the agent affect α only. We choose a total of 15 available attack angles, of which two angles have only two allowed actions and the remaining 13 have three allowed actions, for a total of 43 state-action pairs.

In the first scenario a stationary wind field with velocity of 10 m/s was set on the x axis. This simple situation was already analysed in section 3.5.2, and the expected result in this setting is that the agent, after adequate exploration of the environment, is able to choose the value of the attack angle that maximizes the block velocity in a configuration where there is an equilibrium of forces.

A total of 50000 learning episodes were employed to perform the simulation in this setting. The estimate of the Q matrix was initialized optimistically with the value of 600 for each state-action couple, which means that the agent believes that the block can reach a maximum of 600 meters during his path. The initial condition of the system at the beginning of each episode places the kite at zero groundspeed and at an angle of $\theta_0 = 45^\circ$, 25 meters from the ground, while the block is sitting at the origin. The attack angle was initialized with $\alpha_0 = 12^\circ$ in each episode. The amount

of exploration ϵ allowed to the agent was initially set to 0.1, which means that the agent was acting randomly once every 9 greedy actions, and lowered to 0.05, 0.01 and 0.0001 during the training. At the end of the learning ϵ was set to 0 to get the nominally optimal policy. As for the learning rate η , its value was set to 0.1 at the beginning of the learning and lowered to 0.01 and 0.001 afterwards. The parameters utilized in the learning are summarized in table 4.2.

Parameter	Value
Episodes	50000
Episode duration	200 seconds
Learning rates η	0.1, 0.01, 0.001
Exploration ϵ	0.1, 0.05, 0.01, 0.0001, 0.
θ_0	45°
α_0	12°

Table 4.2: Parameters used in the two-dimensional learning with a constant wind of speed 10 m/s blowing along the x axis and states composed by the attack angle only: $S_t = \alpha_t$.

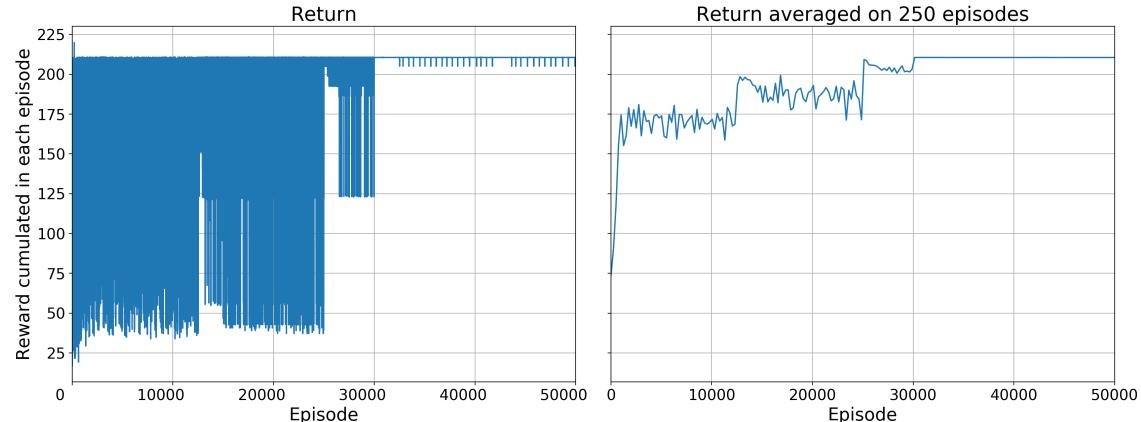


Figure 4.1: Returns of each episode (left) and averaged returns on groups of 250 episodes (right) for the learning performed on the two-dimensional system with constant wind, using the dynamic Cartesian model.

Figure 4.1 shows the return collected at the end of each learning episode and the average return on groups of 250 episodes for the fixed wind setting. The first half of episodes shows a great variance of returns because of the large amount of exploration allowed to the agent ($\epsilon = 0.1$). At the end of the learning the return settles to the value of 210.5 meters. The jumps present in the image in episodes 12500, 25000 and 30000 are due to the variations of ϵ mentioned earlier.

Figure 4.2 shows the trajectory followed by the kite and the block using the policy obtained using equation 4.3 on the final estimate of the Q matrix, and the instantaneous reward harvested by the agent (in terms of block velocity) along the same trajectory. After an initial peak, the block velocity settles to be slightly smaller than 1 m/s . This happens because there is a transient phase in which the relative velocity is bigger since the kite is still, and therefore the aerodynamic forces are

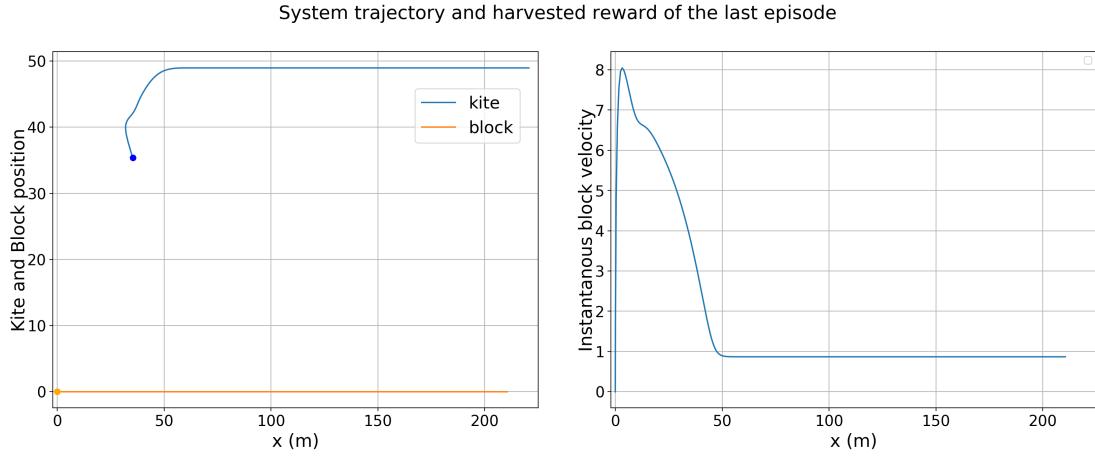


Figure 4.2: Trajectory of the kite and the block (left) and reward obtained at each time step of the trajectory (right) using a greedy policy produced by the learning in the two-dimensional system immersed in a constant wind field with speed $v_w = 10 \text{ m/s}$ on the x axis.

stronger. However, the transient condition cannot be recreated since the nominally optimal policy, as it is defined in this setting, does not allow the controller to vary the angle of attack, but allows only to find the angle of attack connected to the higher block speed in a condition of balance of forces, and to stuck on that. In the transient phase the kite moves higher than the initial position, moving from the starting angle $\theta_0 = 45^\circ$ to an equilibrium angle of $\theta_{eq} \sim 78.27$ degrees, and keeps this position for the rest of the trajectory.

As expected, the optimal policy found by the learning algorithm is to bring the angle of attack to the value of 20 degrees. This behavior reflects the results found in section 3.5.2, where we analyzed the equilibrium motion. There we found that the angle of attack of 20° was the one that provided the maximum block speed, which was, albeit slightly, greater than the one provided by the angle of 18° .

Figure 4.3 shows the final estimate of the Q matrix from which the optimal policy was drawn. Here we show the values for each entry of Q matrix, obtained both by the Cartesian and by the polar implementations. The matrix entries form a gradient that leads to increase the angle of attack up to the value of 20° , starting from almost every angle. Some values, especially among the smaller angles of attack, do not follow precisely this gradient, but this is not a problem since those angles are smaller than the starting attack angle used for the learning. By using different initial conditions this problem can be easily solved. Moreover, figure 4.3 shows that the estimates produced using the two models present consistent numbers.

In figure 4.4 the variation of the estimates of the Q matrices during the learning are shown. Here the jumps due to the decrease in the exploration ϵ at the episodes 12500, 25000 and 30000 are evident.



Figure 4.3: Estimates of the state-action value function obtained with the learning for the two-dimensional model immersed in a constant wind with speed $v_w = 10 \text{ m/s}$ on the x axis. The estimates are computed with both models: the Cartesian one (left) and the polar one (right). The matrices show the entries for each state-action couple, where the states are composed by the attack angles α and the three actions allow to “decrease”, “keep” or “increase” the current angle.

