Planning and control of an autonomous hovering airborne dedicated for the monitoring of structures

Elie Kahale

Université D'Évry Val D'Essonne
École Doctorale Sciences et Ingénierie
Laboratoire IBISC - Équipe SIMOB

THÈSE

présentée et soutenue publiquement le 21 Mars 2014
pour l'obtention du grade de

Docteur de l'Université D'Évry Val D'Essonne
spécialité : AUTOMATIQUE
par :

Elie Georges Kahale

Planification et Commande d'une Plate-Forme Aéroportée Stationnaire Autonome Dédiée à la Surveillance des Ouvrages d'Art

COMPOSITION DU JURY

Président :	R. CHATILA	DR. CNRS, Univ. Pierre et Marie Curie
Rapporteur :	R. LOZANO	DR. CNRS, Univ. Technologie de Compiègne
	M. CHADLI	MCF, HDR, Univ.Picardie Jules Verne
Examinateur :	K. KOZLOWSKI	Pr., Univ. Technologie de Poznan
	H. PIET-LAHANIER	MCF, HDR, ONERA
	J. LERBET	Pr., Univ. Évry Val d'Essonne
Directeur :	Y. BESTAOUI-SEBBANE	MCF, HDR, Univ. Évry Val d'Essonne
Co-encadrant :	P. CASTILLO,	CR. CNRS, Univ. Technologie de Compiègne
Invité :	F. DERKX	DR., IFSTTAR

Abstract

Today, the inspection of structures is carried out through visual assessments effected by qualified inspectors. This procedure is very expensive and can put the personal in dangerous situations. Consequently, the development of an unmanned aerial vehicle equipped with on-board vision systems is privileged nowadays in order to facilitate the access to unreachable zones.

In this context, the main focus in the thesis is developing original methods to deal with planning, reference trajectories generation and tracking issues by a hovering airborne platform. These methods should allow an automation of the flight in the presence of air disturbances and obstacles. Within this framework, we are interested in two kinds of aerial vehicles with hovering capacity: airship and quad-rotors.

Firstly, the mathematical representation of an aerial vehicle in the presence of wind has been realized using the second law of newton.

Secondly, the question of trajectory generation in the presence of wind has been studied: the problem of minimal time was formulated, analyzed analytically and solved numerically. Then, a strategy of trajectory planning based on operational research approaches has been developed.

Thirdly, the problem of trajectory tracking was carried out. A nonlinear robust control law based on Lyapunov analysis has been proposed. In addition, an autopilot based on saturation functions for quad-rotor crafts has been developed.

All methods and algorithms proposed in this thesis have been validated through simulations.

Résumé

Aujourd'hui, l'inspection des ouvrages d'art est réalisée de façon visuelle par des contrôleurs sur l'ensemble de la structure. Cette procédure est couteuse et peut être particulièrement dangereuse pour les intervenants. Pour cela, le développement du système de vision embarquée sur des drones est privilégié ces jours-ci afin de faciliter l'accès aux zones dangereuses.

Dans ce contexte, le travail de cette thèse porte sur l'obtention des méthodes originales permettant la planification, la génération des trajectoires de référence, et le suivi de ces trajectoires par une plate-forme aéroportée stationnaire autonome. Ces méthodes devront habiliter une automatisation du vol en présence de perturbations aérologiques ainsi que des obstacles. Dans ce cadre, nous nous sommes intéressés à deux types de véhicules aériens capable de vol stationnaire : le dirigeable et le quadri-rotors.

Premièrement, la représentation mathématique du véhicule volant en présence du vent a été réalisée en se basant sur la deuxième loi de Newton. Deuxièmement, la problématique de génération de trajectoire en présence de vent a été étudiée : le problème de temps minimal est formulé, analysé analytiquement et résolu numériquement. Ensuite, une stratégie de planification de trajectoire basée sur les approches de recherche opérationnelle a été développée.

Troisièmement, le problème de suivi de trajectoire a été abordé. Une loi de commande non-linéaire robuste basée sur l'analyse de Lyapunov a été proposée. En outre, un pilote automatique basée sur les fonctions de saturations pour un quadri-rotors a été développée.

Les méthodes et algorithmes proposés dans cette thèse ont été validés par des simulations.

Acknowledgements

First and foremost I would like to express my heartfelt gratitude to my supervisor, Mrs. Yasmina Bestaoui and Mr. Pedro Castillo for their enduring guidance, encouragement, patience and support throughout my PhD work. They have been inspiring role models during the last years, and I look forward to learning more from them for much longer.

As a thesis director, Mrs. Yasmina Bestaoui supported me in all stages of this work. She always gave me constant encouragement and advice, despite her busy agenda. Without her coherent and illuminating instruction, this thesis would not have reached its present form.

I am especially grateful with Mr. Pedro Castillo. I highly appreciate his comments and the time that he spent to improve our publications and provide me relevant advice. I will not forget our very frequent teleconferences during which he gave me motivation and encouragement.

My sincere thanks to Mr. Raja Chatila for agreeing to be the president of the thesis defense examining committee. I also wish to express my most faithful gratitude to the principal readers of this dissertation, Mr. Rogelio Lozano and Mr. Mohammed Chadli for the excellent work they have done and the very swift time in which they have done it. Likewise, I would like to thank Mrs. Hélène Piet-Lahanier, Mr. Krzysztof Kozłowski and Mr. Jean Lerbet for accepting to participate in the defense of this thesis.

I gratefully acknowledge to IBISC laboratory and his director Mr. Saïd Mammar for hosting me during the last four years.

I also want to thank all my friends for their support, help and encouragement.

Finally, no words can completely describe how grateful I am to my family: my mother Mouna, my father Georges, my sister Diane, my aunt Solange and my fiancée Souaad. I will be forever thankful with all of them for the support and confidence that they give me.

Contents

List of Figures

List of Tables

Nomenclature

A_{RP}	Aircraft's reference point
a	Linear acceleration
B	Buoyancy force (airships)
$C_{\mathcal{B}}^{\mathcal{S}}$	Transformation matrix from $R_{\mathcal{B}}$ to $R_{\mathcal{S}}$
$C_{\mathcal{B}}^{\mathcal{W}}$	Transformation matrix from $R_{\mathcal{B}}$ to $R_{\mathcal{W}}$
$C_{\mathcal{I}}^{\mathcal{B}}$	Transformation matrix from $R_{\mathcal{I}}$ to $R_{\mathcal{B}}$
$C_{\mathcal{I}}^{\mathcal{W}}$	Transformation matrix from $R_{\mathcal{I}}$ to $R_{\mathcal{W}}$
$C_{\mathcal{L}}^{\mathcal{W}}$	Transformation matrix from $R_{\mathcal{L}}$ to $R_{\mathcal{W}}$
$C_{\mathcal{S}}^{\mathcal{W}}$	Transformation matrix from $R_{\mathcal{S}}$ to $R_{\mathcal{W}}$
$C(V)$	Coriolis and centripetal effects (airships)
$C_A(V)$	Coriolis and centripetal effects due to added mass and inertia phenomena (airships)
C_B	Center of buoyancy (airships)
C_G	Center of gravity (airships)
C_l	Roll stationary coefficient (airships)
C_L	Normal stationary coefficient (airships)
C_m	Yaw stationary coefficient (airships)
C_n	Pitch stationary coefficient (airships)
C_N	Lateral stationary coefficient (airships)
C_T	Tangential stationary coefficient (airships)
C_V	Center of volume (airships)
$\overrightarrow{D}$	Unit vector pointing to Down
D	Drag force (airships)
$\overrightarrow{E}$	Unit vector pointing to East
E_K	Kinetic energy (quad-rotors)
E_P	Potential energy (quad-rotors)
f_a	Aerodynamical force (airships)
f_{AM}	Added mass force (airships)
F_{ext}	External forces
f_T	Thrust force (Airchips)

f_g	Gravity force (airships)
g	Gravitational acceleration
$\mathcal{H}$	Hamiltonian function
I_G	Inertia matrix at C_G (airships)
I_N	Inertia matrix at C_V (airships)
I_{xx}	Moment inertia with respect to x axis (airships)
I_{yy}	Moment inertia with respect to y axis (airships)
I_{zz}	Moment inertia with respect to z axis (airships)
$\mathbb{J}$	inertia matrix (quad-rotors)
$J_1(\eta_2)$	Linear velocity transformation matrix (airships)
$J_2(\eta_2)$	Angular velocity transformation matrix (airships)
ℓ	distance from the motor to the center of gravity (quad-rotors)
L	Lift force (airships)
$\mathcal{L}$	Lagrangian function
$\mathscr{L}$	Robust control Lyapunov function
L_{ref}	Airship's reference length
m	Total mass of the vehicle
M	Inertia matrix (airships)
M_a	Added mass matrix resulting from the linear accelerations (airships)
M_A	Full added mass and inertia matrix (airships)
M_{ext}	External moments
$\overrightarrow{N}$	Unit vector pointing to North
q	Configuration
$\overrightarrow{r}$	Position vector
$R_{\mathcal{B}}$	Body Fixed Frame
$R_{\mathcal{I}}$	Earth Fixed Inertial Frame
$R_{\mathcal{L}}$	Local Horizon Frame
$R_{\mathcal{S}}$	Stability Frame
$R_{\mathcal{W}}$	Wind Relative Frame
$R_{x/\phi}$	Rotation matrix of an angle ϕ about x axis
$R_{y/\theta}$	Rotation matrix of an angle θ about y axis
$\left[r_{C_G}^{C_V}\times\right]$	Skew-symmetric matrix associated to the distance from C_V to C_G (airships)
S_{ref}	Airship's reference area
T	Thrust force
T_{trans}	Translational kinetic energy (quad-rotors)
T_{rot}	Rotational kinetic energy (quad-rotor)
$\mathscr{U}$	Control space

V	Velocity vector
$\mathscr{V}$	All condidate Lyapunov function
$V_1 = [u, v, w]^T$	Linear velocity vector expressed in $R_{\mathcal{B}}$
$V_2 = [p, a, r]^T$	Angular velocity vector expressed in $R_{\mathcal{B}}$
V_a	Relative velocity
V_G	Velocity at C_G (airships)
ν_{hull}	Volume of airship's hull
V_V	Velocity at C_V (airships)
V_W	Wind velocity
$\omega_{R_{\mathcal{W}}/R_{\mathcal{I}}}$	Angular velocity of $R_{\mathcal{W}}$ with respect to $R_{\mathcal{I}}$
W	Weight force
$\mathscr{W}$	Disturbances space
$\mathscr{X}$	State space
Δ_A	Accessibility distribution
Δ_{A_0}	Strong accessibility distribution
α	Angle of attack
β	Side-slip angle
χ	Heading angle
γ	Flight path angle
ψ	Yaw angle
θ	Pitch angle
ϕ	Roll angle
η	Six dimension vector position and orientation of the body fixed frame (airships)
$\eta_1 = [x, y, z]^T$	Position vector of the body fixed frame expressed in $R_{\mathcal{I}}$ (airships)
$\eta_2 = [\phi, \theta, \psi]^T$	Orientation vector of the body fixed frame expressed in $R_{\mathcal{I}}$ (airships)
ρ	Density of air
$\Omega_{R_{\mathcal{W}}/h}$	Angular velocity of $R_{\mathcal{W}}$ about $h_{\mathcal{W}}$
$\Omega_{R_{\mathcal{W}}/y}$	Angular velocity of $R_{\mathcal{W}}$ about $y_{\mathcal{W}}$
φ	Generalized coordinates of quad-rotors craft
Π_G	Angular momentum with respect to C_G (airships)
Π_V	Angular momentum with respect to C_V (airships)
τ_a	Aerodynamic tensor (airships)
τ_p	Propulsion tensor (airships)
τ_s	Static tensor (airships)
τ_{sta}	Stationary phenomena tensor (airships)
$\tau = [\tau_\psi, \tau_\theta, \tau_\phi]^T$	Translational force applied to quad-rotors craft
λ	Co-state variables

Chapter 1

Introduction

Unmanned Aerial Vehicle technology have seen an enormous and promptly development in the last two decades. This kind of aerial vehicles is being increasingly used in military and civilian domains such as, surveillance, reconnaissance, mapping, cartography, border patrol, inspection, homeland security, search and rescue, weather and hurricane monitoring, fire detection, agricultural imaging, traffic monitoring, pollutant estimation, etc... [40, 41, 63, 162].

Based on UAVs shapes and structures, they can be classified in the following categories, [128, 134]:

- Fixed-wing UAV: This type of UAV is capable of flight using forward motion with a relative velocity such that a sufficient lift is generated on wings. In addition, it requires a runway for the take-off and landing.
- Rotary-wing UAV: This kind of aerial vehicle has the advantage of hovering capability and high maneuverability. These aircrafts use one or more propellers to produce the thrust force necessary for motion. In addition, the ability of rotor-crafts to take off and land in limited spaces and to hover above targets, gives such kind of UAVs the superiority over fixed wing aircrafts especially for missions that require hovering flight.
- Lighter than air UAV: Unlike fixed or rotary wing aerial vehicles, the lift of lighter than air vehicles is mainly generated by buoyancy force. This characteristic makes airships noiseless, ecological and very useful for long term environmental applications.

However, other classifications can be found in literature according to a variety of parameters that include, vehicle configuration, shapes, structures, size, weight, endurance and range, maximum altitude, engine type among others.

1.1 Structure Inspection

The inspection of structures like bridges, dams, dikes, towers, etc... is a fundamental tool to ensure the good quality of service and users safety. Such architectures can fail structurally in a catastrophic fashion, or they can fall through obsolescence. However, there are many causes that can weaken a structure. For instance, the growth from microscopic to catastrophic lengths of undetected cyclic load-induced fatigue cracks, the reduction in strength of steel components produced by environmentally-assisted corrosion, etc... In most of the cases, structure failures has had various effects such as property damage, wounded, and fatalities among others.

According to [46, 47], visual inspection is the most basic and also the most common method by which bridges and other civil structures are evaluated. This technique relies in visual assessments that cover the entire structure. It is a long process which requires a large number of qualified inspectors and a various of specialized equipment. In order to ensure the safety inspection personal, powered walkways and platforms to observe the inaccessible zones were created. Experience had revealed that such kind of facilities are difficult or even impossible, in some cases, to use. Consequently, the Laboratoire Central de Ponts et Chaussées (LCPC), today known as IFSTTAR, began to explore the possibility of using an unmanned aerial vehicle equipped with on-board vision systems for structure inspection task in the end of the 1990's. It was not until 2005 that the LCPC acquired its first helicopter type UAV. This UAV has been the subject of several improvements to include new measurement and observation sensors including wireless communications, global positioning, etc... The imagery and data obtained from the UAV is post-processed using a specialized software to reconstruct a virtual model of the structure.

By making the inspection process autonomous, we take advantage of the following points:

- Reduce cost due to minimal personnel and operating costs.
- Reduce inspection time.
- Increase the safety of staff.
- facilitate the access to unreachable and dangerous zones.
- Allow the long-term structure monitoring and following-up the evolution of existing cracks.

The inherent nature of a structure inspection mission using UAVs implies the interaction of different research domains including, UAV stabilization, optimal control, navigation, obstacle avoidance, wireless communications, computer vision, etc...

1.2 Motivation of this thesis and objective

This thesis is motivated by autonomous UAV-based bridge inspection missions. From flight planning point of view, the neighborhood of bridges is a challenging environment due to the presence of obstacles, and different meteorological phenomenon. In addition, resources optimization, risk and operational cost minimization are fundamental issues in these kind of tasks. Therefore, we are interested in optimal trajectory planning in presence of obstacles, and meteorological turbulence such as wind shear, Venturi effect, Karman vortex among others. The main goal of this thesis is to automatize the motion planning and optimize the generated trajectory for bridge inspection such that the task is accomplished in a minimal time.

The nature of inspection mission requires an aerial vehicle capable of hovering flight. Consequently, we are interested in two types of UAVs: airships and quadrotors. The overall UAV system architecture, as presented in Fig. 1.1 consists of five layers: motion planner, trajectory generator, trajectory tracker, autopilots, and the vehicle.

The motion planner creates a plan for the unmanned aerial vehicle defining a set of way-points from the vehicle initial point to the desired final one. While, the trajectory generator creates a flyable trajectory to connect way-points in the same order as it was given to it by the motion planner. Next, the generated trajectory is given as reference signals to the trajectory tracker which, in turn, computes the necessary mid-level control inputs to ensure the trajectory following. Then, these inputs are provided to autopilots as reference values to stabilize the UAV.

Our objective is to propose and develop algorithms to deal with motion planner, trajectory generator, and trajectory tracker levels. The planner and trajectory generator must take into consideration the measurable parameters of meteorological phenomena, the on-board energy limitations, and the presence of obstacles in the environment. Toward this end, a flight planning system based on optimal control was proposed. On the other side, the trajectory tracker should ensure trajectory following in the presence of disturbances such as wind gusts. For this purpose a control strategy based on inverse optimality was introduced.

1.3 Contributions of this thesis

The main contributions of this thesis are the following:

- Adopting kinematic point mass model for quad-rotors crafts.
- Planning in 3D with varying velocity, heading angle, and flight path angle.

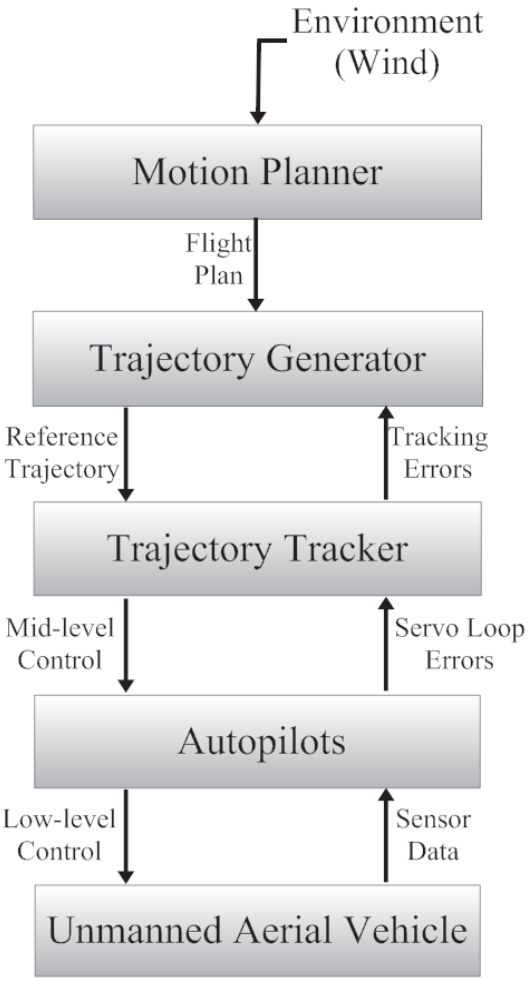

FIGURE 1.1: UAV system architecture.

- Taking into consideration the presence of the wind in the trajectory generation phase and introduce the updated trajectory principle.
- Introducing a robust control strategy to make autonomous trajectory following for kinematic aerial vehicle, and dynamic lighter than air vehicle in a windy environment.

Parts of the work related with this thesis have been previously published.

Related with Chapter 3, the following papers have been published:

– Y. Bestaoui and E. Kahale, "Analysis of Time Optimal 3D Paths for an Autonomous Aircraft with a Piecewise Constant Acceleration", In *49th AIAA Aerospace Sciences Meeting*, Orlando Florida, USA, January 2011.

– Y. Bestaoui and E. Kahale, "Generation of Time Optimal Trajectories of an Autonomous Airship", In *IEEE Workshop on Robot Motion Control (ROMOCO 2011)*, Bukowy Dworek, Poland, June 2011.

– Y. Bestaoui and E. Kahale, "Time Optimal Trajectories of a Lighter Than Air Robot with Second Order Constraints and a Piecewise Constant Velocity Wind", *AIAA Journal of Aerospace Computing Information and Communication*, 10(4): 155-171, April 2013.

Related with Chapter 4, the following papers has been published:

– E. Kahale, Y. Bestaoui, and P. Castillo, "Path tracking of a small autonomous airplane in wind gusts", In *EEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vilamoura, Algarve, Portugal, October 2012.
– E. Kahale, P. Castillo, and Y. Bestaoui, "Autonomous path tracking of a kinematic airship in presence of unknown gust", In *International Conference on Unmanned Aircraft Systems (ICUAS)*, Philadelphia, PA, USA, June 2012.

An extended version of the last paper was also published in

– E. Kahale, P. Castillo, and Y. Bestaoui, "Autonomous path tracking of a kinematic airship in presence of unknown gust", *Journal of Intelligent and Robotic Systems*, 69(1-4): 431-446, January 2013.

1.4 Thesis outline

The manuscript is divided into Three main chapters. General equations of motion for lighter than air vehicles and quadrotors crafts are introduced in Chapter 2. The vehicles are modeled in two different ways: a six degree of freedom model, called rigid body model and devoted for stability and control problem, and a three degree of freedom model, called point mass model used in navigation and guidance control systems. Besides the mathematical representations of the aerial vehicles, the wind gusts and Venturi effect modeling question has been addressed.

Chapter 3 treats the trajectory generation and motion planning problems. In general, the autonomy of a UAV is defined as its capacity to accomplish different types of tasks with a high level of performance, maneuverability and with less oversight of human operators [12, 92]. These tasks require flexible and Powerful algorithms that convert high-level mission specifications from humans into low-level descriptions of the vehicle's motion. The terms Motion Planning and Trajectory Planning are often employed for such kind of problems [111]. In order to connect a starting and a target points, feasible and flyable trajectories must be defined. The feasibility criteria is carried out by motion planning algorithms. This process produce a plan to steer the UAV safely to its target, without taking into account its dynamical constraints. Whilst, the trajectory generation

problem takes the solution obtained by the motion planning algorithm and determines the way to fly along this solution with respect to the vehicle's mechanical limitations. In other words, it guarantees the flyable aspect of the trajectory [12, 111].

Once a feasible and flyable trajectory is generated, it becomes necessary to move one step down into low-level control design. That means, to deal with trajectory tracking question. This problem consists of stabilization of the state, or an output function of the state, to a desired reference value, possibly time-varying [123]. This problem is handled in chapter 4. Finally a general conclusion and perspective are given in chapter 5.

Chapter 2

Modeling

2.1 Introduction

In this chapter, the general equations of motion of an aerial vehicle flying in the Earth's atmosphere are derived and the coordinate systems in which these equations are written are discussed. The aircraft is assumed to be a rigid body and the Newton's laws of motion are used.
The equations governing the translational and rotational motion of an aircraft can be divided into the following two sets:

- Kinematic equations giving the translational and rotational position relative to the earth reference frame.
- Dynamic equations relating forces to translational acceleration $\left(\Sigma\overrightarrow{F_{ext}} = m\overrightarrow{a}\right)$ and moments to rotational acceleration $\left(\Sigma\overrightarrow{M_{ext}} = I\overrightarrow{\dot{\omega}}\right)$.

These equations are referred to as six degree of freedom (6DOF) equations of motion.

From navigation and guidance control system point of view, the vehicle's rotation rates are considered to be small. Thus, only translational equations known as three degree of freedom (3DOF) equations of motion are used. These equations are uncoupled from the rotational equations by assuming negligible rotation rates and neglecting the effect of control surface deflections on aerodynamic forces. For example, to maintain a given speed for an aerial vehicle on cruise flight, the pitching moment is required to be zero through an elevator deflection. This process contributes to the lift and drag forces applied on the aircraft. Thus, by neglecting this contribution, the translational and rotational equations can be uncoupled [81].

On the other hand, the stability and control problems are related to the relative motion of the center of gravity of the aircraft with respect to the ground, and the motion of the vehicle about its center of gravity. Hence, the stability and control studies involve the use of the six degree of freedom equations of motion.

Next, the coordinate systems used to develop the equations of motion are pointed out.

2.2 Coordinate Systems

The vehicle's motion in the space is defined by several coordinate systems which can be classified into two categories: Earth axis and body axis. All coordinate systems used are right handed orthogonal [38, 81, 121].

2.2.1 Earth Axes System

There are two coordinate systems related to the Earth axes. The first one is fixed, known as ***Earth Fixed Inertial Frame*** and denoted as $R_{\mathcal{I}}\left(O, x, y, h\right)$. This axis system is regarded as an inertial reference frame in which Newton's laws of motion are valid, i.e. the rotational velocity of the Earth is neglected [142]. The second reference frame is moving, called ***Local Horizon Frame*** and referred to as $R_{\mathcal{L}}\left(A_{RP}, x_{\mathcal{L}}, y_{\mathcal{L}}, h_{\mathcal{L}}\right)$, see Figure (2.1). Both Earth axes systems are related to the geographic coordinate system **NED** which has the unit vectors $\overrightarrow{N}$ on x axis pointing to the North, $\overrightarrow{E}$ on y axis pointing to the East, and $\overrightarrow{D}$ on h axis pointing toward to the center of the Earth along the gravitational vector. Referring to Figure (2.1), Notice that the horizontal planes defined by (O, x, y) and $(A_{RP}, x_{\mathcal{L}}, y_{\mathcal{L}})$ are parallel. The previous two coordinate systems can be distinguished by the location of their origins. The origin of $R_{\mathcal{I}}$, i.e. O, is an arbitrary point located on the surface of the Earth while the origin of $R_{\mathcal{L}}$, i.e. A_{RP}, is related to the aircraft's reference point.

2.2.2 Body Axes

These coordinate systems have the aircraft's reference point (A_{RP}) as origin, and its axes are defined with respect to the vehicle. Three type of vehicle coordinate system can be distinguished: Body fixed frame, Stability frame and Wind frame.

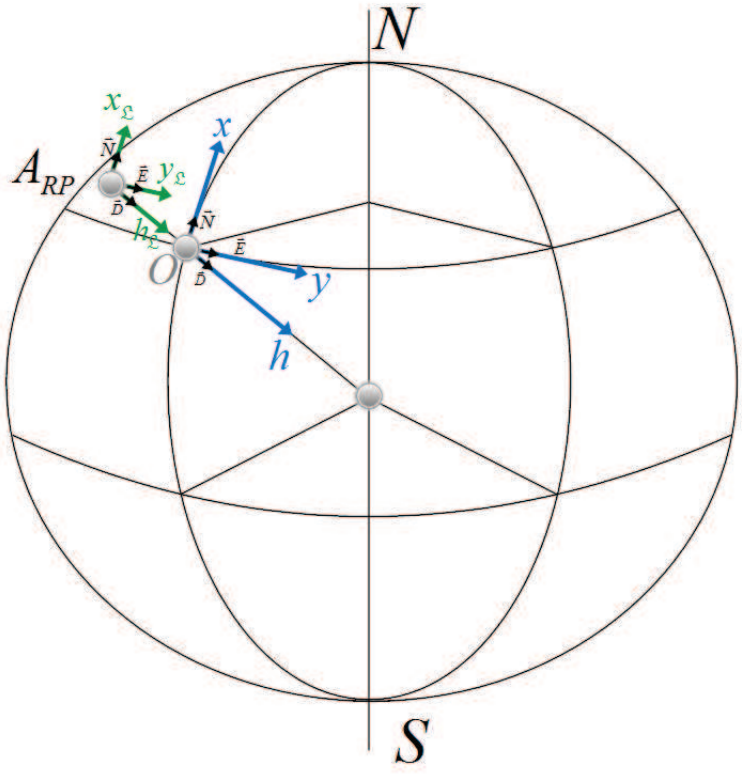

FIGURE 2.1: Earth Coordinate Systems.

2.2.2.1 Body Fixed Frame

The body fixed coordinate system, $R_{\mathcal{B}}\,(A_{RP}, x_{\mathcal{B}}, y_{\mathcal{B}}, h_{\mathcal{B}})$ is attached to the vehicle and constrained to move with it, see Figure (2.2). Thus when the aircraft goes from its initial flight condition the axes move with the vehicle and the motion is quantified in terms of perturbation variables referred to the moving axes. The $(x_{\mathcal{B}} \times h_{\mathcal{B}})$ plane defines the plane of symmetry of the aircraft such that the $x_{\mathcal{B}}$ axis is directed along the axis of symmetry. Thus $y_{\mathcal{B}}$ axis is oriented to the right while $h_{\mathcal{B}}$ axis is pointed downward.

2.2.2.2 Stability Frame

The stability axes, $R_{\mathcal{S}}\,(A_{RP}, x_{\mathcal{S}}, y_{\mathcal{S}}, h_{\mathcal{S}})$, are a special case of body-fixed axes in which the orientation of the vehicle axis system is determined by the equilibrium flight condition, see Figure (2.2). The axis $x_{\mathcal{S}}$ is chosen to coincide with the velocity vector, $\overrightarrow{V}$, at the start of the motion. Therefore, $R_{\mathcal{S}}$ is obtained by rotating $R_{\mathcal{B}}$ about $y_{\mathcal{B}}$ axis through an angle α known as the angle of attack. Remark that $(x_{\mathcal{S}} \times h_{\mathcal{S}})$ remains in the plane of symmetry of the vehicle and $y_{\mathcal{S}}$ coincides with $y_{\mathcal{B}}$. If the reference flight is

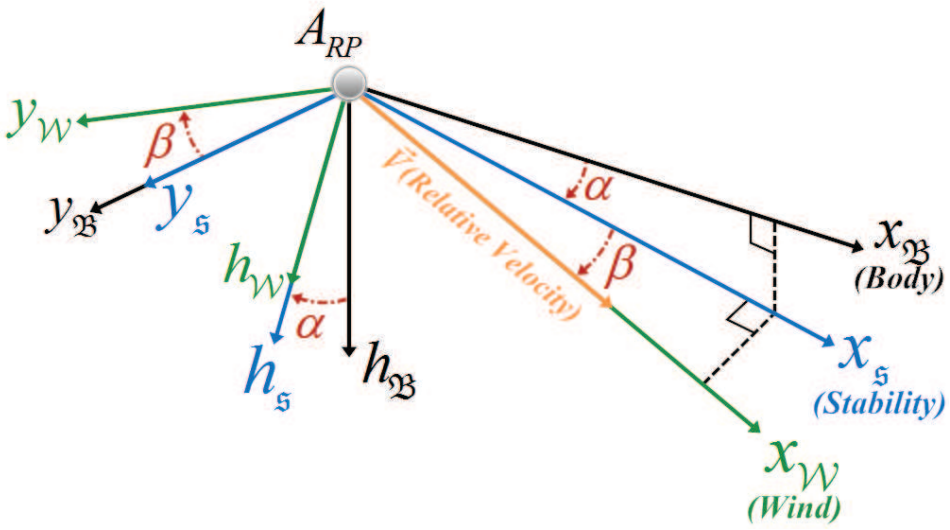

FIGURE 2.2: Aerial Vehicle's Coordinate Systems.

not symmetric, i.e. $\beta \neq 0$, then the $x_{\mathcal{S}}$ axis is chosen to be the projection of $\vec{V}$ in the plane of symmetry.

2.2.2.3 Wind Relative Frame

The axes of the wind relative coordinate system, $R_{\mathcal{W}}\left(A_{RP}, x_{\mathcal{W}}, y_{\mathcal{W}}, h_{\mathcal{W}}\right)$, are related to the flight path of the vehicle, i.e. with respect to the velocity vector, $\vec{V}$, Figure (2.2). This coordinate system is attained by rotating $R_{\mathcal{S}}$ about $h_{\mathcal{S}}$ axis through the sideslip angle, β. Notice that, the relationship between $R_{\mathcal{W}}$ and $R_{\mathcal{B}}$ defines the angle of attack, α, and the sideslip angle, β. On the other hand, if the reference flight is symmetric, i.e. $\beta = 0$ and $\vec{V}$ belongs to the plane of symmetry, then the wind relative axes coincide with the stability axes.

Remark: The aircraft's reference point, i.e. A_{RP}, is an invariant point. It belongs to the vehicle. In fixed and rotary wings aircraft A_{RP} is chosen to be the vehicle's center of gravity (C_G). On the other hand, for a lighter than air vehicles, the center of gravity can not be used as a reference point because its location varies in time depending on pressure. However, two invariant points can be proposed for modeling an airship; the center of volume of the vehicle, C_V, or its nose, N. For this thesis, the center of volume is privileged.

2.2.3 Coordinates Transformation

Previously, we have defined the different coordinate systems used to describe the motion of an aerial vehicle. The relations between such frames are known as coordinates transformation, and represent the orientation of each coordinate system with respect to the others. The transformation matrices between $R_{\mathcal{I}}$, $R_{\mathcal{B}}$, $R_{\mathcal{S}}$, and $R_{\mathcal{W}}$ are the subject of the following paragraphs.

Transformation from $R_{\mathcal{I}}$ to $R_{\mathcal{B}}$ The description of $R_{\mathcal{I}}$ with respect to $R_{\mathcal{B}}$ is made using the following sequence of rotation [155].
Starting from the Earth fixed inertial frame:

1. Rotating about the $h_{\mathcal{I}}$ axis by "ψ", also known as yaw angle. The positive direction of rotation is taken so that the nose turns toward to right.
2. Rotating about the new y axis by "θ", called as pitch angle. The positive direction of rotation is defined in order that the nose points to up.
3. Rotate about the new x axis by "ϕ", also named roll angle. The positive direction of rotation is presented such that the right-side heads to down.

Conversely, to go in the opposite direction, i.e. from the body frame to the Earth fixed inertial frame, the sequence roll, pitch, yaw must be followed. Notice that the yaw, pitch, and roll angles ψ, θ, ϕ are commonly referred to as *Euler angles*. This process is illustrated in Figure (2.3). From previous, we can write

$$\begin{bmatrix} x_{\mathcal{B}} \\ y_{\mathcal{B}} \\ h_{\mathcal{B}} \end{bmatrix} = C_{\mathcal{I}}^{\mathcal{B}} \begin{bmatrix} x_{\mathcal{I}} \\ y_{\mathcal{I}} \\ h_{\mathcal{I}} \end{bmatrix} \tag{2.1}$$

where $C_{\mathcal{I}}^{\mathcal{B}}$ denotes the transformation matrix and is given by

$$C_{\mathcal{I}}^{\mathcal{B}} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & \sin\phi\cos\theta \\ \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi & \cos\phi\cos\theta \end{bmatrix} \tag{2.2}$$

Transformation from $R_{\mathcal{B}}$ to $R_{\mathcal{S}}$ The transformation from body fixed frame to the stability frame is done using the following relationship, see Figure (2.2)

$$\begin{bmatrix} x_{\mathcal{S}} \\ y_{\mathcal{S}} \\ h_{\mathcal{S}} \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix}}_{C_{\mathcal{B}}^{\mathcal{S}}} \begin{bmatrix} x_{\mathcal{B}} \\ y_{\mathcal{B}} \\ h_{\mathcal{B}} \end{bmatrix} \tag{2.3}$$

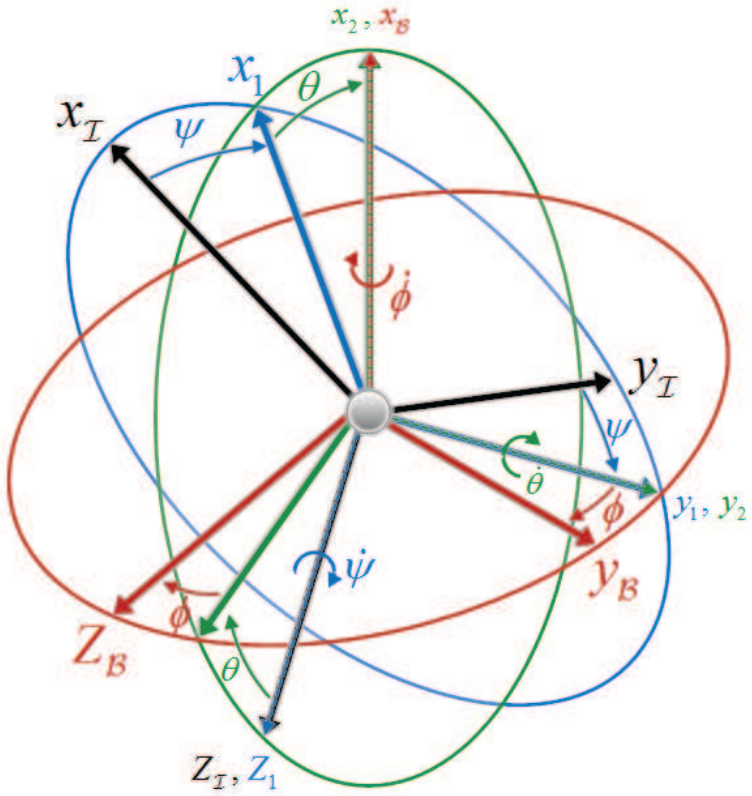

FIGURE 2.3: The Euler angles.

Transformation from $R_\mathcal{S}$ to $R_\mathcal{W}$ From Figure (2.2), we can describe this coordinates transformation as follows

$$\begin{bmatrix} x_\mathcal{W} \\ y_\mathcal{W} \\ h_\mathcal{W} \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\beta & \sin\beta & 0 \\ -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{C_\mathcal{S}^\mathcal{W}} \begin{bmatrix} x_\mathcal{S} \\ y_\mathcal{S} \\ h_\mathcal{S} \end{bmatrix} \tag{2.4}$$

Observe that the complete rotation from body to wind axes is given by

$$\begin{bmatrix} x_\mathcal{W} \\ y_\mathcal{W} \\ h_\mathcal{W} \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\alpha\cos\beta & \sin\beta & \sin\alpha\cos\beta \\ -\cos\alpha\sin\beta & \cos\beta & -\sin\alpha\sin\beta \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix}}_{C_\mathcal{B}^\mathcal{W}=C_\mathcal{B}^\mathcal{S}\cdot C_\mathcal{S}^\mathcal{W}} \begin{bmatrix} x_\mathcal{B} \\ y_\mathcal{B} \\ h_\mathcal{B} \end{bmatrix} \tag{2.5}$$

Transformation from $R_\mathcal{I}$ to $R_\mathcal{W}$ The wind relative coordinate system is related to the Earth fixed inertial frame by a sequence of two rotations.

Starting from $R_{\mathcal{I}}$:

1. Rotating about the $y_{\mathcal{I}}$ axis by "γ", also called as flight path angle. The positive direction of rotation is taken in counter-clockwise direction.
2. Rotating about the new h axis by "χ", also known as heading angle. The positive direction of rotation is taken in clockwise direction.