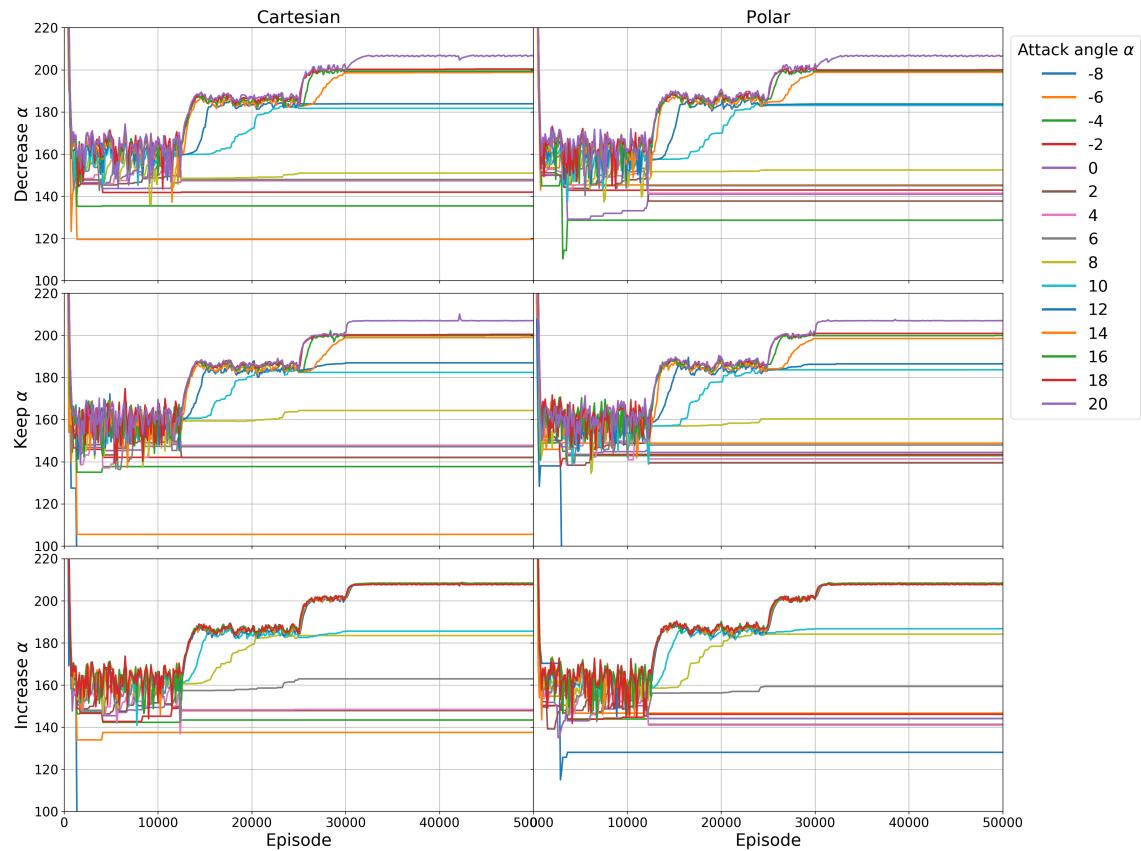


Figure 4.4: Evaluation of the Q matrix during learning for both the two-dimensional Cartesian and polar models. This figure refers to the setting with a constant wind speed of $v_w = 10 \text{ m/s}$ blowing on the x axis.

4.3 The streamfunction flow

In a completely two-dimensional model it is possible to analyze the behavior of the system while it is immersed in an incompressible two-dimensional plane flow. The description of a flow of this kind, in fact, is really simple and can be useful to study the behavior of the system in a varying wind field. In this section, which is based on Kundu's "Fluid Mechanics" [19], we quickly describe the two-dimensional incompressible flows before applying reinforcement learning on a flow of this kind.

To be incompressible a fluid has to meet certain conditions: the fluid must be stationary, the vertical scale must be small enough to avoid density variations and the flow velocity must be much smaller than the speed of sound in it. For gases, the third condition is met when the velocity of the fluid is smaller than 100 m/s. When a flow is incompressible, its density is not function of the pressure. Let us now take the continuity equation, which comes from the law of conservation of mass when applied to a small volume of fluid:

$$-\frac{1}{\rho} \frac{D\rho}{Dt} = \frac{\partial u_i}{\partial x_i}, \quad (4.4)$$

where D/Dt is called material derivative, and it emphasizes the fact that the derivative has been computed following the same fluid element during its motion. If the density of the fluid particles does not change appreciably in the path we are following, namely the fluid is incompressible, equation 4.4 becomes:

$$\frac{\partial u_i}{\partial x_i} = 0, \quad (4.5)$$

which means that the variation of the density of the flow is much smaller than the derivative of the flow velocity in every direction: $\rho^{-1} D\rho/Dt \ll \frac{\partial u_i}{\partial x_i}$.

For a completely two-dimensional flow developing in the x/z Cartesian plane the continuity equation can be written as:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial z} = 0 \quad (4.6)$$

One can now define a function $\Psi(x, z, t)$ such that:

$$\begin{aligned} u &\equiv \frac{\partial \Psi}{\partial z} \\ v &\equiv -\frac{\partial \Psi}{\partial x} \end{aligned} \quad (4.7)$$

If it obeys equation 4.7, $\Psi(x, z, t)$ is called streamfunction [19]. It is then clear that any streamfunction that obeys 4.7 respects the properties of a two-dimensional incompressible flow, and it can be then used to define it.

Let us now remind that the streamlines of the flow are the lines tangent in every point to the velocity direction of the flow. For unsteady flows they change with time. Given a small arc of streamline $d\mathbf{s} = (dx, dz)$ and the vector representing the flow velocity in that arc $\mathbf{u} = (u, v)$, from

the definition of streamline we have that the vector product $d\mathbf{s} \times \mathbf{u} = \mathbf{0}$, being the two parallel to each other. From this definition of streamline we can write:

$$\frac{dx}{u} = \frac{dz}{v} \quad (4.8)$$

Using now 4.7 together with 4.8 one can find that:

$$d\Psi = \frac{\partial\Psi}{\partial x}dx + \frac{\partial\Psi}{\partial z}dz = 0, \quad (4.9)$$

which means that the value of the streamfunction Ψ is constant along a streamline. Hence its name.

Let us now choose a simple streamfunction to create an environment in which to perform the two-dimensional learning. An example can be:

$$\Psi(x, z) = k \frac{z^2}{2} \left[1 + \epsilon \sin\left(\frac{\pi x}{L_x}\right) \sin\left(\frac{\pi z}{L_z}\right) \right] \quad (4.10)$$

Here k , ϵ , L_x and L_z are parameters to select, while x and z are the coordinates of the point in which we are computing the wind velocity, namely the kite position. Here the term $k \frac{z^2}{2}$ gives a linear increment of the wind velocity moving up on the z axis, while the trigonometric functions represent oscillations of the wind velocity on the x and z axis, repeating with periods π/L_x and π/L_z . By calculating the derivatives of $\Psi(x, z)$ one can find the expression for the wind velocity respectively on the x and z axis:

$$u = \frac{\partial\Psi}{\partial z} = \frac{1}{2} kz \left[2 + 2\epsilon \sin\left(\frac{\pi x}{L_x}\right) \sin\left(\frac{\pi z}{L_z}\right) + \frac{\epsilon\pi z}{L_z} \sin\left(\frac{\pi x}{L_x}\right) \cos\left(\frac{\pi z}{L_z}\right) \right] \quad (4.11)$$

$$v = -\frac{\partial\Psi}{\partial x} = -\frac{\epsilon\pi kz^2}{2L_z} \cos\left(\frac{\pi x}{L_x}\right) \sin\left(\frac{\pi z}{L_z}\right) \quad (4.12)$$

These expressions for the wind velocity were used to perform a learning simulation with the SARSA algorithm for the two-dimensional system. The parameters were set to:

$$k = 0.5, \quad \epsilon = 0.2, \quad L_x = 50, \quad L_z = 50 \quad (4.13)$$

With this choice of parameters, the wind velocity on the x axis goes from 0 on the ground to 30 m/s at the maximum height of the kite which is imposed by the tether length $R = 50$ m, while the velocity on the z axis varies in the range ± 2 m/s. Figure 4.5 gives a representation of the streamfunction defined in 4.10 using the parameters defined in 4.13.

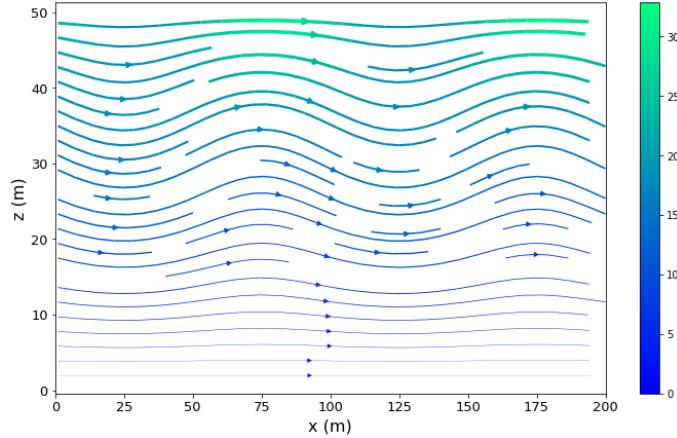


Figure 4.5: Streamfunction defined in equation 4.10 using the parameters defined in 4.13.

4.3.1 Learning in the streamfunction flow

To start the learning in the streamfunction flow, we considered an agent that knows only its attack angle as in the constant wind setting. This means that the agent does not receive any information about the flow. Therefore, in this setup it is necessary to estimate 43 state-action pairs for the Q matrix, as in the previous case. Table 4.3 summarizes the parameters used for this simulation.

Parameter	Value
Episodes	50000
Episode duration	200 seconds
Learning rates η	0.6, 0.1
Exploration ϵ	$10^{-3}, 10^{-4}, 10^{-5}$
θ_0	45°
α_0	12°

Table 4.3: Parameters used in the two-dimensional learning with streamfunction wind field and states composed by the attack angle only: $S_t = \alpha_t$.

In this setting the simulation was performed using the Cartesian model only, since in the Cartesian implementation the integration of the dynamics is simpler than the polar one, and therefore faster to compute. Figure 4.6 shows the return collected at the end of each learning episode and the average return for the learning in the streamfunction wind environment. The ideal attack angle that allows the kite to pull the block to the maximum distance is found quite early in the learning. The big jump in the averaged return graph after 15000 episodes is due to the decrease in the exploration ϵ from 10^{-3} to 10^{-4} .