Thus, in terms of coordinate transformations we have

$$\begin{bmatrix} x_{\mathcal{W}} \\ y_{\mathcal{W}} \\ h_{\mathcal{W}} \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\chi\cos\gamma & \sin\chi\cos\gamma & -\sin\gamma \\ -\sin\chi & \cos\chi & 0 \\ \cos\chi\sin\gamma & \sin\chi\sin\gamma & \cos\gamma \end{bmatrix}}_{C_{\mathcal{I}}^{\mathcal{W}}} \begin{bmatrix} x_{\mathcal{I}} \\ y_{\mathcal{I}} \\ h_{\mathcal{I}} \end{bmatrix} \tag{2.6}$$

Since, the Earth fixed inertial frame and the local horizon frame are parallel to each other. The matrix $C_{\mathcal{I}}^{\mathcal{W}}$ can be used as a transformation matrix from $R_{\mathcal{L}}$ to $R_{\mathcal{W}}$.

In the next section, a full six-degree-of-freedom (6DOF) nonlinear mathematical model for both; *Lighter Than Air Vehicles* and *Quad-Rotors* flying in the space is carried out.

2.3 6DOF Equations of Motion (Rigid Body Model)

This modeling is also known as ***Rigid Body Model*** because it is derived in the body fixed coordinate system, i.e. $R_{\mathcal{B}}$, and it assumes that the aerial vehicle is a rigid body.

2.3.1 Equations of Motion for Lighter Than Air Vehicles

The development of the 6DOF nonlinear equations of motion for an airship is similar to a fixed-wing airplane. The major differences are caused by the structure of the vehicle as it is buoyant and the fact that its motion displaces a large volume of surrounding air. The buoyancy, added mass and added inertia forces, neglected in modeling the airplane's dynamics, add more nonlinear characteristics in the airship's dynamics [11, 59, 98].
In addition, the lighter than air vehicle is an under-actuated system because it has fewer control inputs than degrees of freedom. It is mainly controlled through thrust force and control surfaces. The force inputs are available from two main propellers on each side of gondola, which provides a complementary lift to oppose the weighting mass, as well as a forward thrust controlling the longitudinal speed [17]. Furthermore, the fact of varying the thrust generated by each propeller, through changing the angular velocity of each engine, provides torque to control the rolling motion near hover. On the other side, the flight control surface; i.e. rudders and elevators, of the tail provide torque input to

control pitching and yawing motions.

Deducing the dynamic model of airships requires the following assumptions:

- The hull is considered as a solid. Thus, the aero-elastic phenomena and the motion of lifting gaze inside the hull are ignored.
- The mass of the blimp and its volume are considered as constant.
- The Earth is considered as flat above the flight area.

The general motion of the lighter than air vehicle in 6DOF can be described by the following vectors [11, 59, 139]

$$\eta = \left[\eta_1^T, \eta_2^T\right]^T; \quad \eta_1 = [x, y, z]^T; \quad \eta_2 = [\phi, \theta, \psi]^T \tag{2.7}$$

$$V = \left[V_1^T, V_2^T\right]^T; \quad V_1 = [u, v, w]^T; \quad V_2 = [p, q, r]^T \tag{2.8}$$

where, η denotes the position and orientation of C_V with respect to $R_{\mathcal{I}}$ and V presents the linear and angular velocity expressed in $R_{\mathcal{B}}$. In the next section, the kinematic equations relating the body fixed reference to the fixed Earth inertial reference will be derived.

2.3.1.1 Kinematics

The position of all points belonging to the airship with respect to Earth fixed inertial frame are defined through the orientation of body fixed coordinate system, and the position of its origin, i.e. the center of volume, with respect to $R_{\mathcal{I}}$. Thus, the kinematic equations are obtained by effecting a velocity transformation which leads to

$$\begin{bmatrix} \dot{\eta_1} \\ \dot{\eta_2} \end{bmatrix} = \begin{bmatrix} J_1(\eta_2) & 0_{3\times3} \\ 0_{3\times3} & J_2(\eta_2) \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} \tag{2.9}$$

Remember that V_1 and V_2, defined in Equation (2.8), denote the linear and angular velocity of the vehicle respectively. In addition, both are described in $R_{\mathcal{B}}$. Hence, the first line in Equation (2.9) can be regarded as a linear velocity transformation, while the second line presents an angular velocity transformation [17, 59, 139]. On the other hand, the transformation matrices $J_1(\eta_2)$ and $J_2(\eta_2)$ depend on Euler angles, i.e. ϕ, θ, and ψ. $J_1(\eta_2)$ describes the transformation matrix from $R_{\mathcal{B}}$ to $R_{\mathcal{I}}$ and it is related with $C_{\mathcal{I}}^{\mathcal{B}}$, Equation (2.2), by the following relationship

$$J_1(\eta_2) = \left[C_{\mathcal{I}}^{\mathcal{B}}\right]^T \tag{2.10}$$

Besides, from Figure (2.3) we state

$$V_2 = \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_{x/\phi} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_{x/\phi} R_{y/\theta} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} = J_2^{-1}(\eta_2)\,\dot{\eta_2} \tag{2.11}$$

Expanding the previous equation yields to the following

$$J_2^{-1}(\eta_2) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \cos\theta\sin\phi \\ 0 & -\sin\phi & \cos\theta\cos\phi \end{bmatrix} \Longrightarrow J_2(\eta_2) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \tag{2.12}$$

From Equation (2.9), Equation (2.10), and (2.12), we obtain

$$\begin{aligned}
\dot{x} &= (\cos\psi\cos\theta)\,u + (-\sin\psi\cos\phi + \cos\psi\sin\theta\sin\phi)\,v && \text{(2.13a)}\\
&\quad + (\sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi)\,w \\
\dot{y} &= (\sin\psi\cos\theta)\,u + (\cos\psi\cos\phi + \sin\psi\sin\theta\sin\phi)\,v && \text{(2.13b)}\\
&\quad + (-\cos\psi\sin\phi + \sin\psi\sin\theta\cos\phi)\,w \\
\dot{z} &= (-\sin\theta)\,u + (\cos\theta\sin\phi)\,v + (\cos\theta\cos\phi)\,w && \text{(2.13c)}\\
\dot{\phi} &= (\sin\phi\tan\theta)\,q + (\cos\phi\tan\theta)\,r + p && \text{(2.13d)}\\
\dot{\theta} &= (\cos\phi)\,q + (-\sin\phi)\,r && \text{(2.13e)}\\
\dot{\psi} &= (\sin\phi\sec\theta)\,q + (cos\phi\sec\theta)\,r && \text{(2.13f)}
\end{aligned}$$

The previous mathematical relationships describes the translational and rotational kinematic equations of a lighter than air vehicle moving in three dimensional space. In the next section, the full 6DOF dynamical model of an airship is introduced.

2.3.1.2 Dynamics through Newton-Euler Approach

The mathematical representation of an airship's flight dynamics describes the different forces and moments acting on the vehicle during the flight. As this type of aerial vehicles are filled by a light lifting gas, e.g. helium, the added mass and added inertia effects arise due to the fact that the airship's mass is of the same order of magnitude as the mass of displaced air [11, 17, 98]. These effects are presented as forces and moments with respect to linear and angular accelerations [11].
There are two different approaches used in the derivation of the 6DOF nonlinear dynamical model of the lighter than air vehicles[17]. The first one is based on Newton-Euler's law, while the second one deals with the Lagrange-Hamilton framework.

The Newton-Euler approach in deriving dynamic equations is basically based on Newton's law of motion which relates the forces and moments applied on the vehicle to the resulting translational and rotational accelerations. Hence, for the translational motion we have

$$\sum \overrightarrow{F}_{ext} = m.\overrightarrow{a} \tag{2.14}$$

Where, m denotes the total mass of the vehicle, $\overrightarrow{a}$ presents its linear acceleration and $\overrightarrow{F_{ext}}$ expresses the generalized external force vector.

On the other hand, applying Newton-Euler principle and Koenig theorem in rotational motion leads to

$$\sum \overrightarrow{M}_{ext} = \frac{d\Pi_V}{dt} + \overrightarrow{V}_V \times P \tag{2.15}$$

where M_{ext} describes the moment vector acting on C_V, $\overrightarrow{V}_V$ denotes the velocity at the center of volume, and Π_V presents the angular momentum with respect to C_V given by:

$$\begin{aligned} \Pi_V &= \Pi_G + \overrightarrow{C_V C_G} \times m\overrightarrow{V}_G \\ &= I_G \overrightarrow{V}_2 + \overrightarrow{C_V C_G} \times m\overrightarrow{V}_G \end{aligned} \tag{2.16}$$

with

- Π_G is the angular momentum with respect to the center of gravity.
- $\overrightarrow{C_V C_G}$ is the position vector of C_G with respect to C_V.
- $\overrightarrow{V}_G$ is the velocity of C_G.
- I_G is the inertia matrix of the airship at C_G.

Proceeding from Equation (2.14) and Equation (2.15), the lighter than air vehicle dynamic equations can be written as

$$M\dot{V} = C(V)V + \tau_s + \tau_a + \tau_p \tag{2.17}$$

Here, M presents the vehicle's inertia matrix, $C(V)$ denotes the Coriolis matrix, τ_s, τ_p, and τ_a refer to the static, propulsion system, and aerodynamic tensors respectively. The different components of Equation (2.17) are the subject of the following paragraphs.

Inertia matrix M This matrix is given by [17, 59]

$$M = \begin{bmatrix} mI_{3\times 3} & -m\left[r_{C_G}^{C_V}\times\right] \\ m\left[r_{C_G}^{C_V}\times\right] & I_N \end{bmatrix} \tag{2.18}$$

where, $I_{3\times 3}$ is the identity matrix, I_N is the inertia matrix with respect to C_V which, under the assumption of the vehicle's symmetry about $x_{\mathcal{B}} \times z_{\mathcal{B}}$ plane, takes the form

$$
\begin{aligned}
I_N &= I_G + m\overrightarrow{C_V C_G} \times \left(\overrightarrow{V}_2 \times \overrightarrow{C_V C_G}\right) \\
&= \begin{bmatrix} I_{xx} & 0 & -I_{xz} \\ 0 & I_{yy} & 0 \\ -I_{zx} & 0 & I_{zz} \end{bmatrix}
\end{aligned} \tag{2.19}
$$

with I_{xx}, I_{yy}, and I_{zz} are called the moments of inertia with respect to x, y, and z axes respectively, while the rest of the elements are referred to as the products of inertia. On the other side, $\left[r_{C_G}^{C_V}\times\right]$ expresses the skew-symmetric matrix associated to the distance from C_V to C_G. It is given by

$$
\left[r_{C_G}^{C_V}\times\right] = \begin{bmatrix} 0 & -a_z & 0 \\ a_z & 0 & -a_x \\ 0 & a_x & 0 \end{bmatrix} \tag{2.20}
$$

where,

$$
\overrightarrow{C_V C_G} = \begin{bmatrix} a_x & 0 & a_z \end{bmatrix}^T \tag{2.21}
$$

Coriolis and Centrifugal tensor $C(V)V$ It takes the following form

$$
C(V)V = \begin{bmatrix} V_2 \times mV_1 - V_2 \times m\left[r_{C_G}^{C_V}\times\right] V_2 \\ V_2 \times I_c V_2 + V_1 \times mV_1 + V_2 \times m\left[r_{C_G}^{C_V}\times\right] V_1 - V_1 \times m\left[r_{C_G}^{C_V}\times\right] V_2 \end{bmatrix} \tag{2.22}
$$

Static tensor τ_s It resulted from the static forces applied on the airship independently of its motion. These forces are; the weight force $\overrightarrow{W}$ acting down-wards on the center of gravity, and the buoyancy force $\overrightarrow{B}$ acting up-wards on the center of buoyancy C_B. The magnitude of these forces is given by

$$
\|W\| = m.g \tag{2.23}
$$

$$
\|B\| = \rho.\nu_{hull}.g \tag{2.24}
$$

where, ρ refer to the density of the air and ν_{hull} denotes the volume of the airship's hull. On the other hand, the resulting moments are defined as

$$
\overrightarrow{M}_w = \overrightarrow{C_V C_G} \times \overrightarrow{g} \tag{2.25}
$$

$$
\overrightarrow{M}_B = \overrightarrow{C_V C_B} \times \overrightarrow{g} \tag{2.26}
$$

with, $\overrightarrow{C_V C_G}$, $\overrightarrow{C_V C_B}$ defines the position vector from the vehicle's center of volume C_V to its center of gravity C_G and buoyancy C_B respectively. Then, from previous relations, the static tensor in fixed-body coordinate system can be determined by

$$\tau_s = C_{\mathcal{I}}^{\mathcal{B}} \begin{bmatrix} \overrightarrow{W} + \overrightarrow{B} \\ \overrightarrow{C_V C_G} \times \overrightarrow{W} + \overrightarrow{C_V C_B} \times \overrightarrow{B} \end{bmatrix} \tag{2.27}$$

Aerodynamic Tensor τ_a The aerodynamics phenomena are basically depending on the relative velocity of the vehicle. It can be categorized into two main classes: stationary and non-stationary phenomena.

The stationary aspect is related to forces and moments resulting from the distribution of the pressure around the body, and the friction forces due to the viscosity of air [17, 126, 160]. Thus, the stationary phenomena tensor can be described as

$$\tau_{sta} = \frac{1}{2}\rho V^2 S_{ref} \begin{bmatrix} C_T \\ C_L \\ C_N \\ L_{ref} C_l \\ L_{ref} C_n \\ L_{ref} C_m \end{bmatrix} \tag{2.28}$$

where $S_{ref} = \nu_{hull}^{2/3}$ refer to the airship's reference area, L_{ref} presents the vehicle's reference length, and C_T, C_L, C_N, C_l, C_n, and C_m denote, tangential, normal, lateral, roll, pitch, and yaw stationary coefficients respectively. These coefficients depend on the geometry of the airship and the positions of the control surfaces [17]. In addition, it can be obtained in two different ways. The first one is based on an experimental procedure which consists on collecting data using wind tunnel. While, the second method uses an analytic estimation calculated with a geometric quantities procedure [7, 11, 116].

On the other hand, considering an airship having a mass m which moves along $x_{\mathcal{B}}$ axis with a linear acceleration of $\dot{u}$. The reaction of the surrounding air to this motion is expressed as a force of non-stationary nature, i.e. depending on acceleration. This force acts in the opposite direction of motion and it is proportional to the acceleration $\dot{u}$ [59, 77]. The previous discussion can be described mathematically by

$$X_A = -X_{\dot{u}}.\dot{u} \tag{2.29}$$

Introducing this force in the equation of motion allow us to combine the coefficient $X_{\dot{u}}$ linearly with the mass m of the vehicle. For this reason, $X_{\dot{u}}$ is named added mass coefficient along the $x_{\mathcal{B}}$ axis due to the acceleration $\dot{u}$ [9, 17, 26, 31]. Thus, the full

added mass and inertia matrix taking into account the symmetry of the vehicle with respect to $x_{\mathcal{B}} \times h_{\mathcal{B}}$ plane is defined by [59]

$$M_A = -\begin{bmatrix} X_{\dot{u}} & 0 & X_{\dot{w}} & 0 & X_{\dot{q}} & 0 \\ 0 & Y_{\dot{v}} & 0 & Y_{\dot{p}} & 0 & Y_{\dot{r}} \\ Z_{\dot{u}} & 0 & Z_{\dot{w}} & 0 & Z_{\dot{q}} & 0 \\ 0 & L_{\dot{v}} & 0 & L_{\dot{p}} & 0 & L_{\dot{r}} \\ M_{\dot{u}} & 0 & M_{\dot{w}} & 0 & M_{\dot{q}} & 0 \\ 0 & N_{\dot{v}} & 0 & N_{\dot{p}} & 0 & N_{\dot{r}} \end{bmatrix} \tag{2.30}$$

with $X_{\dot{w}} = Z_{\dot{u}} \ldots$. Moreover, the previous non-stationary phenomenon cause a Coriolis and Centripetal effects which is referred to by the following matrix

$$C_A(V) = \begin{bmatrix} 0 & 0 & 0 & 0 & -a_3 & a_2 \\ 0 & 0 & 0 & a_3 & 0 & -a_1 \\ 0 & 0 & 0 & -a_2 & a_1 & 0 \\ 0 & -a_3 & a_2 & 0 & -b_3 & b_2 \\ a_3 & 0 & -a_1 & b_3 & 0 & -b_1 \\ -a_2 & a_1 & 0 & -b_2 & b_1 & 0 \end{bmatrix} \tag{2.31}$$

where

$$a_1 = X_{\dot{u}}u + X_{\dot{w}}w + X_{\dot{q}}q \tag{2.32}$$
$$a_2 = Y_{\dot{v}}v + Y_{\dot{p}}p + Y_{\dot{r}}r \tag{2.33}$$
$$a_3 = X_{\dot{w}}u + Z_{\dot{w}}w + Z_{\dot{q}}q \tag{2.34}$$
$$b_1 = Y_{\dot{p}}p + L_{\dot{p}}p + L_{\dot{r}}r \tag{2.35}$$
$$b_2 = X_{\dot{q}}q + Z_{\dot{q}}w + M_{\dot{q}}q \tag{2.36}$$
$$b_3 = Y_{\dot{r}}v + L_{\dot{r}}p + N_{\dot{r}}r \tag{2.37}$$

Hence, the aerodynamic tensor is given by

$$\tau_a = \tau_{sta} + M_A\dot{V} + C_A(V)V \tag{2.38}$$

Propulsion system tensor τ_p The lighter than air vehicle is equipped with a propulsion system which generate the required thrust to drive the vehicle forward. Conventionally, it consists of two main propellers placed on each side of the gondola in order to maintain the vehicle's symmetry with respect to $x_{\mathcal{B}} \times z_{\mathcal{B}}$ plane. Their position is denoted by C_{P_1} and C_{P_2} for the first and the second engine respectively. These actuators provide a total thrust force of T, i.e. $\frac{1}{2}T$ each one. Their position vector with respect to $R_{\mathcal{B}}$ are

defined by

$$\overrightarrow{C_V C_{P_1}} = [P_x \quad P_y \quad P_z]^T \tag{2.39}$$
$$\overrightarrow{C_V C_{P_2}} = [P_x \quad -P_y \quad P_z]^T \tag{2.40}$$

Thus, we have

$$\tau_p = \begin{bmatrix} T \\ 0 \\ 0 \\ 0 \\ T.P_z \\ 0 \end{bmatrix} \tag{2.41}$$

From previous relations, the complete 6DOF nonlinear equations of motion for a lighter than air vehicle using Newton-Euler approach is given by Equation (2.9) and Equation (2.17). Remark that the previous modeling does not include the wind velocity. Assuming that the airship is moving in a windy environment in which the wind velocity takes the form

$$V_w = \begin{bmatrix} u_w & v_w & w_w & p_w & q_w & r_w \end{bmatrix}^T \tag{2.42}$$

Then, the lighter than air vehicle moves with the following relative velocity

$$V_a = V - V_w \tag{2.43}$$

2.3.1.3 Dynamics through Lagrangian Approach

The Lagrangian method in deriving the dynamic model of airships deals with two scalar energy function: kinetic energy E_K and potential energy E_P. It involves three basic steps. The first one is to formulate the expression of E_K and E_P. The second one consists of calculating the Lagrangian, denoted as $\mathcal{L}$, according to the following relationship

$$\mathcal{L} = E_K - E_P \tag{2.44}$$

While, the final step is to apply the Lagrange equation given by

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\eta}}\right) - \frac{\partial \mathcal{L}}{\partial \eta} = F \tag{2.45}$$

which in component form corresponds to a set of 6 second-order differential equations. Notice that, Equation (2.45) is valid in any coordinate system, Earth fixed inertial and fixed body frame as long as generalized coordinates are used, i.e. η [59].

For more details about the methods used above in modeling the lighter than air vehicle's equations of motion, i.e. Newton-Euler and Lagrangian, we invite you to refer [7, 11, 17, 59, 77, 116, 126, 160].