As in the constant wind case, the optimal policy in the streamfunction flow is to bring the attack angle to 20 degrees, as figure 4.7 shows. With this policy the agent is able to pull the block for 1175.3 meters in a total of 89.3 seconds from the start of the episode. After this point the kite gets stuck, remaining suspended in the air in a situation of balance of forces, and the reward drops

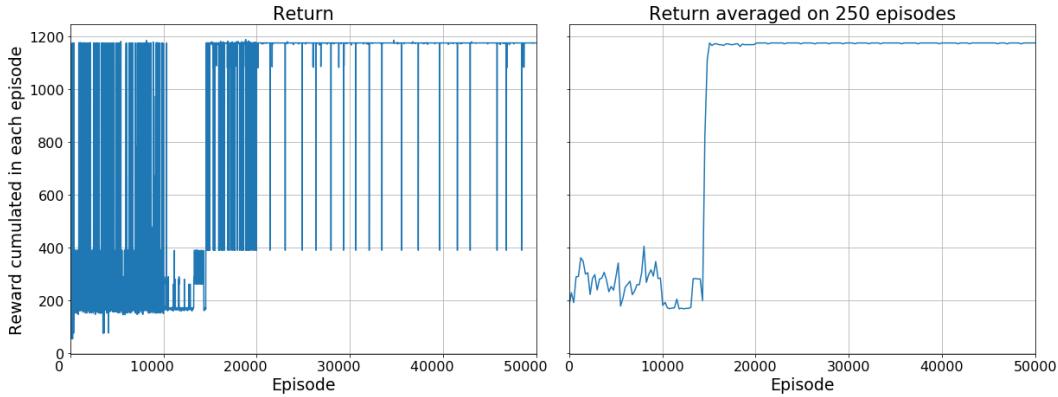


Figure 4.6: Returns of each episode (left) and averaged returns on groups of 250 episodes (right) for the learning performed on the two-dimensional system immersed in the streamfunction wind field, using the dynamic Cartesian model and only the attack angles as states.

to zero. This happens because the motion of the kite is largely driven by the acceleration given by the aerodynamic forces generated in the initial transient, when the relative velocity is higher. However, the direction and magnitude of the lift and drag forces generated by the use of a fixed attack angle are not favourable to an ideally permanent motion, in the streamfunction setting. The aerodynamic forces are the only ones that allow the block to move by overcoming the friction, but in the constant attack angle setting those are not strong enough to allow the motion. Indeed, after a while the kite gets stuck.

Figure 4.8 shows the evolution of both the speeds of the kite and the block when moving into the streamfunction flow using the optimal policy. Both velocities show an oscillating trend that reflects the underlying shape of the periodic flow, decreasing to zero along the path.

From the above findings, it can be concluded that the initial condition regarding the attack angle α_0 and the angle with the horizontal θ_0 can modify positively or negatively the maximum distance traveled by the block in this setting. Regardless of this, the system will eventually find itself stuck in a balance of forces, since in this way one can only influence the transient phase. Starting from $\theta_0 = 45^\circ$ with zero groundspeed for the kite and varying the initial angle of attack, the block reaches the maximum traveled distance by setting the initial attack angle to 14, 16, 18 or 20 degrees. With this choice the block travels for a maximum of 1173.9 m. Instead, lowering θ_0 the length of the path can be increased up to the value of ~ 1276.9 m, achieved with $\theta_0 = \pi/12$ and initial attack angle of $\alpha_0 = 14^\circ$.

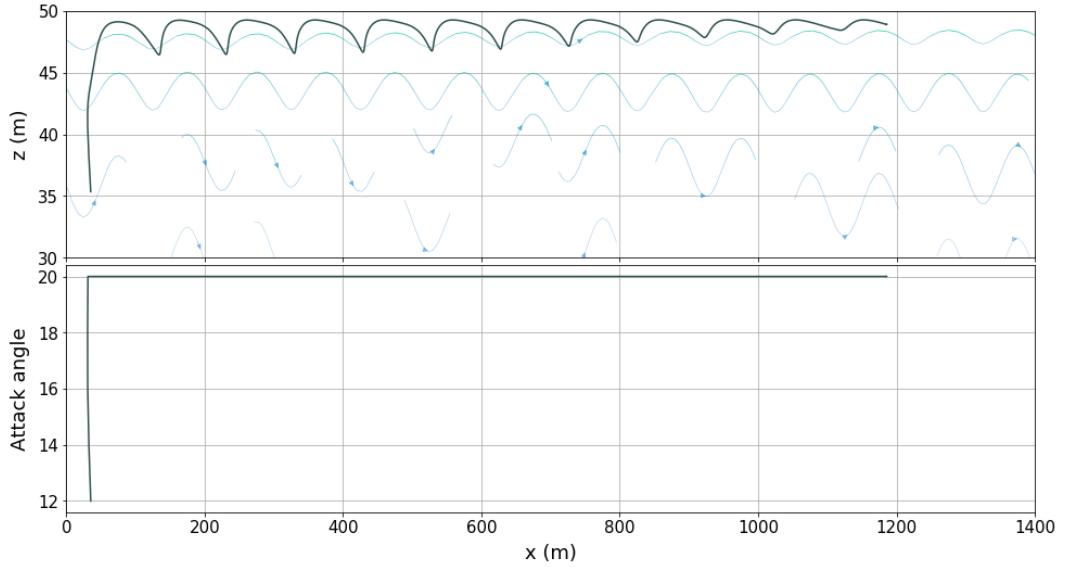


Figure 4.7: Optimal trajectory of the kite (top) and attack angle selected during the trajectory (bottom) for the streamfunction wind field where the state of the system is composed only by the attack angle.

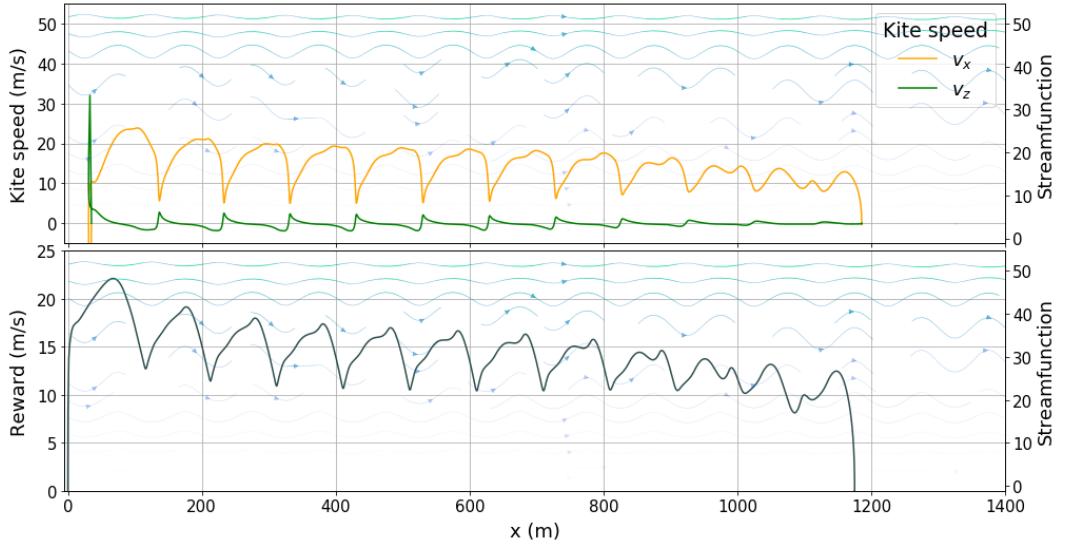


Figure 4.8: On the left axes are shown the kite speed (top) and block speed or instantaneous reward (bottom) obtained in the streamfunction wind field by using the optimal policy. The right axes shows the underlying streamfunction shape in the first 50 meters of height. The evolution of the speeds of the kite and the block reflects the underlying shape of the streamfunction.

4.3.2 Augmenting two-dimensional learning

In the preceding learning we noticed how the agent was able to move the kite in the streamfunction flow without it falling to the ground and pulling the block to the maximum possible distance, knowing only the angle of attack and having the possibility to change it. However, after some time the kite got stuck in a balance of forces and could not move anymore. Allowing the agent to use different attack angles α in different conditions we could make the kite move in the streamfunction flow without getting stuck after exploiting the initial acceleration provided by the transient.

During its motion inside the streamfunction flow the kite encounters a varying wind, since the wind speed on the x and z axis varies in space and therefore during the kite motion. Let us recall the definition of relative velocity between the kite and the wind as $\mathbf{v}_{rel} = \mathbf{v}_k - \mathbf{v}_w$, where \mathbf{v}_k is the kite speed and \mathbf{v}_w the wind speed. Since this quantity can be measured directly on the kite by means of on-board anemometers, it can be profitably utilized by the agent to increase its knowledge of the environment and make better choices.

To insert this knowledge into the learning scheme it is necessary to augment the states of the Markov decision process. To enter this information in the simplest way, the states were augmented by inserting the value of the angle β between the horizontal and the relative speed, which is represented in figure 4.9.

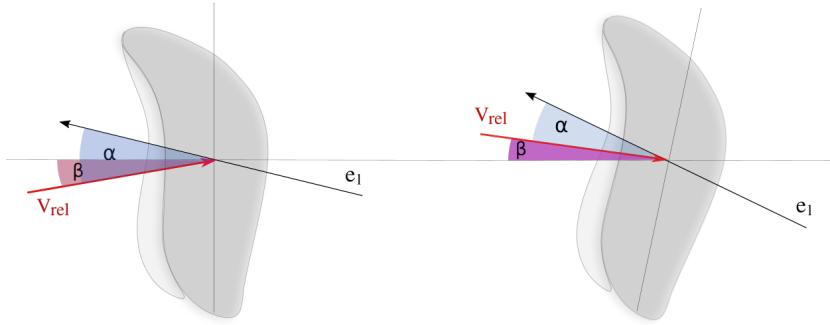


Figure 4.9: Angle β between the horizontal and the relative speed

In this way we are providing the agent with information on the relative speed inclination with respect to the ground, by inserting an additional dimension within the state-action value function. The state of the agent at the time step t is now defined as a couple made by the value of the currently selected attack angle α_t together with the value of the angle between the relative velocity and the horizontal β_t :

$$S_t = (\alpha_t, \beta_t) \quad (4.14)$$

Since the relative velocity between the kite and the wind is a continuous quantity, so is β . Therefore, the easier way to insert it into the finite set of states is to discretize its domain into intervals, and check at every learning step in which of those intervals the true value of β falls. The discretization works as following: given an array of values, if at the current time step β_t is bigger than the i -th value of the array and smaller than the $(i + 1)$ -th, the state of the angle belongs

to the i -th interval. After some trials we noticed that the value of β was always contained in the range going from -3.2° to 3.2° . This range was subdivided into 26 unevenly distributed intervals, as it was noticed that a denser discretization was required at the edges of the range for the agent to take better actions. The selected intervals are: $-3.2, -3.1, -3.0, -2.9, -2.7, -2.4, -2.1, -1.8, -1.5, -1.2, -0.9, -0.6, -0.3, 0., 0.3, 0.6, 0.9, 1.2, 1.5, 1.8, 2.1, 2.4, 2.7, 2.9, 3.0, 3.1$ and 3.2 degrees. Once augmented, the Q matrix counts a total of approximately 1170 state-action pairs.

SARSA was applied on the present setting for a total of 50000 learning episodes. The parameters employed in this simulation are listed in table 4.4.

Parameter	Value
Episodes	50000
Episode duration	200 seconds
Learning rates η	0.5, 0.1, 0.01
Exploration ϵ	0.05, 0.01, 0.001, 0.0001, 0.
θ_0	45°
α_0	12°

Table 4.4: Parameters used in the two-dimensional learning with streamfunction wind field and states composed by the attack angle α and the angle β between the relative velocity and the horizontal: $S_t = (\alpha_t, \beta_t)$.

Figure 4.10 shows the trend of the return harvested by the agent during the learning, and the average return growth. From the image it is clear that now the exploration phase needs more episodes in order to find the actions that lead to the optimal return, which is achieved for the first time after approximately 15000 episodes. The average growth of the returns shows a sharp peak of growth around 40000 episodes, which coincides to the point where the exploration was moved from one random action over 1000 ($\epsilon = 0.001$) to one random action over 10000 ($\epsilon = 0.0001$).

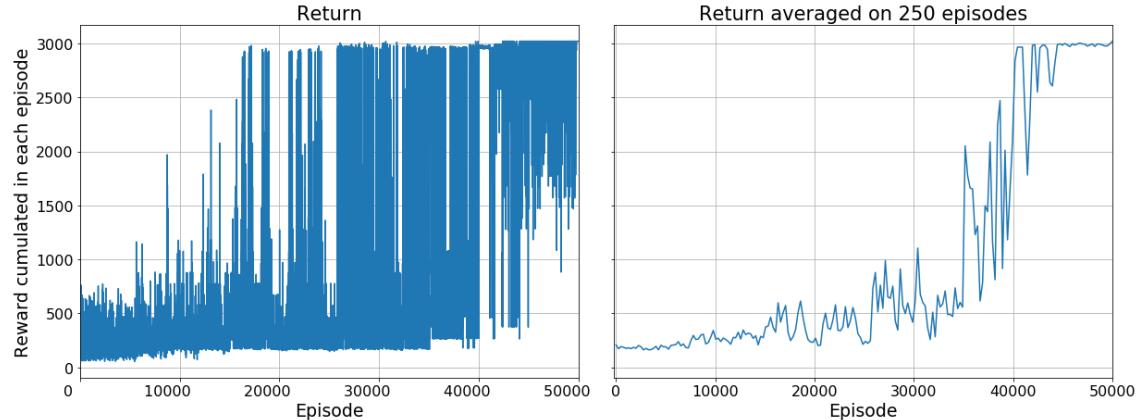


Figure 4.10: Returns of each episode (left) and averaged returns on groups of 250 episodes (right) for the learning performed on the two-dimensional system immersed in the streamfunction wind field, using the dynamic Cartesian model.

At the end of the learning the return stabilizes to the value of 3020.01 m . The road covered by the block in this setting is more than twice the one obtained in the setting with a fixed angle of attack. This happens because, as expected, the agent learns to use different attack angles in different states: most of the time the angle of attack is kept at 20 degrees, but its value is periodically decreased. Figure 4.11 shows the final behavior of the agent, namely the choice of the attack angle as a function of the value of β . As can be seen the value of β jumps abruptly from about 3 to -3 and vice versa, apart from some oscillations from about -3 to -1.5. The angles between -1.5 and 2.5 are never visited using the final policy. When the angle between the relative velocity and the horizontal is negative, close to -3, usually the attack angle is kept to the value of 20; but when β becomes positive and close to 3, the attack angle is sometimes decreased to 18 degrees, with some peaks reaching 16 and 14 degrees. Moreover, from figure 4.11 it is clear that the timing of the agent's actions reflects the periodicity of the streamfunction. Following this policy, as figure 4.12 shows, both the agent and the block are able to keep a stationary periodic velocity, which allows the whole system to avoid getting stuck in a balance of forces. In this ideal condition, indeed, the kite could keep pulling the block forever.

Figure 4.13 shows the final optimal policy used by the agent in the streamfunction wind field. This policy is not homogeneous, and presents big variations of choices among near values of β .

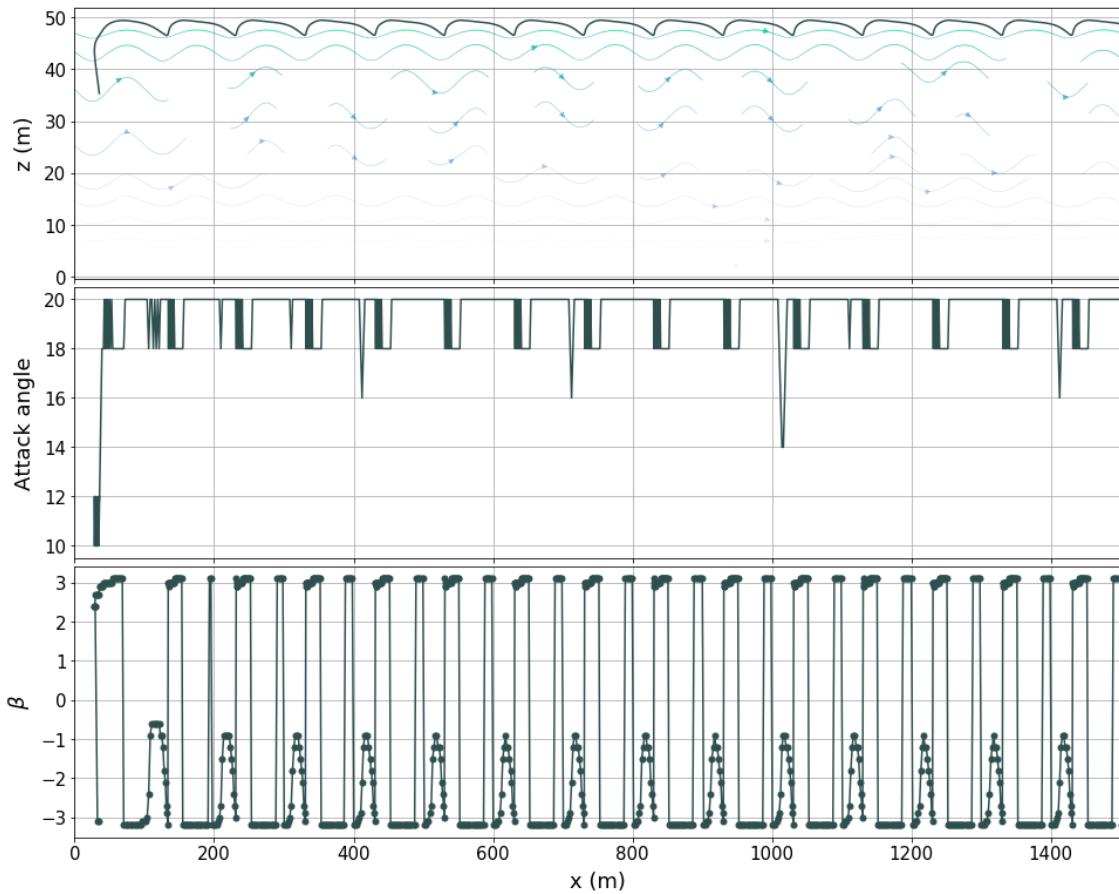


Figure 4.11: Trajectory followed by the kite (top), chosen attack angle α (middle) and measured value of the angle β between the horizontal and the relative velocity (bottom) for the kite-block system moving in the first 1500 meters of the streamfunction field.

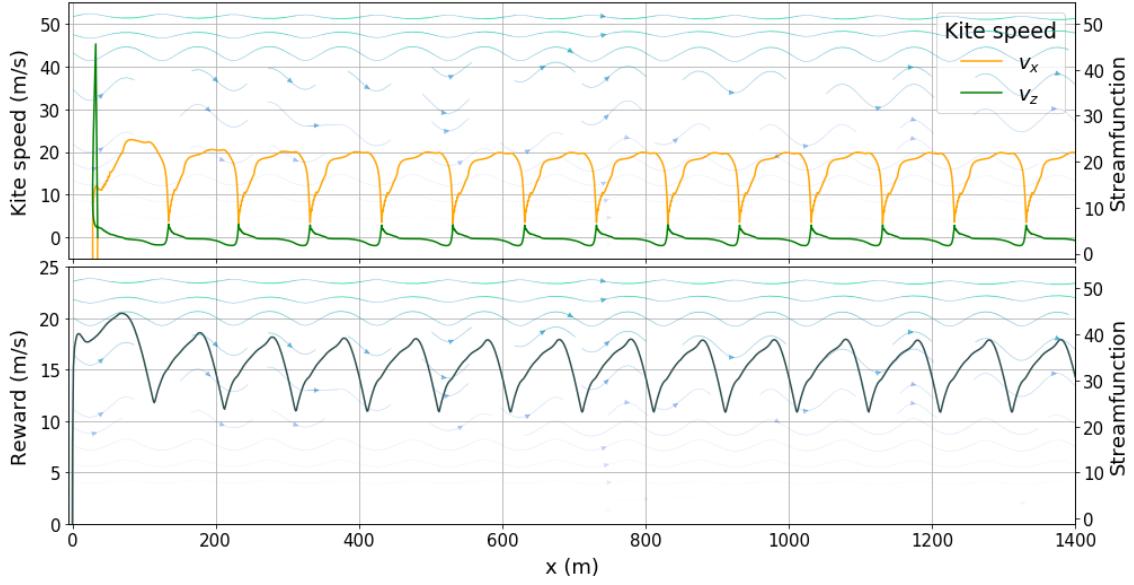


Figure 4.12: On the top, kite velocity on the x and z axes. On the bottom, instantaneous return harvested by the agent (block velocity on the horizontal) for the kite-block system moving in the first 1500 meters of the streamfunction field.