2.3.2 Equations of Motion for Quad-rotors

Quad-rotor is a type of aerial vehicle belonging to rotor-craft family. Contrary to lighter than air vehicles, the lift needed to keep flight in rotor-crafts is produced aerodynamically through rotating wings. A quad-rotor vehicle has four rotors. The front and the rear rotors rotate counterclockwise, while the other two rotors rotate clockwise. In this manner, the gyroscopic effects and the aerodynamic torques tend to cancel in trimmed flight. The force produced by each rotor is proportional to its angular velocity, and the sum of these forces gives the main thrust of the aerial vehicle, see Figure (2.4). Hence, the quad-rotor is an under-actuated system because it has four inputs and six degrees of freedom. The variation of the angular speed of each rotor allow to control the vehicle. The pitch movement is obtained by increasing/decreasing the speed of the rear motor while decreasing/increasing the speed of the front motor. The roll movement is obtained similarly using the lateral motors. Whereas, the yaw movement is obtained by increasing/decreasing the speed of the front and rear motors while decreasing/increasing the speed of the lateral motors. These motions can be accomplished whilst keeping the main thrust of the vehicle constant.

In order to deduce the mathematical model of the quad-rotor, the following assumption is considered:

- The vehicle is assumed to be a solid body evolving in 3D space and subject to one force and three moments.
- The dynamic of the four rotors is relatively fast and therefore it will be neglected as well as the flexibility of the blades.
- The center of gravity is supposed to be located at the intersection of the line joining motors M_1 and M_3 and the line joining motors M_2 and M_4, see Figure (2.4).

The generalized coordinates of the rotor-craft are

$$\varphi = (\zeta, \eta) \in \mathbb{R}^6 \tag{2.46}$$

where $\zeta = (x, y, z) \in \mathbb{R}^3$ denotes the position of the center of gravity of the vehicle relative to the frame $R_{\mathcal{I}}$, and $\eta = (\psi, \theta, \phi) \in \mathbb{S}^3$ presents Euler angles defined in the subsection 2.2.3. These angles define the orientation of the quad-rotor [32, 33, 127]. Notice that ζ and η can be regarded as translational and rotational coordinates respectively.

The full quad-rotor dynamic model is obtained using Euler-Lagrange approach.

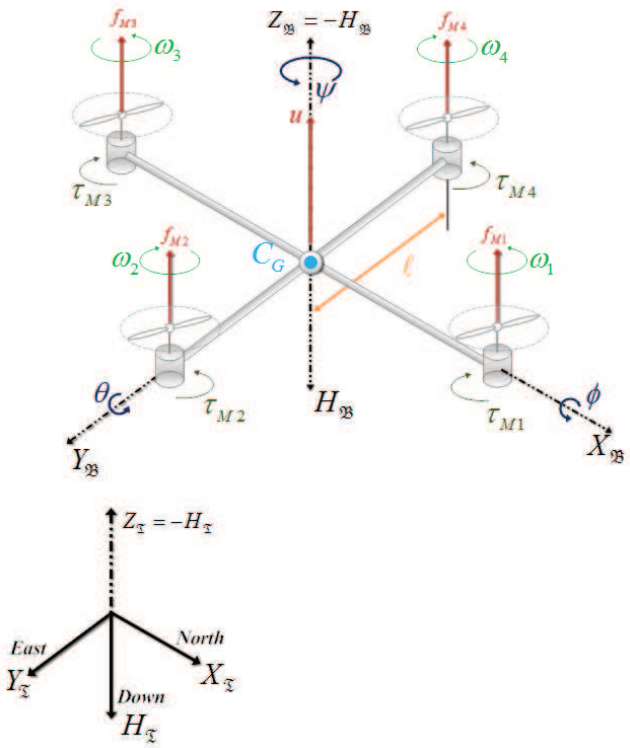

FIGURE 2.4: The model of quad-rotor.

The kinetic energy of the rotor-craft is determined by

$$E_K = T_{trans} + T_{rot} \tag{2.47}$$

where, T_{trans} and T_{rot} refer to the translational and rotational kinetic energy respectively, and they are given by

$$T_{trans} \triangleq \frac{m}{2}\dot{\zeta}^T\dot{\zeta} \tag{2.48}$$

$$T_{rot} \triangleq \frac{1}{2}\dot{\eta}^T\mathbb{J}\dot{\eta} \tag{2.49}$$

with, m denotes the mass of the vehicle, and $\mathbb{J}$ presents the inertia matrix for the full rotational kinetic energy of the rotor-craft expressed directly in terms of the generalized coordinates φ. On the other hand, the potential energy is

$$E_P = mgz \tag{2.50}$$

here, z presents the altitude of the aerial vehicle, and g describes the acceleration due to gravity.

Therefore, using Equation (2.44) the Lagrangian takes the form

$$\begin{aligned} \mathcal{L}(\varphi, \dot{\varphi}) &= T_{trans} + T_{rot} - E_P \\ &= \frac{m}{2}\dot{\zeta}^T\dot{\zeta} + \frac{1}{2}\dot{\eta}^T\mathbb{J}\dot{\eta} - mgz \end{aligned} \tag{2.51}$$

The external generalized force applied on the four-rotor aerial vehicle are presented by the following vector

$$F = \begin{bmatrix} F_\zeta \\ \tau \end{bmatrix} \tag{2.52}$$

where

$$F_\zeta = C_{\mathcal{I}}^{\mathcal{B}}\widehat{F} \in \mathbb{R}^3 \tag{2.53}$$

is the translational force applied to the rotor-craft due to the main thrust, and

$$\tau \triangleq \begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} \in \mathbb{R}^3 \tag{2.54}$$

represents the yaw, pitch, and roll moments. Finally, the rotation matrix $C_{\mathcal{I}}^{\mathcal{B}}$ is defined in Equation (2.2).
From Figure (2.4), it follows that

$$\widehat{F} = \begin{bmatrix} 0 \\ 0 \\ u \end{bmatrix} \tag{2.55}$$

where u is the main thrust of the vehicle and it is given by

$$u = \sum_{i=4}^{4} f_{Mi}; \quad f_{Mi} = k_i w_i^2 \tag{2.56}$$

with $k_i > 0$ is a constant and w_i is the angular speed of the ith motor. On the other side, the generalized torques are expressed as

$$\tau_\psi = \sum_{i=1}^{4} \tau_{Mi} \tag{2.57}$$

$$\tau_\theta = \ell\,(f_{M2} - f_{M4}) \tag{2.58}$$

$$\tau_\phi = \ell\,(f_{M3} - f_{M1}) \tag{2.59}$$

here, ℓ is the distance from the motor to the center of gravity and τ_{Mi} is the torque produced by the ith motor.

Therefore, applying the Euler-Lagrange mechanics, Equation (2.45), leads to the following

$$m\ddot{\zeta} + \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} = F_\zeta \tag{2.60}$$

$$\mathbb{J}\ddot{\eta} + C(\eta, \dot{\eta})\dot{\eta} = \tau \tag{2.61}$$

where,

$$C(\eta, \dot{\eta}) = \dot{\mathbb{J}} - \frac{1}{2}\frac{\partial}{\partial \eta}\left(\dot{\eta}^T \mathbb{J}\right) \tag{2.62}$$

is referred to as the Coriolis terms and it contains the gyroscopic and centrifugal terms associated with η and depending on $\mathbb{J}$.
Finally, from Equation (2.60) and Equation (2.61) we obtain

$$m\ddot{x} = -u\sin\theta \tag{2.63a}$$

$$m\ddot{y} = u\cos\theta\sin\phi \tag{2.63b}$$

$$m\ddot{z} = u\cos\theta - mg \tag{2.63c}$$

$$\ddot{\psi} = \widetilde{\tau}_\psi \tag{2.63d}$$

$$\ddot{\theta} = \widetilde{\tau}_\theta \tag{2.63e}$$

$$\ddot{\phi} = \widetilde{\tau}_\phi \tag{2.63f}$$

where, $\widetilde{\tau}_\psi$, $\widetilde{\tau}_\theta$, and $\widetilde{\tau}_\phi$ denote the yawing moment, pitching moment, and rolling moment respectively. The relation between these moments and the torques τ_ψ, τ_θ, and τ_ϕ is given by

$$\widetilde{\tau} = \begin{bmatrix} \widetilde{\tau}_\psi \\ \widetilde{\tau}_\theta \\ \widetilde{\tau}_\phi \end{bmatrix} = \mathbb{J}^{-1}\left(\tau - C(\eta, \dot{\eta})\dot{\eta}\right) \tag{2.64}$$

For more details see [32, 33, 127].

2.4 3DOF Equations of motion (Point Mass Model)

As we have seen previously, the attitude of an aerial vehicle is defined by its position, orientation, and velocity. The value of these variables at a specific moment of time constitute a vector known as configuration vector q. In order to reach a specified/final configuration q_f from the actual/initial one q_0, the time profile of the aircraft's attitude variables must be determined at all the movement time. Therefore, from trajectory generation point of view, the aerial vehicle is presented as a point (its reference point

A_{RP}) and only its translational equations are considered. This modeling is known as ***Point Mass Model*** and it describes the inertial velocity vector $\overrightarrow{V}$ with respect to $R_{\mathcal{I}}$ and the external forces acting on the vehicle.

2.4.1 Assumptions

The derivation of the equations of motion for an aerial vehicle, considered as a point mass, flying inside the Earth's atmosphere requires the following assumptions [81, 82]

- The Earth is flat, its rotational velocity is neglected and the acceleration of the gravity is constant and perpendicular to the surface of the Earth.
- The flight is symmetric which involves that the sideslip angle β is assumed to be controlled to zero, and both the thrust force and the aerodynamic forces lie in the plane of symmetry of the vehicle. This assumption guarantees the mathematical accuracy of the point mass modeling.

2.4.2 Kinematic Equations

Kinematic equations are used to derive the differential equations for x, y and z which represent the location of the vehicle with respect to $R_{\mathcal{I}}$ [81]. Referring to Figure (2.5), we can state

$$\dot{\overrightarrow{r}} = \overrightarrow{V_I} \tag{2.65}$$

where $\overrightarrow{r}$ defines the position vector and $\overrightarrow{V_I}$ denotes the inertial velocity vector which has the form

$$\overrightarrow{V_I} = \overrightarrow{V} + \overrightarrow{V_w} \tag{2.66}$$

with $\overrightarrow{V}$ represents the relative velocity vector and $\overrightarrow{V_w}$ describes the wind velocity vector. In addition, $\overrightarrow{V} \in R_{\mathcal{I}}$ is defined by its magnitude V, the heading angle χ (measured from the North to the projection of $\overrightarrow{V}$ in $R_{\mathcal{L}}$), and the flight path angle γ (vertically up to $\overrightarrow{V}$). $\overrightarrow{V_w} \in R_{\mathcal{I}}$ is composed by $[W_x \; W_y \; W_z]^T$.

Therefore from the previous and Figure (2.5), the right hand of Equation (2.65) becomes

$$\overrightarrow{V_I} = (V\cos\chi\cos\gamma + W_x)\,\overrightarrow{N} + (V\sin\chi\cos\gamma + W_y)\,\overrightarrow{E} + (-V\sin\gamma - W_z)\,\overrightarrow{D} \tag{2.67}$$

On the other hand, referring to Figure (2.5), we can state

$$\overrightarrow{r} = x\overrightarrow{N} + y\overrightarrow{E} - z\overrightarrow{D} \tag{2.68}$$

Where, x and y define the down-range and cross-range respectively, and z denotes the altitude.

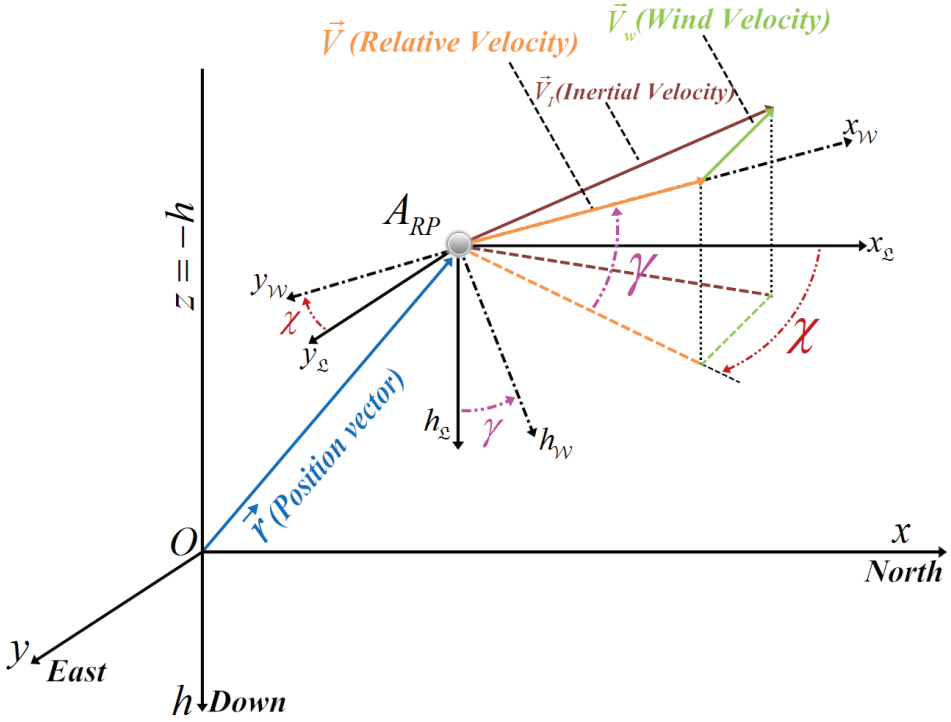

FIGURE 2.5: Aerial Vehicle's Inertial Velocity.

Since the unit vectors $\overrightarrow{N}$, $\overrightarrow{E}$, and $\overrightarrow{D}$ are constants, Equation (2.65) take the form

$$(V\cos\chi\cos\gamma+W_x)\,\overrightarrow{N}+(V\sin\chi\cos\gamma+W_y)\,\overrightarrow{E}-(V\sin\gamma+W_z)\,\overrightarrow{D}=\dot{x}\overrightarrow{N}+\dot{y}\overrightarrow{E}-\dot{z}\overrightarrow{D} \tag{2.69}$$

Then, the previous equation leads to the following

$$\dot{x} = V\cos\chi\cos\gamma + W_x \tag{2.70a}$$
$$\dot{y} = V\sin\chi\cos\gamma + W_y \tag{2.70b}$$
$$\dot{z} = -\dot{h} = V\sin\gamma + W_z \tag{2.70c}$$

These equations define the three-dimensional translational kinematic equations of motion for an aerial vehicle taking into account the wind effect.

2.4.3 Dynamic Equations

The dynamic equations are used to deduce the differential equations that describe the vehicle's accelerations. It is derived from Newton's second law of motion

$$m\overrightarrow{a} = \sum \overrightarrow{F} \tag{2.71}$$

where m represents the mass of the vehicle, $\overrightarrow{a} = \dot{\overrightarrow{V}}_I$ denotes the inertial acceleration, and $\overrightarrow{F}$ describes the external forces acting on the aerial vehicle. Observe that, the Equation (2.71) is written with respect to the Earth fixed inertial frame, because it represents a convenient base to follow the motion of the vehicle. In addition, Remember that the relative velocity vector is described in $R_{\mathcal{W}}$ and this frame is a rotating coordinate system in which Newton's Laws do not apply. Hence, we need a formula which transform the time derivative between a fixed coordinate system and rotating one [165].

Relative Angular Motion Considering two coordinate systems $O_1x_1y_1z_1$ and $Oxyz$ the first one is fixed while the second one is rotating with respect to the first one with an angular velocity ω. Let $\overrightarrow{i}$, $\overrightarrow{j}$, and $\overrightarrow{k}$ be the unit vectors along the axes of the rotating system Figure (2.6) and $\overrightarrow{A}$ be an arbitrary vector expressed as following

$$\overrightarrow{A} = A_x \overrightarrow{i} + A_y \overrightarrow{j} + A_z \overrightarrow{k} \tag{2.72}$$

where, A_x, A_y, and A_z denote the components of $\overrightarrow{A}$ along the rotating axes. Since $Oxyz$ is rotating, its associated unit vectors $\overrightarrow{i}$, $\overrightarrow{j}$, and $\overrightarrow{k}$ are function of time. Thus, the time derivative of $\overrightarrow{A}$ with respect to the fixed coordinate system is given as

$$\frac{d\overrightarrow{A}}{dt} = \left(\frac{dA_x}{dt}\overrightarrow{i} + \frac{dA_y}{dt}\overrightarrow{j} + \frac{dA_z}{dt}\overrightarrow{k}\right) + \left(A_x\frac{d\overrightarrow{i}}{dt} + A_y\frac{d\overrightarrow{j}}{dt} + A_z\frac{d\overrightarrow{k}}{dt}\right) \tag{2.73}$$

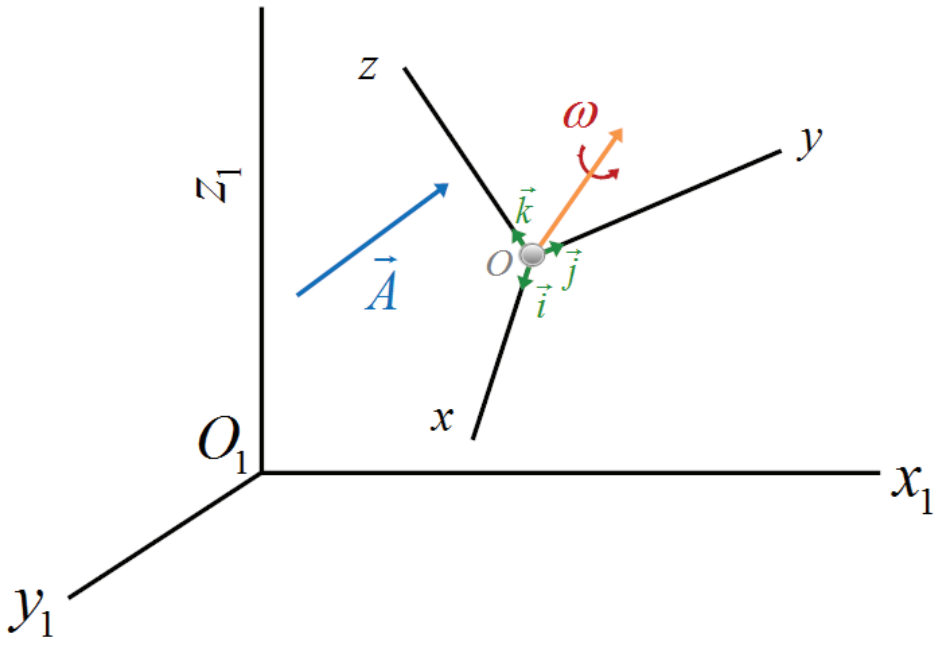

FIGURE 2.6: Relative Angular Motion.