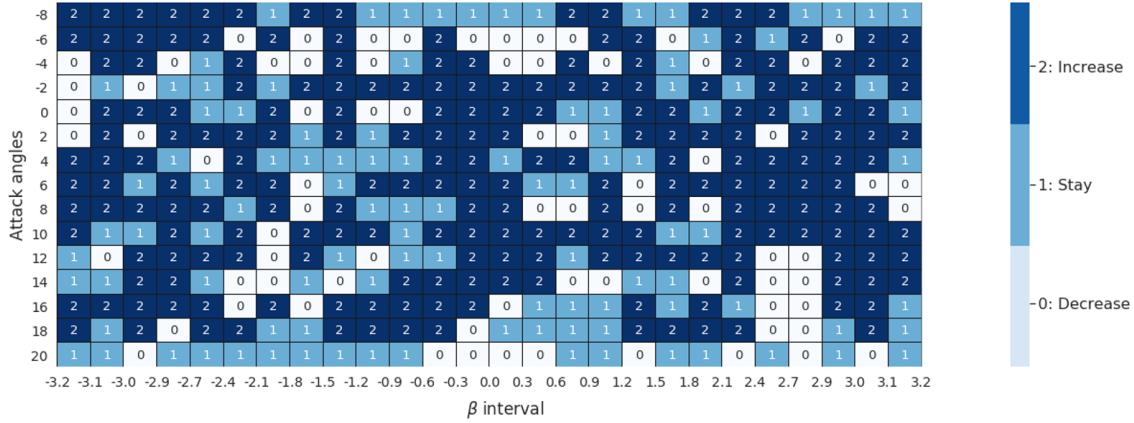


Figure 4.13: Policy learned by the agent for the kite moving in the streamfunction wind field, allowing the agent to know the angle β between the horizontal and the relative velocity.

4.4 Three-dimensional learning with constant wind

The three-dimensional setting, as explained in section 3.4.2, was modeled in a Cartesian space where the block lies on the x/y plane and it is able to move on a rail aligned with the x axis while the kite is free to move on the sphere at fixed distance R from the block, and exploits the aerodynamic forces to pull the block along the rail. In this setting both the attack angle α and bank angle ψ are accounted for, and it is assumed that the agent is always able to measure and adjust the two angles within a set of allowed values. Therefore, both angles need to be components of the states of the Markov decision process. In the simplest case the new state at time t is defined as the couple made by the values of the attack angle α_t and the bank angle ψ_t at that time step:

$$S_t = (\alpha_t, \psi_t). \quad (4.15)$$

The discrete range of attack angles used in this setting is defined in section 3.5.1, and is the same as that used for the two-dimensional model. For the bank angle a narrow set of fixed values was selected, namely: -10, -8, -4, -6, -2, -1, 0, 1, 2, 4, 6, 8 and 10 degrees. We remind that, for both angles, the actions are defined as the possibilities to increase or decrease the current value of the angle, or keep it the same. Even in the simpler case, the size of the Q matrix we need to evaluate using SARSA is now significantly bigger than the one of the two-dimensional model. Indeed, having three actions for almost each value of α and ψ , a total of approximately 1755 state-action pairs need to be estimated, with respect to the 43 states-action pairs of the two-dimensional setting, where ψ was always set to zero.

The system was then immersed into a constant wind field with speed of 10 m/s on the x axis, as done in the two-dimensional case, and trained for a total of 100000 episodes. In this setup the duration of each episode has been lengthened to 1000 seconds, as it was observed that 200 seconds were not enough to allow the agent to adequately learn the optimal policy. The values selected for the learning rate η and the exploration ϵ have been changed too: the initial learning rate was set to 0.6 and decreased during the learning to 0.1 and 0.01, while the rate of exploration was kept smaller, initialized to 0.0001 and moved down to zero during the training. The parameters employed in this setting are summarized in table 4.5.

Parameter	Value
Episodes	100000
Episode duration	1000 seconds
Learning rates η	0.6, 0.1, 0.01
Exploration ϵ	0.001, 0.0001, 0.0001, 0.
θ_0	45°
ϕ_0	0
α_0	12°

Table 4.5: Parameters used in the three-dimensional learning with a constant wind of speed 10 m/s blowing along the x axis and states composed by the bank and attack angles: $S_t = (\alpha_t, \psi_t)$.

The optimal policy produced by the learning process is to set the bank angle to 0 and the attack angle to 20° and keep these same values for the whole episode. In this setting the agent could learn non-constant trajectories, where it would need to vary both the attack and bank angle in a periodic

setting. However, from the learning results seems that the most convenient choice is again to keep a constant value for the control angles.

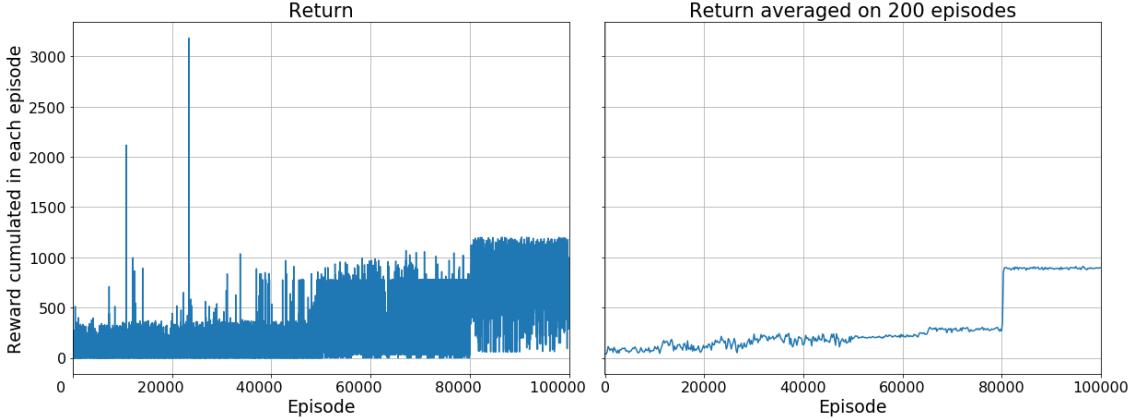


Figure 4.14: Returns of each episode (left) and averaged returns on groups of 100 episodes (right) for the learning performed on the three-dimensional system immersed in a constant wind field with speed 10 m/s on the x axis.

With the optimal policy the agent is able to pull the block for the whole episode, covering a total of 905.02 meters in 1000 seconds. However, as figure 4.14 shows, some extremely high peaks of return are happening during the training, whose values are way bigger than the total return the agent gets by moving with the final policy. The sequence of actions leading to those peaks is not preserved during the learning, meaning that the learning was not long enough to allow the agent learn this policy, or that it is not possible to learn this behavior relying only on the given states. Arguably, these large returns were obtained thanks to the exploration provided to the agent who, being several times in the same state, acted first in one way and then in another. This behavior is not allowed by the final deterministic policy we are searching for, but these high returns are telling us that with the right choice of the states and with a fairly long training, the kite could be able to pull the block up to that distance.

4.4.1 Augmenting three-dimensional learning

For the agent to reach better returns we decided to allow him to behave differently under different conditions. Therefore, as for the two-dimensional case, the states of the Markov decision process have been augmented. The new state at time t has been defined as a tuple made by the attack angle α_t , the bank angle ψ_t , and the angle β_t between the horizontal and the projection of the relative velocity on the x/z plane:

$$S_t = (\alpha_t, \psi_t, \beta_t) \quad (4.16)$$

The angle β has been discretized into the same 26 states used for the two-dimensional case with the streamfunction wind field. This time the Q matrix is much larger, counting a total of roughly 45.630 states. A total of 150000 learning episodes has been used to train the kite in this setting. The learning rates η employed for the problem varied from 0.7 to 0.01 during learning,

while the amount of exploration ϵ from 0.0001 to 0. All the learning parameters used in the present simulation are listed in table 4.6.

Parameter	Value
Episodes	150000
Episode duration	1000 seconds
Learning rates η	0.7, 0.1, 0.01
Exploration ϵ	0.0001, 0.00001, 0.00001, 0.
θ_0	45°
ϕ_0	0
α_0	12°

Table 4.6: Parameters used in the three-dimensional learning with a constant wind of 10 m/s blowing along the x axis and states composed by the attack angle α , the bank angle ψ and the angle β between the horizontal and the projection of the relative velocity on the x/z plane: $S_t = (\alpha_t, \psi_t, \beta_t)$.

Figure 4.15 shows the trend of the returns as the number of the episode increases, together with their average growth. It is clear that augmenting the states of the Q matrix the agent is capable to find a policy that allows him to obtain remarkably better returns at the end of the learning, 6 times as much as in the previous case. Indeed, as figure 4.16 shows, the instantaneous reward provided by the optimal deterministic policy computed using equation 4.3 is constantly higher than the one collected using a fixed control angles, even if it is more variable. In figure 4.17 the corresponding choices for the attack and bank angles are shown. With the new setting for the states, instead of fixing the angle of attack at 20 degrees, the agent prefers to adopt variable values ranging from 8 to 14 degrees. The same applies to the bank angle, whose values fluctuate between 0 and 6 degrees.

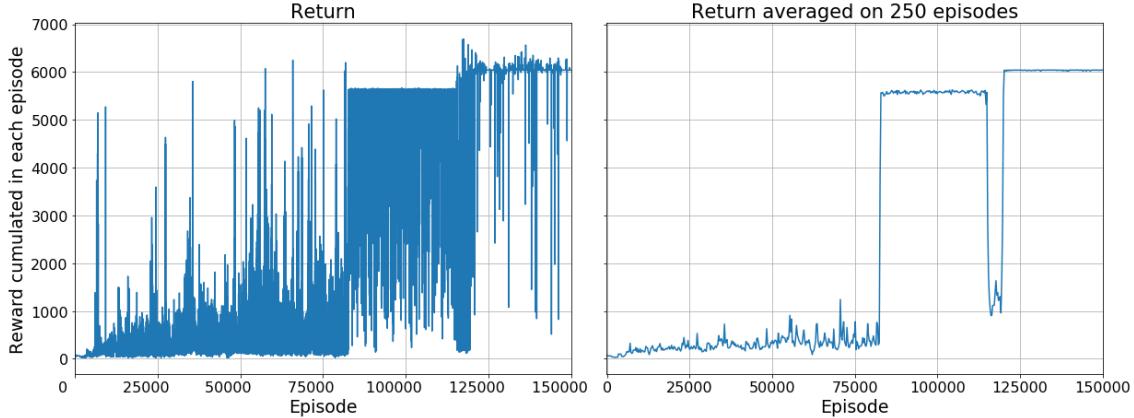


Figure 4.15: Returns of each episode (left) and averaged returns on groups of 250 episodes (right) for the learning performed on the three-dimensional system immersed in a constant wind field with speed 10 m/s blowing on the x axis.

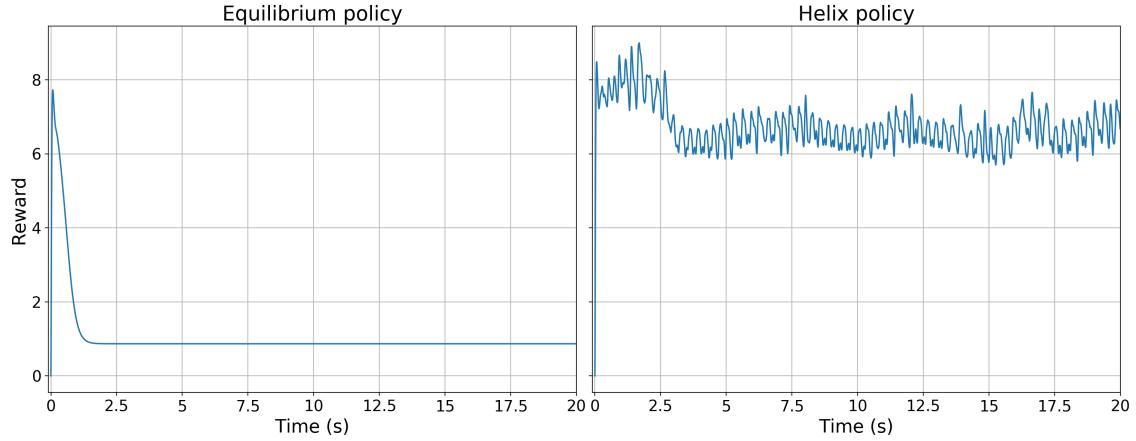


Figure 4.16: Comparison of the instantaneous rewards harvested in the first 20 seconds using the policy learned with states $S_t = (\alpha_t, \psi_t)$ (left image), and with states $S_t = (\alpha_t, \psi_t, \beta_t)$ (right image), where β is the angle between the horizontal and the projection of the relative velocity on the x/z plane.

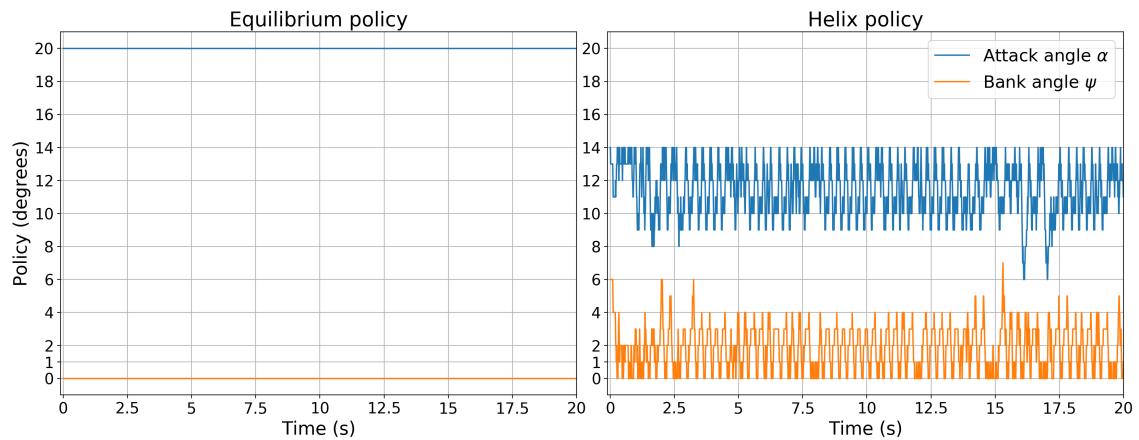


Figure 4.17: Comparison of the control angles employed in the first 20 seconds using the policy learned with states $S_t = (\alpha_t, \psi_t)$ (left image), and with states $S_t = (\alpha_t, \psi_t, \beta_t)$ (right image), where β is the angle between the horizontal and the projection of the relative velocity on the x/z plane.

The resulting policy is difficult to visualize given the high number of states of the matrix, but it can be said with enough certainty that it is not yet converged to the optimal one, since as figures 4.16 and 4.17 show that the agent's behavior is still very fluctuating without showing a precise periodicity (even in the long run) while the addressed problem is stationary. Therefore this exploratory learning cloud be improved even more by performing a more accurate choice of the state values.

Figures 4.18 and 4.19 show the trajectory followed by the kite by using the final policy. As can be seen, the oscillations of the control angles in the constant wind environment lead the kite to perform an helix-shaped motion. Although this movement may be more difficult to control since it requires a constant change of the control angles, the helix policy allows the block to travel an incredibly greater amount of road with respect to the equilibrium policy: in 1000 seconds the block covers 6590.04 meters, an improvement of a factor of six.

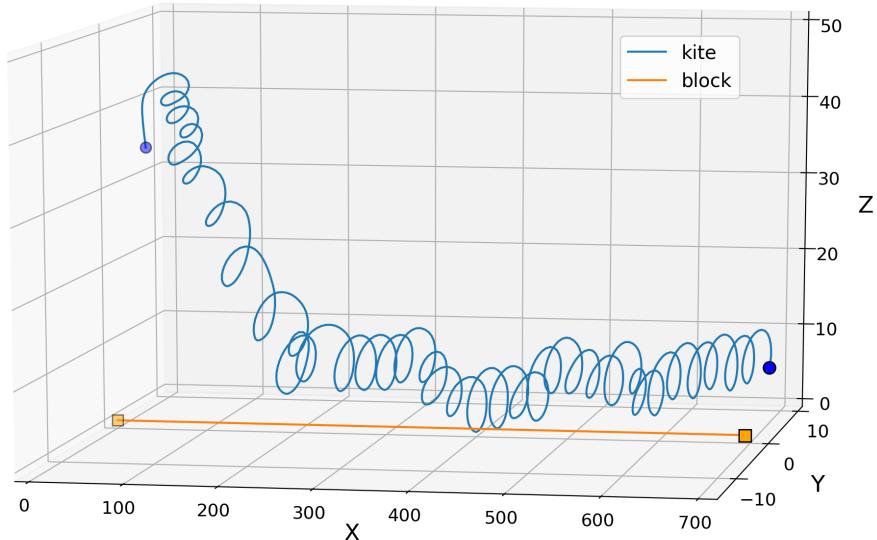


Figure 4.18: Trajectory followed in the first 101 seconds by the three-dimensional kite-block system using the policy derived from the application of the SARSA learning algorithm to the model. The initial position of the kite and the block is identified by the transparent symbols, and sees the block at the origin of the axis, while the kite is at 25 meters from the ground with $\theta_0 = 45^\circ$ and $\phi_0 = 0$. The position after 101 seconds is given by the opaque symbols. The motion of the kite takes the form of an helix, while the block moves along the rail with a mean velocity of 6.58 m/s on the positive direction of the x axis.