It is known that the linear velocity at a fixed point P in a rotating frame Figure (2.7) is given by

$$\overrightarrow{V} = \frac{d\overrightarrow{r}}{dt} = \overrightarrow{\omega} \times \overrightarrow{r} \tag{2.74}$$

where $\overrightarrow{r}$ refer to the position vector. From the previous we can state

$$\frac{d\overrightarrow{i}}{dt} = \overrightarrow{\omega} \times \overrightarrow{i} \tag{2.75}$$

$$\frac{d\overrightarrow{j}}{dt} = \overrightarrow{\omega} \times \overrightarrow{j} \tag{2.76}$$

$$\frac{d\overrightarrow{k}}{dt} = \overrightarrow{\omega} \times \overrightarrow{k} \tag{2.77}$$

Using Equation (2.75), the second term on the right side of Equation (2.73) becomes

$$A_x \frac{d\overrightarrow{i}}{dt} + A_y \frac{d\overrightarrow{j}}{dt} + A_z \frac{d\overrightarrow{k}}{dt} = \overrightarrow{\omega} \times \overrightarrow{A} \tag{2.78}$$

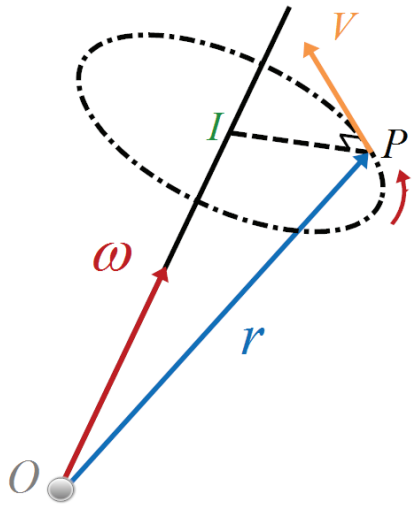

FIGURE 2.7: Kinematics of Rotation.

Besides, as the unit vectors $\overrightarrow{i}$, $\overrightarrow{j}$, and $\overrightarrow{k}$ are constant with respect to the rotating frame. The first term on the right side of Equation (2.73) corresponds to the derivative of $\overrightarrow{A}$ with respect to $Oxyz$ and it is denoted as $\frac{\partial \overrightarrow{A}}{\partial t}$. Hence, the transformation relationship of the time derivative from a rotating coordinate system to a fixed one is given by

$$\left.\frac{d\overrightarrow{A}}{dt}\right|_{fixed} = \left.\frac{\partial \overrightarrow{A}}{\partial t}\right|_{rotating} + \overrightarrow{\omega} \times \overrightarrow{A} \tag{2.79}$$

Coming back to the vehicle's acceleration $\overrightarrow{a}$ in Equation (2.71). From Equation (2.66) we find

$$\overrightarrow{a} = \frac{d\overrightarrow{V_I}}{dt} = \frac{d\overrightarrow{V}}{dt} + \frac{d\overrightarrow{V_w}}{dt} \tag{2.80}$$

Then, using Equation (2.79) we can write

$$\left.\frac{d\overrightarrow{V}}{dt}\right|_{R_\mathcal{I}} = \left.\frac{\partial\overrightarrow{V}}{\partial t}\right|_{R_\mathcal{W}} + \overrightarrow{\omega}_{R_\mathcal{W}/R_\mathcal{I}} \times \overrightarrow{V} \tag{2.81}$$

Where, $\overrightarrow{\omega}_{R_\mathcal{W}/R_\mathcal{I}}$ denotes the angular rotation of $R_\mathcal{W}$ with respect to $R_\mathcal{I}$. This angular velocity is expressed by

$$\overrightarrow{\omega}_{R_\mathcal{W}/R_\mathcal{I}} = \underbrace{\begin{bmatrix} 0 \\ \dot{\gamma} \\ 0 \end{bmatrix}}_{\Omega_{R_\mathcal{W}/y}} + \underbrace{\begin{bmatrix} \cos\gamma & 0 & -\sin\gamma \\ 0 & 1 & 0 \\ \sin\gamma & 0 & \cos\gamma \end{bmatrix}}_{\mathrm{Rot}_{R_\mathcal{W}/y}} \cdot \underbrace{\begin{bmatrix} 0 \\ 0 \\ \dot{\chi} \end{bmatrix}}_{\Omega_{R_\mathcal{W}/h}} = \begin{bmatrix} -\dot{\chi}\sin\gamma \\ \dot{\gamma} \\ \dot{\chi}\cos\gamma \end{bmatrix} \tag{2.82}$$

with $\Omega_{R_\mathcal{W}/y}$ and $\Omega_{R_\mathcal{W}/h}$ present the angular velocity of $R_\mathcal{W}$ about $y_\mathcal{W}$ and $h_\mathcal{W}$ respectively, and $\mathrm{Rot}_{R_\mathcal{W}/y}$ describes the rotation matrix of $R_\mathcal{W}$ about $y_\mathcal{W}$. Thus, Equation (2.81) becomes

$$\frac{d\overrightarrow{V}}{dt} = \begin{bmatrix} \dot{V} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -\dot{\chi}\sin\gamma \\ \dot{\gamma} \\ \dot{\chi}\cos\gamma \end{bmatrix} \times \begin{bmatrix} V \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{V} \\ \dot{\chi}V\cos\gamma \\ -\dot{\gamma}V \end{bmatrix} \tag{2.83}$$

The wind rate term, i.e. the second term of the right side in Equation (2.80), is given in $R_\mathcal{I}$ by

$$\left.\frac{dV_w}{dt}\right|_{R_\mathcal{W}} = C_\mathcal{L}^\mathcal{W} \left.\frac{dV_w}{dt}\right|_{R_\mathcal{I}} \tag{2.84}$$

with

$$\left.\frac{dV_w}{dt}\right|_{R_\mathcal{I}} = \begin{bmatrix} \dot{W}_x \\ \dot{W}_y \\ \dot{W}_z \end{bmatrix}$$

In the following section, we will examine the external forces acting on lighter than air vehicles.

2.4.3.1 Translational Dynamic of Lighter Than Air Vehicles

As we have seen previously, in the section 2.3.1.2, the external forces acting on the aircraft are: the added mass force (f_{AM}), the gravity force (f_g), the aerodynamical force

(f_a) and the thrust force (f_T). For Point Mass Modeling method, it is more convenient to describe these forces in $R_{\mathcal{W}}$, see Figure (2.8).

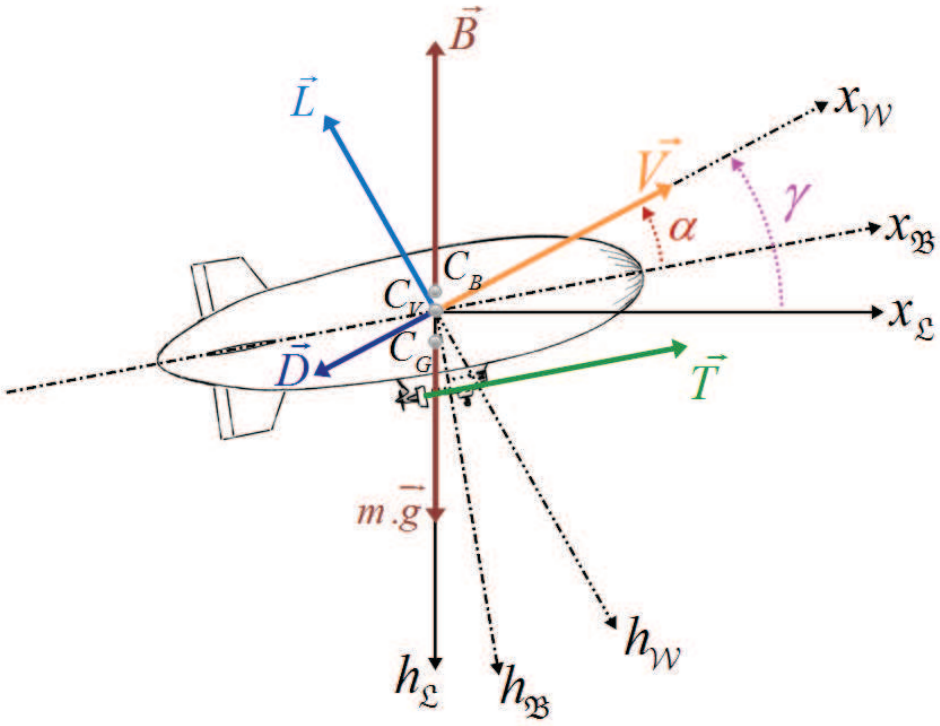

FIGURE 2.8: Forces acting on Lighter than air vehicles in flight.

Added mass phenomena: It is given by the relation

$$F_{AM} = -(C_{\mathcal{W}}^{\mathcal{B}})^T M_a C_{\mathcal{W}}^{\mathcal{B}} \frac{dV}{dt} \tag{2.85}$$

where M_a presents the added mass matrix resulting from the linear accelerations applied on the aerial vehicle. It takes the form

$$M_a = \begin{bmatrix} X_{\dot{u}} & 0 & X_{\dot{w}} \\ 0 & Y_{\dot{v}} & 0 \\ Z_{\dot{u}} & 0 & Z_{\dot{w}} \end{bmatrix} \tag{2.86}$$

with $X_{\dot{w}} = Z_{\dot{u}}$

$(C_{\mathcal{W}}^{\mathcal{B}})^T M_a C_{\mathcal{W}}^{\mathcal{B}}$ is a positive definite matrix with a main diagonal dominant. This matrix can be approximated as following

$$(C_{\mathcal{W}}^{\mathcal{B}})^T M_a C_{\mathcal{W}}^{\mathcal{B}} = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{33} \end{bmatrix}$$

with

$$\begin{aligned} m_{11} &= X_{\dot{u}} \cos^2\alpha + (Z_{\dot{u}} + X_{\dot{w}}) \cos\alpha \sin\alpha + Z_{\dot{w}} \sin^2\alpha \\ m_{22} &= Y_{\dot{v}} \\ m_{33} &= X_{\dot{u}} \sin^2\alpha - (Z_{\dot{u}} + X_{\dot{w}}) \cos\alpha \sin\alpha + Z_{\dot{w}} \cos^2\alpha \end{aligned}$$

Thus, the added mass force equation yields

$$F_{AM} = - \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{33} \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\chi} V \cos\gamma \\ -\dot{\gamma} V \end{bmatrix} \tag{2.87}$$

Gravity force: It is given by the difference between the airship weight force $\overrightarrow{W}$, and the buoyancy force $\overrightarrow{B}$. Then the gravity force f_g is expressed as

$$f_g = C_{\mathcal{L}}^{\mathcal{W}} \begin{pmatrix} 0 \\ 0 \\ W - B \end{pmatrix} = \begin{pmatrix} -(W - B)\sin\gamma \\ 0 \\ (W - B)\cos\gamma \end{pmatrix} \tag{2.88}$$

where, W and B are defined in the same way as in Equation (2.23) and Equation (2.24).

Aerodynamical force: This force includes the drag force $\overrightarrow{D}$, opposite to $\overrightarrow{V}$, and the lift force $\overrightarrow{L}$, orthogonal to $\overrightarrow{V}$, see Figure (2.8). These forces are expressed by the following equations

$$\begin{aligned} L &= \tfrac{1}{2} C_L(M, \alpha) V^2 S_{ref} \rho \\ D &= \tfrac{1}{2} C_D(M, \alpha) V^2 S_{ref} \rho \end{aligned} \tag{2.89}$$

with C_L and C_D denote the lift and drag parameters, respectively, M represents the Mach number and α the attack angle.

On the other hand, atmospheric density is computed using the standard atmosphere. Generally, the lift coefficient is a linear function of the angle of attack whilst the drag

coefficient is a quadratic function of $C_L(M,\alpha)$.

$$C_L(M,\alpha) = C_{L_0}(M) + k_{L_\alpha}(M)\alpha \tag{2.90a}$$
$$C_D(M,\alpha) = C_{D_0} + KC_L^2 = k_{D_0}(M) + k_{D_1}(M)\alpha + k_{D_2}(M)\alpha^2 \tag{2.90b}$$
$$\tag{2.90c}$$

where C_{L_0}, C_{L_1} and $k_{D_0}, k_{D_1}, k_{D_2}$ are resulting coefficients with respect to α. The induced drag factor K can be determined from the aerodynamic efficiency E_{max} and the zero-lift drag coefficient C_{D_0} as

$$K = \frac{1}{4C_{D_0}E_{max}^2}$$

Observe that the model parameters can be estimated via wind tunnel experiments, more details, see [165]. Therefore, f_a is denoted by

$$f_a = \begin{bmatrix} -D \\ 0 \\ -L \end{bmatrix} \tag{2.91}$$

Thrust force: In a symmetrical flight, the thrust vector $\overrightarrow{T}$ is always in the plane of symmetry. This force is described in $R_{\mathcal{W}}$ as

$$f_T = C_{\mathcal{B}}^{\mathcal{W}} \begin{bmatrix} T \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} T\cos\alpha \\ 0 \\ -T\sin\alpha \end{bmatrix} \tag{2.92}$$

In order to determine the equations of motion, it is more convenient to associate the aerodynamic and thrust forces. Then

$$f_T + f_a = \begin{bmatrix} T\cos\alpha - D \\ 0 \\ -(T\sin\alpha + L) \end{bmatrix} \tag{2.93}$$

Notice from the previous equation that the aerodynamic and thrust forces contain two components; $F_T = T\cos\alpha - D$, force along the velocity vector and, $F_N = T\sin\alpha + L$, force orthogonal to the velocity in the symmetry plane.

In longitudinal flight, F_N is in the vertical plane of the vehicle and the lateral force can be neglected. In order to perform a turning flight, the lateral force must be included. This force is obtained by a control action on the ailerons when rotating the vector $\overrightarrow{L}$ and

$\overrightarrow{F_N}$ about $\overrightarrow{V}$ through an angle σ called bank angle, see Figure (2.9). Mathematically the previous is written as

$$f_{aT} = \begin{bmatrix} T\cos\alpha - D \\ (T\sin\alpha + L)\sin\sigma \\ -(T\sin\alpha + L)\cos\sigma \end{bmatrix} \tag{2.94}$$

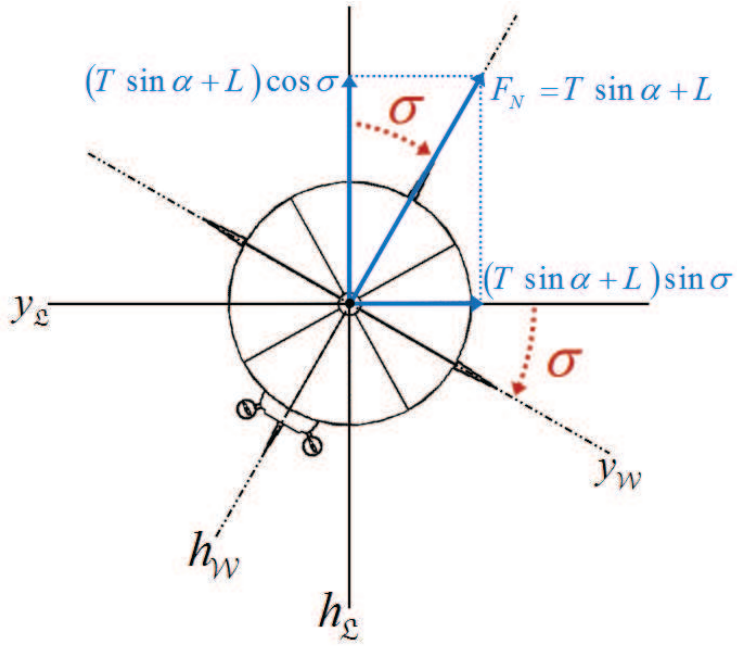

FIGURE 2.9: Forces acting on Airships in turning flight.

Then, from Equations (2.80), (2.83), (2.84), (2.87), (2.88) and (2.94) we obtain

$$\dot{\gamma} = \frac{(T\sin\alpha + L)\cos\sigma - \cos\gamma(B - mg)}{(m + m_{33})V} + m\frac{\dot{W}_x\cos\chi\sin\gamma + \dot{W}_y\sin\chi\sin\gamma + \dot{W}_z\cos\gamma}{(m + m_{33})V} \tag{2.95a}$$

$$\dot{\chi} = \frac{(T\sin\alpha + L)\sin\sigma}{(m + m_{22})V\cos\gamma} + m\frac{\dot{W}_x\sin\chi - \dot{W}_y\cos\chi}{(m + m_{22})V\cos\gamma} \tag{2.95b}$$

$$\dot{V} = \frac{T\cos\alpha - D - \sin\gamma(B - mg)}{m + m_{11}} - \frac{m}{m + m_{11}}(\dot{W}_x\cos\gamma\cos\chi + \dot{W}_y\sin\chi\cos\gamma - \dot{W}_z\sin\gamma) \tag{2.95c}$$

Therefore, Equation (2.70) and Equation (2.95) present the complete Point Mass Model describing the translational equations of motion for lighter than air vehicles.

2.4.4 Translational Equations of Motion for Quad-rotor crafts

According to literature [32, 33, 127, 162], the quad-rotor aerial vehicles was always regarded to as a rigid body and represented by its full 6DOF nonlinear translational and rotational dynamics (see section 2.3.2). In this section, a new approach using Point

Mass Model to describe the quad-rotor crafts is adopted. This modeling is employed to generate the reference trajectories for the aerial vehicle.

In general, the pitch angle is given as the sum of the flight path angle and the angle of attack [155]

$$\theta = \gamma + \alpha \tag{2.96}$$

But, for quad-rotor craft, the angle of attack is very small and consequently, it can be neglected ($\alpha \approx 0$). Then, Equation (2.96) becomes

$$\theta = \gamma \tag{2.97}$$

Besides, as the heading angle χ is measured from the north to the projection of $\overrightarrow{V}$ in $x_{\mathcal{L}} \times y_{\mathcal{L}}$, this angle can be regarded as the yaw angle ψ. Thus,

$$\chi = \psi \tag{2.98}$$

Therefore, from the previous, Equation (2.70) takes the following form

$$\dot{x} = V \cos\psi \cos\theta + W_x \tag{2.99a}$$
$$\dot{y} = V \sin\psi \cos\theta + W_y \tag{2.99b}$$
$$\dot{z} = V \sin\theta + W_z \tag{2.99c}$$

Remember that the vehicle accelerations, Equation (2.83), are given as

- Longitudinal acceleration: $a_{long} = \dot{V}$
- Lateral acceleration: $a_{lat} = V\dot{\chi}\cos\gamma$
- Vertical acceleration: $a_{vert} = V\dot{\gamma}$

Furthermore, notice, from Equation (2.70) and Equation (2.99), that γ, χ (θ, ψ for quad-rotors), and V define the attitude of the aerial vehicle. Consequently and without loss of generality, $\dot{\gamma}$, $\dot{\chi}$, and $\dot{V}$ can be considered as control inputs for lighter than air vehicle. So

$$\begin{aligned} \dot{\gamma} &= u_1 \\ \dot{\chi} &= u_2 \\ \dot{V} &= u_3 \end{aligned} \tag{2.100}$$

while $\dot{\theta}$, $\dot{\psi}$, and $\dot{V}$ are supposed to be the control inputs for quad-rotor craft. Then,

$$\begin{aligned} \dot{\theta} &= u_1 \\ \dot{\psi} &= u_2 \\ \dot{V} &= u_3 \end{aligned} \tag{2.101}$$

In order to incorporate the performance and structural limitations of a real airship/quad-rotor, appropriate constraints are included in the model. These restrictions take the form of bounds on control and some states variables, as follows

$$|u_1| \leq U_{1max} \tag{2.102a}$$
$$|u_2| \leq U_{2max} \tag{2.102b}$$
$$|u_3| \leq U_{3max} \tag{2.102c}$$
$$|\gamma| \leq \gamma_{max} \tag{2.102d}$$
$$|V| \leq V_{max} \tag{2.102e}$$

Remember that, for quad-rotors craft $\gamma = \theta$. Thus, Equation (2.102d) becomes

$$|\theta| \leq \theta_{max} \tag{2.103}$$

For more details see [81, 82, 155]. In the rest of this thesis, Equation (2.70) and Equation (2.100) will be used to describe the kinematic equations of motion for Lighter than air/Quad-rotors crafts.