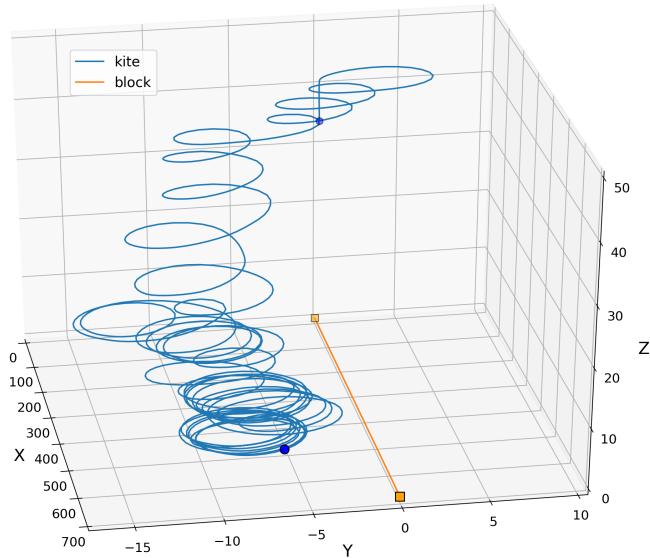
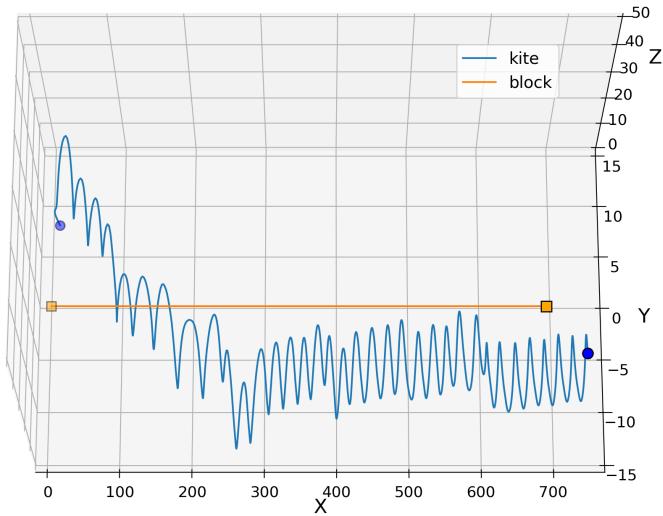


Figure 4.19: View from above (on the top) and view from the front (bottom) of the first 101 seconds of motion of the three-dimensional kite-block system. The initial position of the kite and the block is identified by the transparent symbols, and sees the block at the origin of the axis, while the kite is at 25 meters from the ground, with $\theta_0 = 45^\circ$ and $\phi_0 = 0$. The position after 101 seconds is given by the opaque symbols. The motion of the kite takes the form of an helix, while the block moves along the rail with a mean velocity of 6.58 m/s in the positive direction of the x axis.

By analyzing the behavior of the kite flying with the helix strategy it can be clearly understood why this type of motion is so effective. The helix shape allows the instantaneous speed of the kite to be much more perpendicular to the wind than if it was constantly moving along the x axis. As explained in Chapter 1, section 1.2.1, this kind of motion is called *crosswind*, and it is known to be the most effective to maximize the power produced by an airborne wind energy device, and thus to maximize the speed of a towed object. Indeed, the more the kite motion is crosswind, the more the total aerodynamic force acting on the airborne device becomes parallel to the wind direction. In an ideal crosswind motion – impracticable in most cases, including this – the aerodynamic force can be totally converted into tether tension and consequently exploited to pull the block. In the real motion the cable needs to form a non-zero angle with the ground to allow the flight, and this prevents the motion to be completely crosswind and thus the fraction of usable aerodynamic force is reduced. Nevertheless, even in the non-ideal case the crosswind motion is preferable compared to having a kite flying parallel to the wind.

Figures 4.20 and 4.21 allow to see the difference between the two types of motion. The first image represents the constant motion with fixed attack and bank angles, where the kite travels keeping a distance from the ground of 48.95 meters, very close to the maximum height of 50 meters set by the cable length. The green vector represents the wind speed, which measures 10 m/s and is directed along the x axis, while the red one represents the scaled vector given by the sum of aerodynamic lift and drag. As explained in Chapter 1, section 1.2.2, the component of the aerodynamic force that actually contributes to pull the block is the one projected along the wind velocity direction, which is given by $F_{aer} \cos(\chi)$, where χ is the angle between the aerodynamic force and the wind direction. In this setting the aerodynamic force forms an angle of 78.68° with the wind, and has a fixed value of 273.45 N. The aerodynamic force which actually contributes to pull the block is then given by: $F_{aer} \cos(\chi) = 273.45 \cdot \cos(78.68) = 53.62 \text{ N}$.

The second image (figure 4.21) shows the kite in crosswind motion, where the attack and bank angles oscillate and the kite moves in an helix-shaped trajectory. In this case the angle between the aerodynamic force and the direction of the wind oscillates between 10 and 21 degrees, with a mean value of 18.33, averaged on the whole trajectory. The modulus of the aerodynamic force oscillates between 100 and 250 N, with an average value of 153.03 N. As a result, the amount of aerodynamic force that actually contributes to the block motion is $153.03 \cdot \cos(18.33) = 145.26 \text{ N}$, which is almost three times larger than the previous value. We can compute the power extracted from the kite-block system in the two cases. In the first case the kite is moving at constant velocity, meaning that the horizontal component of the aerodynamic force equals the friction force, therefore $F^\mu = 53.62 \text{ N}$ along the whole trajectory. In this setting the block travels for a total of 905.02 m in 1000 seconds. The extracted power results:

$$P_{extracted} = \frac{53.62 \text{ N} \cdot 905.02 \text{ m}}{1000 \text{ s}} = 48.53 \text{ W} \quad (4.17)$$

In the second setting the block is not traveling with constant velocity and the system is not in equilibrium. The friction force in this case varies in the whole trajectory, and has a mean value of 144.74 N. Here the block travels for a total of 6590.04 m in 1000 seconds. Therefore the extracted power results:

$$P_{extracted} = \frac{144.74 \text{ N} \cdot 6590.04 \text{ m}}{1000 \text{ s}} = 953.84 \text{ W} \quad (4.18)$$

Which is 20 times the previous value.

If we want to extract usable power from this device, for example by replacing the block with a generator, we could estimate the maximum usable power using the formula extrapolated by Diehl, described in Chapter 1, section 1.2.2, where we have:

$$P_{max} = \frac{2}{27} \rho A v_w^3 C_R \left(\frac{C_R}{C_D} \right)^2, \quad (4.19)$$

where $C_R = \sqrt{C_L^2 + C_D^2}$.

In the scenario with constant motion the agent adjusts the angle of attack on the value of 20° , whose relative coefficients of lift and drag read $C_L = 1.05$ and $C_D = 0.21$. Computing equation 4.19 the maximum extractable power results: $P_{max} = 12.60$ kW. In the helix-shape scenario, the attack angle oscillates between 8 and 14 degrees. The mean value for the lift and drag coefficients in this framework is $\hat{C}_L = 0.86$, $\hat{C}_D = 0.09$, and maximum extractable power results $P_{max} = 35.63$ kW. Since we know the angle between the aerodynamic forces and the wind we can use the latter to refine formula 4.19, obtaining:

$$P_{max} = \frac{2}{27} \rho A v_w^3 C_R \left(\frac{C_R}{C_D} \right)^2 \cos(\chi)^3 \quad (4.20)$$

With the new formula, the computed values for the power arise to be respectively 95.31 W and 30.47 kW respectively, where the power in the first setting is reduced by a factor of 100. From this simplified analysis it is evident that the control of the trajectory followed by the airborne device is important not only to prevent the kite to fall, but also to improve the amount of extractable power.

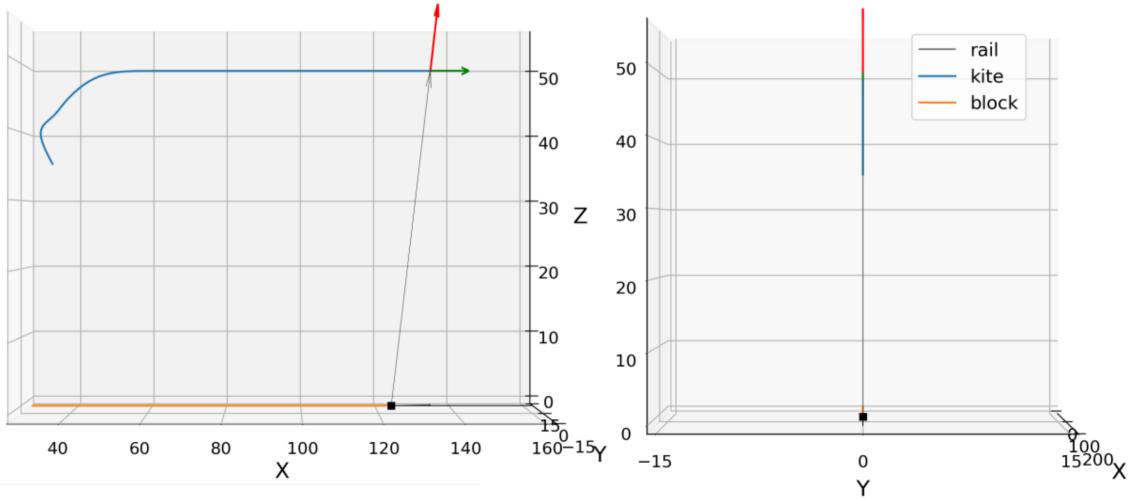


Figure 4.20: Motion given by the policy with fixed attack angle (20°) and fixed bank angle (0°). The green vector represents the wind speed of 10 m/s directed along the x axis. The red vector gives the direction of the aerodynamic forces acting on the kite. The aerodynamic forces are all contained in the x/y plane and form an angle of ~ 78 degrees with respect to the wind direction.

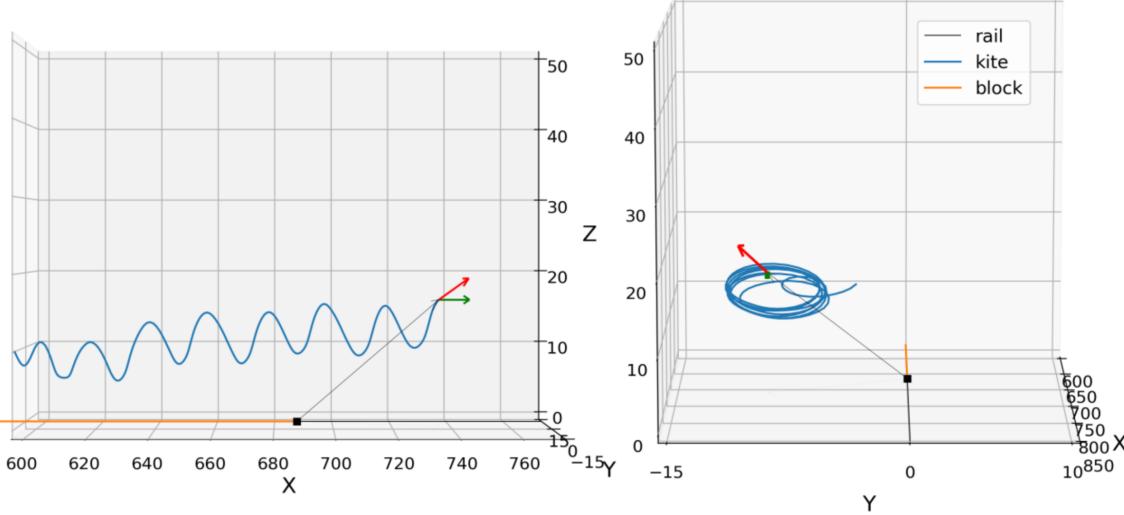


Figure 4.21: Motion given by the helix policy with oscillating attack and bank angles. The green vector represents the wind speed of 10 m/s directed along the x axis. The red vector gives the direction of the aerodynamic forces acting on the kite. The aerodynamic forces direction is variable, and forms an angle between 10 and 21 degrees with respect to the wind direction.

Chapter 5

Perspectives

In this thesis we have shown how it is possible to use reinforcement learning to optimize the power extraction by an airborne wind energy system. The proposed model, developed in Chapter 3, consists of a block connected to a kite by means of a fixed-length tether. The block is allowed to move along a straight rail on the ground, aligned with the main wind direction. The kite exploits the aerodynamic forces with the aim of maximizing the block velocity along the rail. We employed the SARSA algorithm to let the kite learn by trial and error which are the best actions to take in order to maximize the block velocity. In Chapter 4 we showed that the final policy obtained from the learning approximates the nearly-optimal crosswind motion by moving in an helix-shaped trajectory.

Improvement of the present work can be sought in many directions. A variable-length tether could be inserted, allowing to study the system while exploiting the pumping mode. A more realistic model of the tether can be developed, where the tether is allowed to bend and its drag is accounted for. Moreover, the block could be replaced by a fixed or moving ground station for power extraction.

Currently, we are moving in the direction of considering more realistic atmospheric flow environments. The kite-block system will be trained in a wind with varying speed, which will be realized by implementing a Large Eddy Simulation (LES) of the turbulent Couette channel flow. The channel measures 50 meters on the z axis and 100 meters in both the periodic x and y dimensions. The simulation is carried out using the open source software Nek5000, and produces a turbulent flow with mean velocity of 10 m/s along the x axis. The turbulent flow simulation generates a total of 10000 frames of data containing the wind speed along the three directions for the whole channel. The frames are at most 0.02 seconds apart, for a total of 200 seconds of simulation. The system composed by the kite and the block will be immersed in this flow and trained with the SARSA algorithm. This will allow us to understand how nearly-optimal strategies can cope with the strong fluctuations induced by atmospheric turbulence and suggest effective controls for field applications.

Bibliography

- [1] Uwe Ahrens, Moritz Diehl, and Roland Schmehl. *Airborne Wind Energy*. Green Energy and Technology. Springer, 2013. DOI: [10.1007/978-3-642-39965-7](https://doi.org/10.1007/978-3-642-39965-7).
- [2] Cristina Archer. “An Introduction to Meteorology for Airborne Wind Energy”. In: *Green Energy and Technology* (Oct. 2013), pp. 81–94. DOI: [10.1007/978-3-642-39965-7_5](https://doi.org/10.1007/978-3-642-39965-7_5).
- [3] Cristina Archer and Ken Caldeira. “Global Assessment of High-Altitude Wind Power”. In: *Energies* 2 (June 2009). DOI: [10.3390/en20200307](https://doi.org/10.3390/en20200307).
- [4] Cristina Archer, Luca Delle Monache, and Daran Rife. “Airborne wind energy: Optimal locations and variability”. In: *Renewable Energy* 64 (Apr. 2014), pp. 180–186. DOI: [10.1016/j.renene.2013.10.044](https://doi.org/10.1016/j.renene.2013.10.044).
- [5] Ivan Argatov, P. Rautakorpi, and R. Silvennoinen. “Estimation of the mechanical energy output of the kite wind generator”. In: *Renewable Energy* 34 (June 2009), pp. 1525–1532. DOI: [10.1016/j.renene.2008.11.001](https://doi.org/10.1016/j.renene.2008.11.001).
- [6] Philip Bechtle et al. “Airborne wind energy resource analysis”. In: *Renewable Energy* 141 (Apr. 2019). DOI: [10.1016/j.renene.2019.03.118](https://doi.org/10.1016/j.renene.2019.03.118).
- [7] Massimo Canale, Lorenzo Fagiano, and Mario Milanese. “High Altitude Wind Energy Generation Using Controlled Power Kites”. In: *Control Systems Technology, IEEE Transactions on* 18 (Apr. 2010), pp. 279–293. DOI: [10.1109/TCST.2009.2017933](https://doi.org/10.1109/TCST.2009.2017933).
- [8] Massimo Canale, Lorenzo Fagiano, and Mario Milanese. “Power Kites for Wind Energy Generation [Applications of Control]”. In: *Control Systems Magazine, IEEE* 27 (Jan. 2008), pp. 25–38. DOI: [10.1109/MCS.2007.909465](https://doi.org/10.1109/MCS.2007.909465).
- [9] Massimo Canale et al. “Control of tethered airfoils for a new class of wind energy generator”. In: Jan. 2007, pp. 4020–4026. DOI: [10.1109/CDC.2006.376775](https://doi.org/10.1109/CDC.2006.376775).
- [10] Antonello Cherubini et al. “Airborne Wind Energy Systems: A review of the technologies”. In: *Renewable and Sustainable Energy Reviews* 51 (Aug. 2015), pp. 1461–1476. DOI: [10.1016/j.rser.2015.07.053](https://doi.org/10.1016/j.rser.2015.07.053).
- [11] Sean Costello et al. “Analysis of the Maximum Efficiency of Kite-Power Systems”. In: *Journal of Renewable and Sustainable Energy* 7 (Sept. 2015). DOI: [10.1063/1.4931111](https://doi.org/10.1063/1.4931111).
- [12] Michael Erhard and Hans Strauch. “Automatic Control of Pumping Cycles for the SkySails Prototype in Airborne Wind Energy”. In: Apr. 2018, pp. 189–213. ISBN: 978-981-10-1946-3. DOI: [10.1007/978-981-10-1947-0_9](https://doi.org/10.1007/978-981-10-1947-0_9).
- [13] Bernard Etkin. *Dynamics of Atmospheric Flight*. Dover Books on Aeronautical Engineering. Dover Publications, 2005.

- [14] Lorenzo Fagiano et al. “Automatic Crosswind Flight of Tethered Wings for Airborne Wind Energy: Modeling, Control Design, and Experimental Results”. In: *IEEE Transactions on Control Systems Technology* 22 (Jan. 2013). DOI: 10.1109/TCST.2013.2279592.
- [15] Boris Houska and Moritz Diehl. “Optimal Control for Power Generating Kites”. In: *2007 European Control Conference, ECC 2007* (Jan. 2006).
- [16] IEA. *World Energy Outlook 2018*. 2018. URL: <https://www.iea.org/reports/world-energy-outlook-2018>.
- [17] IEA. *World Energy Outlook 2019*. 2019.
- [18] A. Ilzhöfer, Boris Houska, and Moritz Diehl. “Nonlinear MPC of kites under varying wind conditions for a new class of large-scale wind power generators”. In: *International Journal of Robust and Nonlinear Control* 17 (Nov. 2007), pp. 1590–1599. DOI: 10.1002/rnc.1210.
- [19] Pijush Kundu, Ira Cohen, and David Dowling. *Fluid Mechanics*. Elsevier Academic Press, 2004.
- [20] Miles L. Loyd. “Crosswind kite power”. In: *Journal of Energy* (1980), pp. 106–111. DOI: 10.2514/3.48021.
- [21] *Makani Power*. <https://makanipower.com/>. Accessed: June 2020.
- [22] Elena Malz et al. “A reference model for airborne wind energy systems for optimization and control”. In: *Renewable Energy* 140 (Mar. 2019). DOI: 10.1016/j.renene.2019.03.111.
- [23] National Science Foundation. Retrieved June 2020 from. 2015. URL: https://www.nsf.gov/discoveries/disc_summ.jsp?cntn_id=134023.
- [24] Wubbo Ockels. “Laddermill, a novel concept to exploit the energy in the airspace”. In: *Aircraft Design* 4 (June 2001), pp. 81–97. DOI: 10.1016/S1369-8869(01)00002-7.
- [25] Peter R. Payne and Charles McCutchen. “Self-Erecting Windmill”. US Patent application US3987987A. Oct. 1976.
- [26] REN21. Retrieved June 2020 from. 2019. URL: <https://www.ren21.net/gsr-2019>.
- [27] Skysails Group. <https://skysails-group.com/>. Accessed: June 2020.
- [28] Richard Sutton and Andrew Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.
- [29] Paul Williams, Bas Lansdorp, and Wubbo Ockels. “Optimal Cross-Wind Towing and Power Generation with Tethered Kites”. In: *Journal of Guidance Control and Dynamics - J GUID CONTROL DYNAM* 31 (Jan. 2008), pp. 81–93. DOI: 10.2514/1.30089.
- [30] Paul Williams, Bas Lansdorp, and Wubbo Ockels. “Modeling and Control of a Kite on a Variable Length Flexible Inelastic Tether”. In: *Collection of Technical Papers - 2007 AIAA Modeling and Simulation Technologies Conference* 2 (Aug. 2007). DOI: 10.2514/6.2007-6705.
- [31] Mario Zanon et al. “Airborne Wind Energy Based on Dual Airfoils”. In: *Control Systems Technology, IEEE Transactions on* 21 (July 2013), pp. 1215–1222. DOI: 10.1109/TCST.2013.2257781.