2.5 Wind Modeling

Wind modeling has a significant role in the design of aerial vehicles. It helps to study and analyze the behavior of the aircraft facing the wind. In this work, we assume that the aerial vehicle flies in the troposphere layer. The air masses are in a constant motion and the region is characterized by gusty winds and turbulence.

As the movement of the air, in this layer, is similar to a fluid flowing over a solid object, the troposphere can be divided into two distinct regions. In the first, the effect of the Earth's surface friction on the air motion is negligible and it is known as the free atmosphere, while in the second one, the effect of viscosity cannot be neglected and it is called the boundary layer. The second region is typically extended over a several hundred meters to 2km roughly. It is depending on the landform and time of day/daytime, while the entire troposphere extends to 10-20 km approximately (it is larger in the tropics and shallower near the Polar Regions).

In general, the wind speed can be modeled as a sum of two components: a nominal deterministic component, available through meteorological forecasts or measured with a Doppler radar, and a stochastic component, representing deviations from the nominal

one [56, 105]. In the following, two classical mathematical models of gusts will be presented.

2.5.1 Gusts Modeling

Discrete Gusts: The most used approximation, representing a sharp edged gust, is one minus cosine formulation, see Figure (2.10). The gust velocity is defined as,

$$V_{wind} = \frac{1}{2} V_m \left[1 - cos\left(\frac{\pi x}{H}\right)\right] \tag{2.104}$$

where V_m denotes the gust amplitude, H is the distance from the start point to the point at which the gust reaches a maximum value and x represents the traveled distance. This formulation can be considered as a single representative section of the broader spectrum of continuous turbulence. However, in order to represent random continuous gusts that an aircraft may encounter, it became necessary to use statistical methods, particularly the method involving the power spectral density.

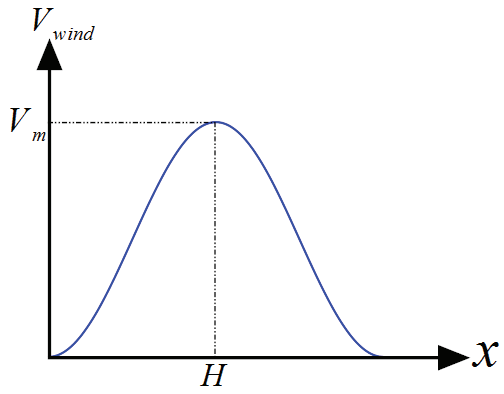

FIGURE 2.10: Discrete gust model using one minus cosine formula.

Power Spectral Density (PSD): It is based on the frequency representation for describing the square of a random variable which is originally considered in time domain. The turbulence model using this technique is assumed to be stationary, Gaussian, random process. There are two particular approximations for the PSD function of atmospheric turbulence: the von Kármán and Dryden formula, each proposing a separate function for gusts in the longitudinal, lateral and vertical directions. Dryden PSD function is more favored than von Kármán one because it is simpler. Thus, it is given as:

– Longitudinal wind velocity

$$\Phi_u(\omega) = \frac{2\sigma_u^2 L_u}{\pi V} \cdot \frac{1}{1 + \left(L_u \frac{\omega}{V}\right)^2} \tag{2.105}$$

– Lateral wind velocity

$$\Phi_v(\omega) = \frac{\sigma_v^2 L_v}{\pi V} \cdot \frac{1 + 3\left(L_v \frac{\omega}{V}\right)^2}{\left[1 + \left(L_v \frac{\omega}{V}\right)^2\right]^2} \tag{2.106}$$

– Vertical wind velocity

$$\Phi_w(\omega) = \frac{\sigma_w^2 L_w}{\pi V} \cdot \frac{1 + 3\left(L_w \frac{\omega}{V}\right)^2}{\left[1 + \left(L_w \frac{\omega}{V}\right)^2\right]^2} \tag{2.107}$$

where Φ_i describes the PSD function, σ_i represents the root-mean-square (RMS) gust velocity, L_i signifies the scale of turbulence and V is the aircraft velocity. The Figure (2.11) shows Dryden gust model with parameters $\sigma = 7$ m/sec and $L = 533.4$ m, more details see [55, 121, 132].

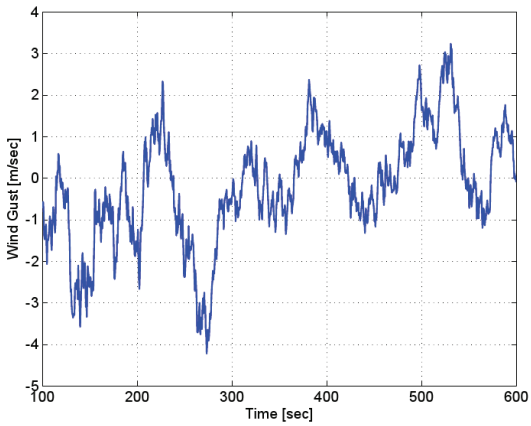

FIGURE 2.11: Dryden gust.

In this thesis, the gusty wind was generated using a combination of a several discrete gusts and a white noise as it is shown in the Figure (4.1).

2.5.2 Venturi effect

The Venturi effect was discovered in 1797 by Giovanni Battista Venturi. It applies to confined flows and refers to the increase in fluid speed or flow rate due to a decrease of the flow section, where the flow rate and flow cross-sectional area are inversely proportional [164]. Since the increase in fluid speed is generally accompanied by a decrease in pressure, the venturi effect is also used to refer to Bernoulli's principle. Currently this terminology

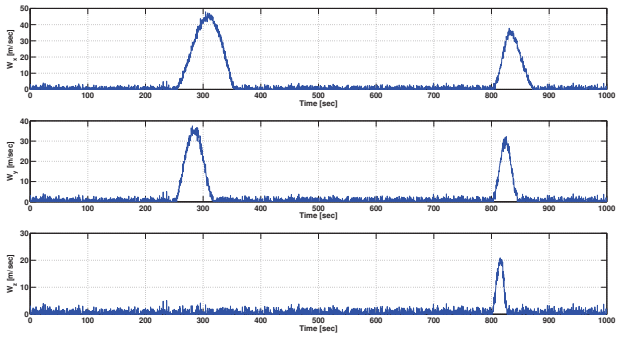

FIGURE 2.12: Wind gust applied to the aerial vehicle.

is being applied in the wider context of both confined and non-confined flows [24]. The term of "Venturi effect" can be seen in two different ways depending on the application areas. For example, in medicine and automotive industry it describes a pressure drop due to increase in speed. While, in wind engineering/urban aerodynamics as well as civil engineering "Venturi effect" means the increase in speed due to flow constriction.

The vicinity of bridges is an ideal environment to the appearance of Venturi effect due to its structural design. In fact, the passages between pillars can be responsible for increased wind speed, see Figure (2.13).

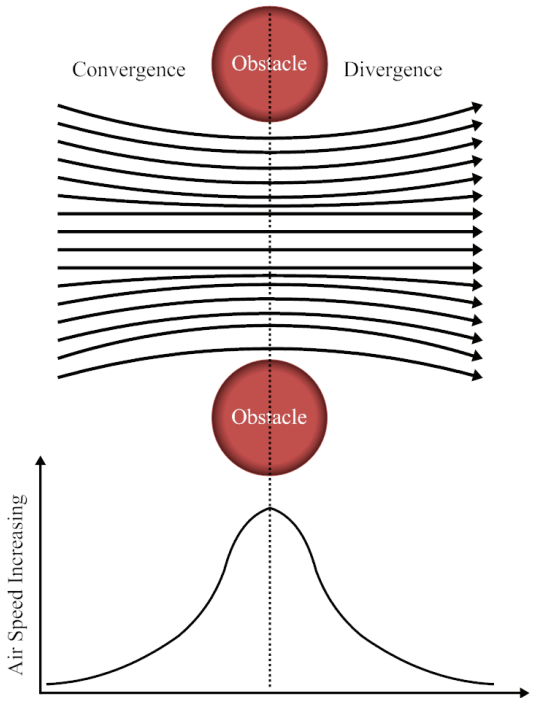

FIGURE 2.13: Venturi effect.

From Figure (2.13) observe that Venturi effect consists of:

- An increment in wind speed magnitude.
- A divergence and a convergence of air flow around the obstacle (pillar).

In order to represent mathematically the previous statement, we first introduce the following definitions

Definition 2.1. A Gaussian function is a continous function having the following form

$$\mathscr{G}(x) = a_{\mathscr{G}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{2.108}$$

where, $a_{\mathscr{G}}$ represents the height of the curve's peak, μ denotes the position of the center of the peak, and σ describes the deviation.

Definition 2.2. A Sigmoid function is a continous function having an "S" shape. It is given by

$$Sig(x) = \frac{1}{1 + e^{-a_{sig}(x-x_0)}} \tag{2.109}$$

where, a_{sig} defines the sharpness of the curve, and x_0 is the position of the switch, see Figure (2.14).

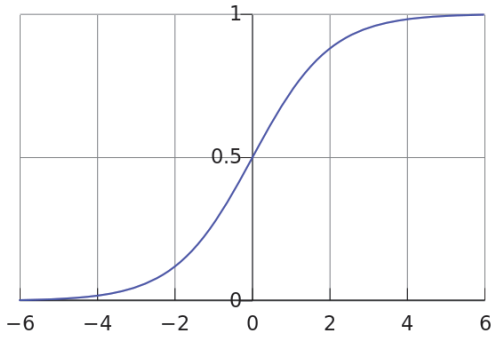

FIGURE 2.14: Sigmoid curve with $x_0 = 0$ and $a_{sig} = 1$.

Definition 2.3. We say that $\mathscr{E}(x)$ is an enable function on some interval $\{x_1, x_2\}$ if and only if

$$\mathscr{E}(x) = \begin{cases} 1 & ; \quad x_1 \leq x \leq x_2 \\ 0 & ; \quad \text{otherwise} \end{cases} \tag{2.110}$$

A continuous approximation for $\mathscr{E}(x)$ is given by

$$\tilde{\mathscr{E}}(x) = Sig(x_1) - Sig(x_2) \tag{2.111}$$

Considering the presence of two pillars, supposing a horizontal plan and assuming that the wind is in the direction of x-axis, see Figure (2.15). Then the wind velocity components on x and y axes are given as follows

$$W_x = \mathscr{G}(x) \tag{2.112a}$$

$$W_y = W_{y1} + W_{y2} \tag{2.112b}$$

where, W_{y1} and W_{y2} presents the component divergence and convergence of the air flow for the first and the second pillar respectively. These terms are defined as following

$$W_{y1} = a_y(y - y_{p1})\tilde{\mathscr{E}}(y_1)\left[\tilde{\mathscr{E}}(x_1) - \tilde{\mathscr{E}}(x_2)\right] \tag{2.113a}$$

$$W_{y2} = a_y(y - y_{p2})\tilde{\mathscr{E}}(y_2)\left[\tilde{\mathscr{E}}(x_1) - \tilde{\mathscr{E}}(x_2)\right] \tag{2.113b}$$

in the previous equation,

- y_{p1} and y_{p1} are the y-coordinate of the center of the first and the second pillar respectively.
- $\tilde{\mathscr{E}}(x_i)$ and $\tilde{\mathscr{E}}(y_i)$ (for $i = 1, 2$) identify the neighborhood of the first and second pillars on x and y axes respectively.
- the term $\tilde{\mathscr{E}}(x_1) - \tilde{\mathscr{E}}(x_2)$ associates the divergence and convergence areas, according to x axis, to the linear variation $a_y(y - y_{pi})$; $i = 1, 2$.

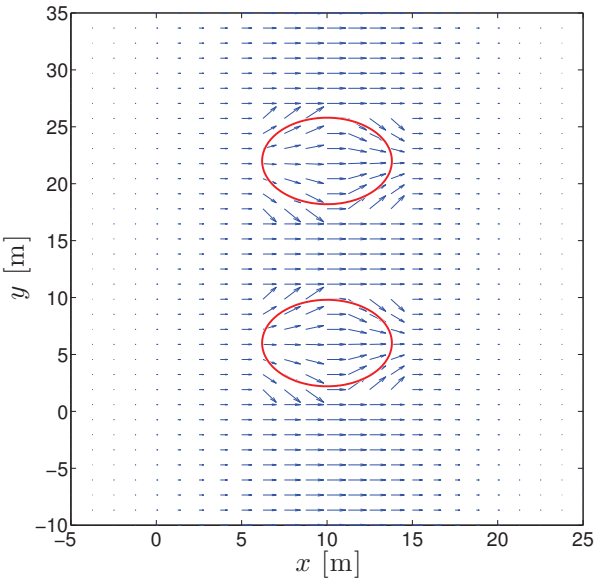

FIGURE 2.15: Venturi effect in the presence of two pillars.

2.6 Conclusion

In this chapter, general equations of motion for lighter than air vehicles and quadrotors crafts were introduced. The translational and rotational motion was uncoupled, and the vehicles were modeled in two different ways. Firstly, a six degree of freedom equations of motion, called rigid body model, was developed for lighter than air vehicle using Newton-Euler approach, and for quadrotor crafts through Euler-Lagrange technique. This model is devoted to treat the stability and control problems. Next, a three degree of freedom translational equations of motion, called point mass model, was introduced. This model is based on Newton's second law and used to deal with navigation and guidance control systems. Translational kinematics and dynamics for airships was realized, while the kinematic model was adopted to present quadrotors crafts.

Besides the mathematical representations of the aerial vehicles, the wind modeling issue has been addressed. Two models for gusts were presented. The first one represents a sharp edged gust, while the second one is based on the frequency representation. In addition, a mathematical relation modeling Venturi effect in the neighborhood of a bridge was developed.

After describing the equations of motion of lighter than air vehicles and quadrotors crafts, we address the questions of motion planning, trajectory generation, and tracking for aerial vehicles in next chapters.

Chapter 3

Trajectory Generation and Motion Planning

3.1 Introduction

One of the most important feature of an Unmanned Aerial Vehicle (UAV) is its capacity to accomplish different type of tasks with a high level of performance, maneuverability and with less oversight of human operators. This characteristic is also known as ***autonomy*** [12, 92]. These tasks require flexible and Powerful algorithms that convert high-level mission specifications from humans into low-level descriptions of the vehicle's motion. The terms ***Motion Planning*** and ***Trajectory Planning*** are often employed for such kind of problems [111].
In order to connect a starting and a target points, feasible and flyable trajectories must be defined. The feasibility criteria is carried out by motion planning algorithms. This process produce a plan to steer the UAV safely to its target, without taking into account its dynamical constraints. Whilst, the trajectory generation problem takes the solution obtained by the motion planning algorithm and determines the way to fly along this solution with respect to the vehicle's mechanical limitations. In other words, it guarantees the flyable aspect of the trajectory [12, 111].

The rest of this chapter is organized as follows: section 3.2 presents a state of art on trajectory generation and motion planning. Section 3.3 discusses solutions proposed for bridge inspection task. Finally, a conclusion is addressed in the section 3.4.

3.2 State of Art

3.2.1 Trajectory Generation

In addition to leading an aerial vehicle from an initial configuration to a final one and complying with the vehicle's physical constraints, there are often other requirements to take into account in the trajectory generation process. For instance, limiting curvature, reducing the traveled distance and consumed energy, etc...

In general, these are integrated in the design procedure as the optimization of a suitable cost criterion along the trajectory. The optimality of a trajectory can be defined according to several objectives, like minimizing the transfer time or the energy [85, 87]. Traditionally, trajectories are optimized by the application of numerical optimal control methods that are based on the calculus of variations. Dubins [51] considered a particle moving with a constant velocity in the plane and subjected a constraint on trajectory curvature. This problem is equivalent to the minimum time optimal trajectory problem under a constant velocity and limitation on heading rate, without wind. He proved the existence of shortest paths for his problem and showed that the optimal trajectory is one of the following six solutions: $\{RSL, RSR, LSR, LSL, RLR, LRL\}$ where R for Right turn, S for straight line and L for Left turn. Knowing that all sub paths are allowed to have zero length. So, his optimal trajectories are a combination of arc of circles and segments of lines. Boukraa et al [28] presented a 3D trim trajectories planner algorithm for an autonomous plane. The proposed algorithm used a sequence of five elementary trim trajectories to generate a 3D global trajectory in space. A family of trim trajectories in level flight is used in all these references to construct paths. In the papers cited above, the atmosphere was considered to be an isotropic and homogeneous medium, i.e. when there is no wind and the air density is constant with altitude.

However, the natural wind proved itself to be a major parameter to successful flights of aerial vehicles. It mostly affects a trajectory through its speed. McGee et al [120] have described a method for finding the minimum time path from an initial position and orientation to a final position and orientation in the 2D plane for an airplane with a bounded turning rate in the presence of a known constant wind with a magnitude less than the airplane velocity. The problem statement is equivalent to finding the minimum time path from an initial configuration to a final one, over a moving virtual target, where the velocity of the virtual target is equal and opposite to the velocity of the wind. Nelson et al [131] have introduced a method for a mini aerial vehicle path following based on the concept of vector field in the presence of constant wind disturbances. Rysdyk [143] has presented a path formulation for maneuvering of a fixed wing aircraft in wind. The inertial path of a fixed wing aircraft circling in wind can be formulated as a trochoid

curve. Jardin [83] has considered the minimum time problem for an airplane traveling horizontally between two points in a variable wind field. Bijlsma [22] has referred to route determination for an airplane traveling horizontally between two given points so the transition time is minimized assuming that the wind field is fully known beforehand for the complete passage. In these papers, only 2D horizontal motion was considered.

3.2.1.1 Space-Time Separation

Considering a given trajectory $C_q(t) \in \mathbb{R}^3$, for $t \in [0, T]$, which leads the aerial vehicle from an initial point $C_q(0) = C_{q0}$ to a final point $C_q(T) = C_{qf}$. This trajectory can be broken down into a geometric path $C_q(s) = [x(s) \quad y(s) \quad z(s)]^T$ and a timing law $s = s(t)$. Where s denotes the curvilinear abscissa and satisfies

$$s \in [0, L] \tag{3.1}$$

with, L: the length of the path.
Then, the velocity of the vehicle can be expressed as following

$$V = \frac{ds}{dt} \tag{3.2}$$

The previous space-time separation implies

$$\dot{C}_q = \frac{dC_q}{dt} = \frac{dC_q}{ds} \cdot \frac{ds}{dt} = C'_q \cdot V \tag{3.3}$$

In addition, each point of the path $C_q(s)$ can be described using three unitary orthogonal vectors: tangent ($\overrightarrow{T}$), normal ($\overrightarrow{N}$) and binormal ($\overrightarrow{B}$). These vectors form the so-called Frenet-Serret frame.

3.2.1.2 Curvature and Torsion

The importance of curvature κ and torsion τ comes from the fact that any curve in $\mathbb{R}^3$ can be completely defined by these variables [102].
The curvature is directly proportional to the lateral acceleration of the vehicle and, it is inversely proportional to the minimum curvature radius (R) that can be flew by the aircraft. It is expressed as following

$$\kappa = \frac{\|C'_q \times C''_q\|}{\|C'_q\|^3} \tag{3.4}$$

In the other side, torsion is proportional to the vertical acceleration which is also physically limited. In addition, the value of τ is inversely proportional to the minimum

torsion radius (ϕ). It is presented as

$$\tau = \frac{\left(C'_q \times C''_q\right) \cdot C'''_q}{\|C'_q \times C''_q\|^2} \tag{3.5}$$

The first derivative of the flight path $C'_q = [x' \quad y' \quad z']^T$ is deduced from equations (2.70) and (3.3) as following

$$x' = \cos\chi\cos\gamma + \frac{W_x}{V} \tag{3.6a}$$

$$y' = \sin\chi\cos\gamma + \frac{W_y}{V} \tag{3.6b}$$

$$z' = \sin\gamma + \frac{W_z}{V} \tag{3.6c}$$

While, the second and third derivatives are given as below

$$C''_q = \frac{\ddot{C}_q - VV'C'_q}{V^2} \tag{3.7}$$

$$C'''_q = \frac{\dddot{C}_q - 3V'V^2C''_q - \left(V''V + V'^2\right)C'_qV}{V^3} \tag{3.8}$$

In this manner, the curvature and torsion can formally be described as:

$$\kappa = f\left(\gamma, \chi, V, \gamma', \chi', V', W_x, W_y, W_z, W'_x, W'_y, W'_z\right) \tag{3.9a}$$

$$\tau = f\left(\begin{array}{c}\gamma, \chi, V, \gamma', \chi', V', \gamma'', \chi'', V'', W_x, \\ W_y, W_z, W'_x, W'_y, W'_z, W''_x, W''_y, W''_z\end{array}\right) \tag{3.9b}$$

where, for lighter than air vehicles, γ', χ' and V' are obtained directly by dividing the equation (2.95) by V as following

$$\gamma' = \frac{(T\sin\alpha + L)\cos\sigma - mg\cos\gamma}{mV^2} + \frac{\dot{W}_x\cos\chi\sin\gamma + \dot{W}_y\sin\chi\sin\gamma + \dot{W}_z\cos\gamma}{V} \tag{3.10a}$$

$$\chi' = \frac{(T\sin\alpha + L)\sin\sigma}{mV^2\cos\gamma} + \frac{\dot{W}_x\sin\chi - \dot{W}_y\cos\chi}{V\cos\gamma} \tag{3.10b}$$

$$V' = \frac{T\cos\alpha - D - mg\sin\gamma}{mV} - \dot{W}_x\cos\gamma\cos\chi - \dot{W}_y\sin\chi\cos\gamma + \dot{W}_z\sin\gamma \tag{3.10c}$$

Whilst, their second derivatives are given by

$$\gamma'' = \frac{\ddot{\gamma} - \gamma'V'V}{V^2} \tag{3.11a}$$

$$\chi'' = \frac{\ddot{\chi} - \chi'V'V}{V^2} \tag{3.11b}$$

$$V'' = \frac{\ddot{V} - V'^2V}{V^2} \tag{3.11c}$$

3.2.1.3 Constraints

A path is said to be flyable if it meets with the kinematic and dynamic constraints of the vehicle. Such motion constraints are introduced as bounds on the curvature (κ) and the torsion (τ) of the path $C_q(s)$

$$|\kappa| \leq \kappa_{max} \tag{3.12a}$$

$$|\tau| \leq \tau_{max} \tag{3.12b}$$

Notice that, κ_{max} and τ_{max} are established from constraints imposed on the control and state variables, Equation (2.102).

3.2.1.4 Cartesian Polynomials

After determining the required path $C_q(s)$; $s \in [s_0, s_f]$, the related timing law must be chosen so that resulting trajectory respects the restrictions imposed on the vehicle, Equation (2.102). For example, in order to take the inputs bounds introduced to the aircraft, Equation (2.102a), Equation (2.102b), and Equation (2.102c), into consideration, it is necessary to verify whether the inputs (linear acceleration and angular velocities) along the planned trajectory are admissible. When this is not achieved, a *uniform scaling* is used to slow down the timing law. To do that, it is more convenient to rewrite the timing law by replacing t with the normalized time variable $\mathcal{T} = t/T$. Then, from Equation (2.100), Equation (3.3) and Equation (3.2) we have

$$\gamma' = u_1 \frac{ds}{d\mathcal{T}} \frac{d\mathcal{T}}{dt} = u_1 \frac{ds}{d\mathcal{T}} \frac{1}{T} \tag{3.13}$$

$$\chi' = u_2 \frac{ds}{d\mathcal{T}} \frac{d\mathcal{T}}{dt} = u_2 \frac{ds}{d\mathcal{T}} \frac{1}{T} \tag{3.14}$$

$$V' = u_3 \frac{ds}{d\mathcal{T}} \frac{d\mathcal{T}}{dt} = u_3 \frac{ds}{d\mathcal{T}} \frac{1}{T} \tag{3.15}$$

Therefore, it is sufficient to increase the duration of the trajectory T to reduce uniformly γ', χ' and V' so that the vehicle's inputs remain within the given bounds.

Planning a trajectory directly without separating the geometric path from the timing law could be effected by using the time variable t directly instead of the path parameter s. A disadvantage of this approach is the fact that the duration of the trajectory must be fixed ($T = s_f$). Thus, the uniform scaling cannot be used to satisfy bounds on the vehicle's input. That is because an increment (or decrement) of T implies a modification on the geometric path associated to the planned trajectory [148].

Third-order Cartesian polynomials are known to be simple in real time implementation [49]. Then, for two successive configurations (x_i, y_i, z_i, γ_i, χ_i, V_i) and (x_{i+1}, y_{i+1}, z_{i+1}, γ_{i+1}, χ_{i+1}, V_{i+1}) we have:

$$x(s) = s^3 x_{i+1} - (s-1)^3 x_i + \alpha_x s^2(s-1) + \beta_x s(s-1)^2 \tag{3.16a}$$
$$x(s) = s^3 y_{i+1} - (s-1)^3 y_i + \alpha_y s^2(s-1) + \beta_y s(s-1)^2 \tag{3.16b}$$
$$x(s) = s^3 z_{i+1} - (s-1)^3 z_i + \alpha_z s^2(s-1) + \beta_z s(s-1)^2 \tag{3.16c}$$

With,

$$\begin{aligned}
&\alpha_x = TV_{i+1}\cos(\gamma_{i+1})\cos(\chi_{i+1}) - 3x_{i+1}; && \beta_x = TV_{i+1}\cos(\gamma_{i+1})\cos(\chi_{i+1}) + 3x_{i+1};\\
&\alpha_y = TV_{i+1}\cos(\gamma_{i+1})\sin(\chi_{i+1}) - 3y_{i+1}; && \beta_y = TV_{i+1}\cos(\gamma_{i+1})\sin(\chi_{i+1}) + 3y_{i+1};\\
&\alpha_z = TV_{i+1}\sin(\gamma_{i+1}) - 3z_{i+1}; && \beta_z = TV_{i+1}\sin(\gamma_{i+1}) + 3z_{i+1};
\end{aligned} \tag{3.17}$$

For more details, please refer to [17, 49, 148].

3.2.2 Motion Planning

Flight planning involves creating a plan to guide an aerial robot from its initial position to a desired destination way point [17]. Historically, two fields have contributed to trajectory or motion planning methods: robotics and dynamics & control. The first topic has a stronger focus on computational issues and real-time robot control, while the second one emphasizes the dynamic behavior and more specific aspects of trajectory performance [69].

The aim of motion planning algorithms is to define feasible trajectories to reach a desired location. These trajectories must ensure the safety of the flight by taking into account obstacles, collision avoidance and dangerous regions. A dangerous region is a space with a critical flight condition like unusual low or high pressure, hazardous wind storms, or any other factor affecting the flight. There are a variety of motion planning problem types that can be found in the literature. Such classification is made according to:

- The knowledge of the environment: it is said to be *static* for a perfect knowledge and *dynamic* in the opposite case.
- The type of obstacles: it is called *time-invariant* if the obstacles are fixed and *time-variant* when the obstacles are allowed to move.
- The presence of constraints on the vehicle: *differentially constrained* (or *kinodynamic*) when the vehicle's equation of motion act as constraints on the path. While for *differentially unconstrained* problems, the vehicle may use infinite accelerations to acheve a path.

Motion planning under differential constraints can be considered as a variant of classical ***T**wo-**P**oint **B**oundary **V**alue **P**roblem* (**TPBVP**) [76]. For a classical TPBVP, the initial and final points are given, and the objective is to compute a trajectory through the state space that connects initial and final states and which satisfies the differential constraints. Motion planning involves the additional complication of avoiding obstacles in the state space. Unfortunately, the techniques used to solve TPBVPs are not suitable for motion planning because they are not designed to handle obstacles.

In next subsections, an overview on most successful *differentially-constrained* planning algorithms in literature is presented.

3.2.2.1 Cell decomposition

The main idea of these approaches is to decompose the configuration space $\mathcal{C}$ into a number of disjoint sets, called cells. The resulting structure of $\mathcal{C}$ called connectivity graph G is considered as an important element of cell decomposition methods. Each cell is represented as a node in this graph; and two nodes are connected by an edge if and only if the two corresponding cells are adjacent.

Cell decomposition algorithms can be classified as *exact* or *approximate*. The major difference between the previous two classes is the methods used to build the structure of $\mathcal{C}$. For exact methods, the union of all cells equal to the closure of the collision-free configuration space. Whereas in approximate approaches, the structure of $\mathcal{C}$ is obtained with cells which have a simple predefined shape.

3.2.2.2 Probabilistic roadmap approach

For high-dimensional planning problems, the exact algorithms are computationally too expansive to calculate an explicit representation of the configuration space. Therefore, probabilistic motion planning approaches have achieved a substantial attention throughout the last decades because of their capability to solve high-dimensional problems in acceptable execution times.

Probabilistic roadmap planers [3, 42, 94] work in two phases. The first one is a *learning phase* while the second one is a *query phase*. In the learning phase, a graph, called a roadmap, is built by randomly sampling the configuration space. The free-collision configurations form the vertices of the roadmap. Then, the neighboring vertices are connected by edges if their corresponding configurations can be connected by continuous path through a local planner. The most common local planner simply checks a straight line connecting the two configurations.

On the other side, during the query phase, the algorithm tries to connect the initial and final configurations to the roadmap. Then, it search for a path connecting these two states.

It is important to note that if the environment is static, then the roadmap can be reused for further queries. Therefore, the algorithms used for dynamic environment are classified as multiple query methods. In [25] a single query variant called Lazy PRM. In this approach, the roadmap is built in the whole configuration space instead of the collusion-free configuration space. Once, a path has been found in the query phase, a collision checking algorithm is used to check the feasibility of the path.

However, the performance of Probabilistic roadmap planners can degrade if the solution has to pass through a narrow passage. To cope with a such problem, several variations and extensions have been proposed in [80, 122]. Another difficulty is the fast growing number of nodes and samples in the roadmap. Solutions for a such problem can be found in [1, 27, 79, 104, 151].

3.2.2.3 Rapidly-expanding random Tree (RRT)

Another probabilistic planing approach is **R**apidly-**E**xpanding **R**andom **T**ree (RRT). The basic idea of this method is building a tree which starts from the initial configuration and explore the collision-free configuration space. The exploration is done through a random sampling of the state space and expanding the search tree step-by-step. The objective is to build a graph structure covering the configuration space with nodes, explored positions and edges describing control inputs needed to move from node to node [17, 99].

In each step, a random state (x_{rand}) is chosen. Next, the nearest vertex (x_{near}) in the tree (with respect to a chosen metric) is selected. Then, a new edge going from x_{near} and pointing to x_{rand} as well as describing the control effort needed to reach is added. Finally, between all potential new configurations, the closest state to the desired final configuration (x_f) is designated as a new state (x_{new}) and added to the tree as a new vertex. This process is repeated until reaching the final configurations (x_f).

The RRT was described for the first time in [111, 112]. In these two papers, the vehicle is considered holonomic; neither dynamics nor kinematic constraints are considered. Another variation of RRT algorithm is described in [113]. The proposed algorithm is based on two trees. The first one is grown from the initial configuration while the second one from the desired final state. In every step, both trees try to connect to the same state in configuration space. Thus, a path is often found very quickly at the cost of diverging

a lot from an optimal path. The major advantage of RRT-like methods is that they are perfectly suitable for nonholonomic and differentially constrained planning problem [114].

3.2.2.4 A* Algorithm

The A* algorithm belongs to graph search algorithms. This kind of approaches were developed for computer science to find the shortest path between two nodes of connected graphs. They were designed for computer networks to develop routing protocols and were applied to path planning through decomposition of the path in waypoint sequences [44].

The A* algorithm is considered as one of the most important solvers explicitly oriented to motion-robotics [74]. It searches a graph efficiently with respect to a chosen heuristic and cost function; and returns an optimal path if the heuristic is optimistic. An optimistic or admissible heuristic always returns a value less than or equal to the cost of the shortest path from the current node to the goal node within the graph. The cost function used in A* algorithm is the sum of the following two terms:

- Cost to go H : defined as a heuristic estimation of the distance from the neighbouring cell x_{near} to the final one x_f.
- Cost to come G : Known as the distance between the expanded cell x and the neighbouring one x_{near}.

The G-value is 0 for the starting configuration and it increases while the algorithm expands successive states. On the other side, the H-value is used to drive the states expansion toward the desired final configuration, reducing this way the amount of expanded cells and improving the convergence.

The input of A* algorithm is the graph itself. So, the nodes can be naturally embedded into the aerial vehicle free configuration space and have a specified coordinates. The edges correspond to adjacent nodes and have the value of the cost appropriate to the travel. The output of A* algorithm is a back pointer path, in other words a sequence of nodes starting from the goal configuration and coming back to the start states. More details on A* algorithm can be found in [17, 34, 44, 49].

3.2.2.5 Potential field planners

In these approaches, an artificial potential field is used in order to steer the aerial vehicle from its initial configuration to a desired final one. The robot moves under the influence of a potential field which is obtained as a superposition of an attractive potential to the goal configuration and a repulsive potential from the obstacle regions. In other

words, the key idea relies on the assignment of a potential function to the free space and simulating the vehicle as a particle reacting to the forces resulted from the potential field [17].

The planning arise in a step-by-step manner. At each configuration, the most promising direction of local motion is indicated by a negative gradient. Since the value of the potential function can be regarded as an energy, it follows that the gradient of the potential is a force. This force points in the direction that locally maximally increases the potential. The combination of repulsive and attractive forces should direct the robot from the start state to the final one and guarantee collision-free aspect of the path. When the vehicle reaches its final destination the gradient vanishes. The main weakness of potential field methods is the local minima. In fact, the presence of a local minima may trap the vehicle in some configuration which differ from the desired one. A number of different potential fields have been proposed to reduce the number of local minima and the size of their region of attraction, please see [110].

In general there are two classes of potential fields known to satisfy properties of a navigation function: those based on a harmonic function and those based on solving optimal distance-to-go. More details in literature on this methods can be found in [17, 37, 69, 72, 97, 141].

3.2.2.6 Maneuver automaton

The concept of **M**aneuver **A**utomaton (MA) for aerial vehicles is based on the observation that the human pilots achieve agile control using a combination of trim trajectories and maneuvers. By definition, a maneuver is a finite time non-equilibrium transition between two trims [61]. In addition, MA approach helps to reduce the computational complexity of the trajectory optimization because it transcribes the dynamics of the aerial vehicle into a finite dimensional space. The main feature of a maneuver model is the selection of appropriate motion primitives.

An exact description of the concept of maneuver automaton within the context of autonomous guidance is given in [60]. The automaton set of trim and maneuvers are used to pre-compute a cost-to-go map. Next, this map is used online with a greedy guidance policy, while the states that are located between the pre-computed values are obtained via interpolation. In [145] a maneuver automaton was integrated in the receding horizon optimization framework. The fixed trim trajectories was replaced by controllable linear modes, while the maneuvers are open-loop trajectories. This model is more faithful to human control strategy. In [48] the authors extends the previous framework to allow interpolation between maneuver boundary conditions within a class of maneuvers.

Thus, an additional flexibility for the initiation and completion of maneuvers is provided.

3.2.2.7 From operational research problems to motion planning

In most of UAV's applications, the aerial vehicle required to visit a collections of predefined regions taking into account the vehicle restrictions like consumed fuel, flight time, etc... Such necessity resulted in the appearance of a novel class of optimal motion planning problems [144]. A region is said to be visited if the vehicle passes through it, and the objective of the planner is to find the shortest path connecting these regions.

Such problems is similar to classical Traveling Salesman Problem (TSP) and Vehicle Routing Problem (VRP) and their variants. Thus, the tools developed for solving TSP and VRP can be adopted for aerial vehicles planning problems.

3.2.2.8 Traveling Salesman Problem

The traveling salesman problem (TSP) is a well known combinatorial optimization problem. The statement of the TSP is simple and yet hundreds of papers have been devoted to this problem due to its multiple applications. The TSP consists on finding the optimal route that, given a map of cities, the salesman has to visit all the cities only once to complete a tour such that the length of the tour is the shortest among all possible tours for that map, see Figure (3.1)

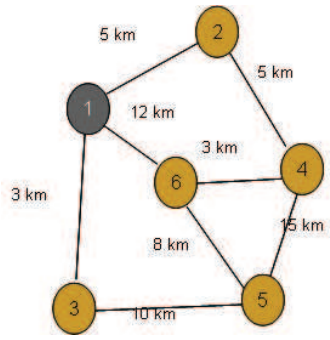

FIGURE 3.1: Traveling salesman problem. Initial map.

Several surveys have been written including [16, 107]. The author in [107], classifies the TSP algorithms into exact methods and approximate methods.

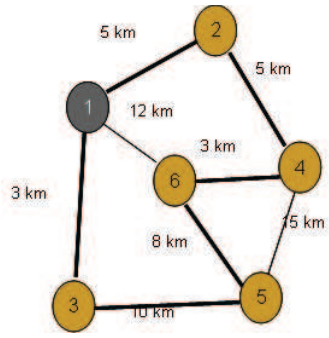

FIGURE 3.2: Traveling salesman problem. Optimal route.

Exact Methods

The exact algorithms are explained in the context of integer linear programming including methods such as Branch-and-bound, the shortest spanning arborescence bound, the shortest spanning tree bound, the 2-matching lower bound, etc. In general the formulation of the TSP considers a minimization problem subject to constraints such as degree constraints, sub-tour elimination constraints, and connectivity constraints.

Heuristic Methods

The TSP is a NP-hard problem, then heuristic methods to solve this problem is a natural option. The research on TSP using heuristic methods has two branches, worst case performance and good empirical performance. Guaranteed worst case performance methods consider a symetrical TSP on a graph G. A simple solution is computed using the length of a shortest spanning tree T on G. An improvement on this method is based on the fact that the shortest spanning is in general not Eulerian. However, an Eulerian graph can be derived by adding some links by means of a minimum cost matching algorithm, see [35]. The good empirical performance methods include tour construction methods using either nearest-neighbor algorithms or insertion algorithms, tour improvement methods using k-opt algorithms, simulated annealing and composite algorithms.

Classical algorithms are known to result in exponential computational complexities. Methods based on population optimization overcome this problems. Evolutionary Computation [53], Neural Networks [75], Ant Systems [50], Particle Swarm Optimization [52], Simulated Annealing [100] are among the problem solving techniques inspired from observing nature.

3.2.2.9 Vehicle Routing Problem

The Vehicle Routing Problem (VRP) can be described as the problem of designing optimal delivery or collection routes from one or several depots to a number of geographically scattered cities or customers, subject to side constraints, see Figure (3.3). It arises naturally as a central problem in the fields of transportation, distribution and logistic [43]. In addition, the VRP and its many extensions have been studied extensively in the literature, so they easily lend themselves to unmanned aerial vehicles task management problems.

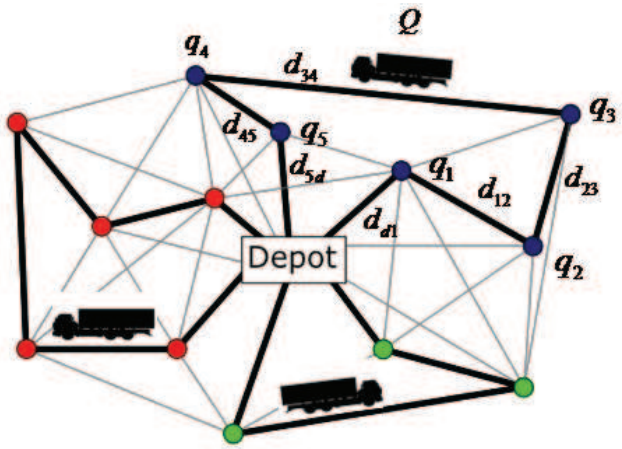

FIGURE 3.3: Classification of the vehicle routing problem with capacity constraints.

In [166], a vehicle routing problem algorithm was applied to an intelligence, surveillance, and reconnaissance (ISR) scenario for multiple UAVs. Task allocation optimization problem for a fleet of UAVs with coupled tasks and relative timing constraints is studied in [2]. The proposed methods minimize the mission completion time for the fleet, and the different UAV capabilities and no-fly zones were taken into account. Whereas, complex quantitative timing constraints are incorporated into vehicle routing problem in [93]. In that paper, the authors employ Metric Temporal Logic (MTL) as a natural language to specify complicated tasks, and provide an algorithm based on Mixed-Integer Linear Programming to solve the problem with an application to multi-UAV mission planning. Recently, the authors in [157] present Fuel Constrained UAV Routing Problem (FCURP) for a given set of targets and depots, and one unmanned aerial vehicle. Their objective is to find a path for the UAV such that each target is visited at least once by the vehicle, the fuel constraint in never violated along the path, and the total fuel required by the

aerial vehicle is minimum. Other research on UAV routing and mission planning problem can be found in [15, 57, 140, 146, 147].

Problem Formulation

Let $G = (V, A)$ be a graph where $V = \{V_1, V_2, ..., V_n\}$ is a set of vertices representing locations to be visited by the UAV with its ground base at V_1, and A is the set of arcs. Suppose that $C : A \to \mathbb{R}^+$ denote the cost function with $C(V_i, V_j)$ representing the cost of traveling from vertex V_i to vertex V_j. Observe that, $C(V_i, V_j)$ can be interpreted as consumed fuel, path length, travel time, etc...

In addition, assume that there are m vehicles available at the ground base, and all vehicles are identical and have the same capacity. Then, the vehicle routing problem consists of designing a set of beast-cost vehicle routes in such way that:

- Each city in $\{V_2, ..., V_n\}$ is visited exactly once by exactly one vehicle.
- All vehicle routes start and end at the ground base.
- Some side constraints are satisfied.

Variants of Vehicle Routing Problem

According to [161], the variants of the vehicle routing problem can be classified as shown in Figure (3.4)

- Capacitated Vehicle Routing Problem (CVRP): CVRP is a vehicle routing problem in which the vehicles have limited carrying capacity of the goods that must be delivered.
- Distance and Capacitated Vehicle Routing Problem (DCVRP): DCVRP is similar to CVRP with an additional restriction on the allowed maximum distance traveled in each tour.
- Vehicle Routing Problem with Time Windows (VRPTW):The additional restriction in VRPTW is a time window associated with each customer, defining an interval wherein the customer has to be supplied.
- Vehicle Routing Problem with Backhauling (VRPB): It is an extension of VRP that includes both a set of customers to whom products are to be delivered, and a set of vendors whose goods need to be transported back to the distribution center. In addition, on each route all deliveries have to be made before any goods can be picked up to avoid rearranging the loads on the vehicle.
- Vehicle Routing Problem with Backhauling and Time Windows (VRPBTW): It is a combination between VRPB and VRPTW.
- Vehicle Routing Problem with Pickup and Delivery (VRPPD): It is a VRP in which the customers have the possibility to return some commodities. In addition, VRPPD must take into account that the goods can be fitted into the vehicle. This restriction

make the planning problem more difficult and it can lead to bad utilization of the vehicles capacities, increase traveled distance or rise the number of needed vehicles.

- Vehicle Routing Problem with Pickup and Delivery and Time Windows (VRPPDTW): It is a hybrid between VRPPD and VRPTW.

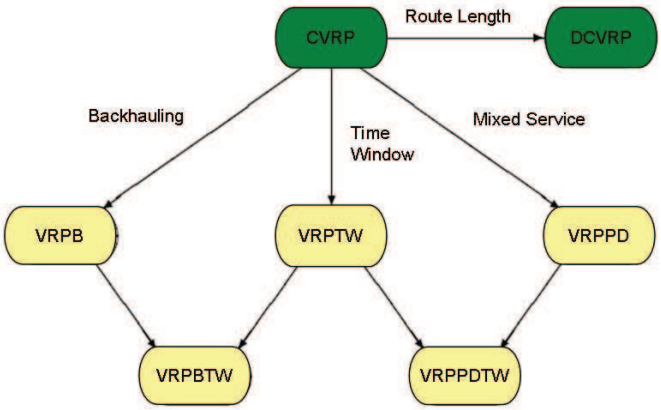

FIGURE 3.4: Vehicle routing problem with capacity constraints.

VRP approaches consist of heuristic and exact algorithms. The heuristic methods are characterized by finding solutions that are close to optimum quickly. But they have no worst case guarantee. On the other side, exact algorithms provide solutions for instance of Practical sizes [93].

Exact Methods

The exact algorithms for the VRP can be classified into three broad categories:

1. Direct tree search methods.
2. Dynamic programming.
3. Integer linear programming.

As the number of proposed algorithms is very large, we refer the reader to the survey [108].

Heuristic Methods

Heuristic algorithms for the VRP can often be derived from procedures derived from the TSP. The nearest neighbour algorithm, insertion algorithms and tour improvement procedures can be applied to CVRPs and DVRPs almost without modifications. However, when applying these methods to VRPs care must be taken to ensure that only feasible vehicle routes are created. VRP heuristic methods can also be subcategorized as:

1. One step [36], [67].
2. Two steps [68], [58].
3. Improvement methods [169], [68].

For more details, please see [39, 103].

3.3 Bridge Inspection

In this section we present a trajectory generation approach and a motion planning strategy for an unmanned aerial vehicle dedicated to bridge inspection tasks. First of all, a controllability analysis is carried out. Then, we deal with the problem of trajectory generation in the presence of wind and obstacles. Finally, we develop a motion planning strategy to respect the maximum flight time allowed for the aerial vehicle.

3.3.1 Accessibility and Controllability

3.3.1.1 Affine nonlinear system

An affine nonlinear system with drift can be defined as

$$\dot{X} = f(X) + \sum_{i=1}^{m} g_i(X) u_i \tag{3.18}$$

where $X \in M \subset \mathcal{R}^n$ with M a smooth manifold, $f(X)$, $g_i(X)$ are smooth vector fields on M, and $U \in \mathcal{R}^m$. $f(X)$ is called the drift vector field, and $g_i(X)$; $i \in m$, the input vector fields. This class of nonlinear systems was the subject of numerous researches [23, 88, 106, 117, 133, 158].

Notice that the Equation (2.70) and Equation (2.100) can be rewritten as an affine nonlinear system, thus it yields

$$\dot{X} = f(X) + g_1 u_1 + g_2 u_2 + g_3 u_3 \tag{3.19}$$

where

$$f(X) = \begin{bmatrix} V\cos\chi\cos\gamma + W_x \\ V\sin\chi\cos\gamma + W_y \\ V\sin\gamma + W_z \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \begin{aligned} g_1 &= [0,0,0,1,0,0]^T \\ g_2 &= [0,0,0,0,1,0]^T \\ g_3 &= [0,0,0,0,0,1]^T \end{aligned}$$

3.3.1.2 Controllability

The system (3.18) is said to be ***controllable*** if for any two points x_0 and x_f in M there exists an admissible control $u(t)$ defined on some time interval $[0, T]$ such that the system (3.18) with initial condition x_0 reaches the point x_f in time T [23].

In general for nonlinear systems controllability is a notion that is too difficult to verify, however a more restricted (local) version of controllability called accessibility can be employed. To introduce this notion it is necessary to define what it means by a reachable set for a nonlinear system.

3.3.1.3 Reachable set

Let ψ be a neighborhood of the point $X \in M$ and $R^{\psi}(x_0, t)$ indicates the set of reachable states from x_0 at time t by trajectories remaining inside ψ and satisfying (3.18). Then, the reachable set from x_0 at time T is defined as

$$R^M(x_0, T) = \bigcup_{0 \leq t \leq T} R^M(x_0, t) \tag{3.20}$$

3.3.1.4 Accessibility

The following four notions of accessibility can be distinguished

- System (3.18) is ***accessible*** from $x_0 \in M$ if for every $T > 0$, $R^M(x_0, T)$ contains a nonempty open set.
- System (3.18) is ***locally accessible*** from $x_0 \in M$ if for every $T > 0$, $R^{\psi}(x_0, T)$ contains a nonempty open set.
- System (3.18) is said to be ***strongly accessible*** from $x_0 \in M$ if the set $R^M(x_0, T)$ contains a nonempty open set for any $T > 0$ sufficiently small.
- System (3.18) is said to be ***locally strongly accessible*** from $x_0 \in M$ if for any neighborhood ψ of x_0 the set $R^{\psi}(x_0, T)$ contains a nonempty open set for any $T > 0$ sufficiently small.

On the other hand, the following definitions are fundamental tools in proving accessibility property.

Definition 3.1. Given two vector fields $f(x), g(x)$, their ***Lie bracket*** denoted by $[f(x), g(x)]$ is given as following

$$[f(x), g(x)] = \frac{\partial g(x)}{\partial x} f(x) - \frac{\partial f(x)}{\partial x} g(x)$$

Definition 3.2. The ***accessibility algebra*** A of the system (3.18) is the smallest Lie algebra of vector fields on M that contains the vector fields f and $g_1, ..., g_m$.

Definition 3.3. The ***accessibility distribution*** $\triangle_A$ of (3.18) is the distribution generated by the vector fields in A; i.e. $A(x)$ is the span of vector fields v in A at x. Thus, $\triangle_A$ can be determined as

$$\triangle_A = \text{span}\{v \mid v \in A\} \tag{3.21}$$

In other words, $\triangle_A$ is the involutive closure of $\triangle = \text{span}\{f, g_1, ..., g_m\}$. The computation of $\triangle_A$ may be organized as an iterative procedure

$$\triangle_A = \text{span}\{v \mid v \in A_i, \forall i \geq 1\}$$

with

$$\begin{aligned} \triangle_1 &= \triangle = \text{span}\{f, g_1, ..., g_m\} \\ \triangle_i &= \triangle_{i-1} + \text{span}\{[g, v] \mid g \in \triangle_1, v \in \triangle_{i-1}\}; i \geq 2 \end{aligned} \tag{3.22}$$

This procedure stops after K steps, where K is the smallest integer such that $\triangle_{K+1} = \triangle_K > \triangle_A$.

Remark: The accessibility distribution $\triangle_A$ of (3.19) can be investigated as follows

$$\triangle_A = \triangle_3 = \text{span} \left\{ \begin{array}{c} f,\ g_1,\ g_2,\ g_3, \\ g_4 = [f, g_1],\quad g_5 = [f, g_2], \\ g_6 = [f, g_3],\quad g_7 = [g_2, [f, g_2]] \end{array} \right\}$$

Definition 3.4. Let A to be the accessibility algebra of (3.18). The ***strong accessibility algebra*** A_0 is defined to be the smallest sub-algebra containing g_1, g_2, ..., g_m and satisfies $[f, v_0] \in A_0$ for all $v_0 \in A_0$.

Definition 3.5. The ***strong accessibility distribution*** $\triangle_{A_0}$ of (3.18) is defined to be the involutive distribution generated by the vector fields in A_0.

Remark: The strong accessibility distribution $\triangle_{A_0}$ of (3.19) from Definition 3.5 is given by

$$\triangle_{A_0} = \text{span} \left\{ \begin{array}{c} g_1, g_2, g_3, \\ g_4 = [f, g_1], g_5 = [f, g_2], \\ g_6 = [f, g_3], g_7 = [g_2, [f, g_2]] \end{array} \right\} \tag{3.23}$$

On the other hand to prove the local accessibility property, the following theorem is used [23].

Theorem 3.6. *Consider the system (3.18) and assume that the vector fields are C^∞. If*

$$dim \triangle_A(x_0) = n \tag{3.24}$$

i. e., the accessibility algebra spans the tangent space to M at x_0, then for any $T > 0$, the set $R^\psi(x_0, T)$ has a nonempty interior; i.e. the system has the accessibility property from x_0 for all neighborhoods ψ of x_0 and all $T > 0$.

Remark: Applying the previous Theorem to (3.19) we obtain

$$-V^2 \cos\gamma \left(\begin{array}{c} V + W_z \sin\gamma + W_y \sin\chi \cos\gamma + \\ + W_x \cos\chi \cos\gamma \end{array} \right) \neq 0 \tag{3.25}$$

Hence, the system (3.19) verifies the Lie Algebra rank condition and is locally accessible for $V \neq 0$, or $\gamma \neq \frac{\pi}{2}$, or $(V + W_z \sin\gamma + W_y \sin\chi \cos\gamma + W_x \cos\chi \cos\gamma) \neq 0$.
The last condition can be rewritten as

$$V \neq -[W_x \cos\chi \cos\gamma + W_y \sin\chi \cos\gamma + W_z \sin\gamma] \tag{3.26}$$

The left hand of the previous equation presents the projection of the wind velocity vector on $x_{\mathcal{W}}$. It follows that $V \neq -W_x|_{\in R_{\mathcal{W}}}$

Similarly to prove the local strong accessibility property, the following theorem is used [133].

Theorem 3.7. *Consider the system (3.18). Suppose that*

$$dim \triangle_{A_0} = n, \tag{3.27}$$

then the system is locally strongly accessible from x_0.

Remark: Using Theorem 3.7 the following condition is obtained

$$V^2 \cos\gamma \neq 0 \tag{3.28}$$

Therefore the system (3.19) is locally strongly accessible if $V \neq 0$ or $\gamma \neq \frac{\pi}{2}$.

Notice that

1. Condition (3.24) is known as: ***Lie Algebra Rank Condition*** (LARC).
2. If the system (3.18) is driftless, i.e., $f(X) = 0$, the accessibility property characterizes the controllability.

From the previous study, we conclude that the kinematic equations of the system can be used to stabilize the aerial vehicle (airship/quad-rotor) around a trajectory. Observe that, these equations do not provide us the (strong) accessibility property to stabilize the vehicle in hovering flight mode ($V = 0$).

3.3.1.5 Controllability of kinematic aerial vehicle in the presence of constant wind

In this section, we investigate another approach to find the necessary and sufficient condition for the system (3.19) to be completely controllable. This method is a generalization of the controllability analysis presented in [10] for a Dubins vehicle moving in a plane in the presence of constant wind.

Assuming that the aircraft starts from an initial configuration $q_0 = [x_0,\, y_0,\, z_0,\, \gamma_0,\, \chi_0,\, V_0]$ to a final one $q_f = [x_f,\, y_f,\, z_f,\, \gamma_f,\, \chi_f,\, V_f]$ as shown in Figure (3.5). In addition, the wind velocity (V_w) is supposed to be a constant with a specified direction γ_w and χ_w. Then, the guidance problem from q_0 to q_f in the presence of wind can be regarded as a moving-target problem [10, 96]. For this purpose, it is more convenient to rewrite Equation (3.19) as shown below

$$\dot{x}_n = V\cos\chi\cos\gamma \tag{3.29a}$$
$$\dot{y}_n = V\sin\chi\cos\gamma \tag{3.29b}$$
$$\dot{z}_n = V\sin\gamma \tag{3.29c}$$
$$\dot{\gamma} = u_1 \tag{3.29d}$$
$$\dot{\chi} = u_2 \tag{3.29e}$$
$$\dot{V} = u_3 \tag{3.29f}$$

where, $x_n \triangleq x - w_x t$, $y_n \triangleq y - w_y t$, and $z_n \triangleq z - w_z t$.

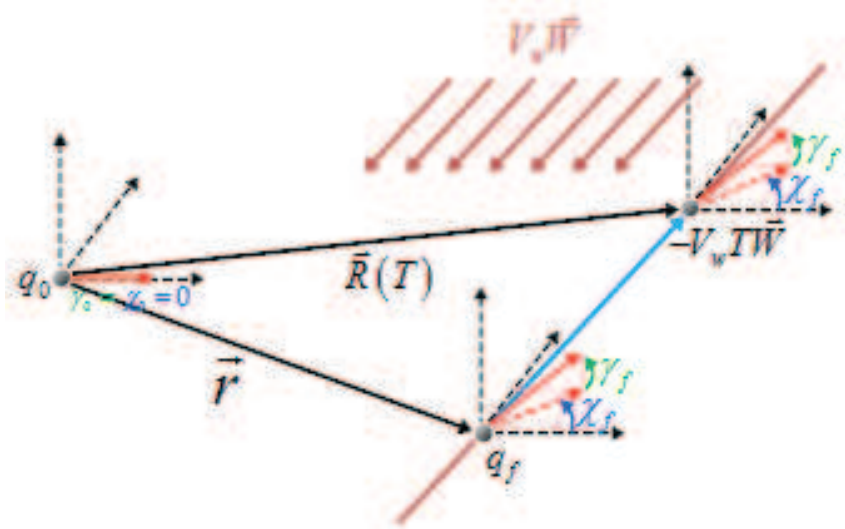

FIGURE 3.5: Moving target problem.

Thus, the aerial vehicle in the presence of a constant wind, i.e. Equation (3.19), can reach the desired final configuration q_f at a time $T > 0$ if and only if the vehicle in the absence of wind, i.e. Equation (3.29), intercepts at T a particle that moves along the half-line $\epsilon = \{\overrightarrow{p} \in \mathcal{R}^3 : \overrightarrow{p} = \overrightarrow{r} - V_w\overrightarrow{W}, t \geq 0\}$ with a constant wind speed V_w.

Denoting the position vectors of the vehicle, described in Equation (3.29), and the particle by $\overrightarrow{R}(t)$ and $\overrightarrow{P}(t)$ respectively. It follows that the existence of a time $T > 0$ such that

$$\overrightarrow{R}(T) = \overrightarrow{r} - V_wT\overrightarrow{W} \tag{3.30}$$

where, $\overrightarrow{R}(t) = \overrightarrow{q_0q_i}$ and $\overrightarrow{r} = \overrightarrow{q_0q_f}$ as shown in Figure (3.5); characterize the controllability of the aerial vehicle presented in Equation (3.19).

Obviously, if $V_w < \|\overrightarrow{V}\|_{max}$, the aircraft travels faster than the particle. Moreover, since the vehicle is locally accessible (under the conditions : $V \neq 0$ and $\gamma \neq \frac{\pi}{2}$), there is some time $t_1 > 0$ such that the vehicle intersects ϵ with a flight path angle of $-\gamma_w$ and a heading angle of $-\chi_w$. So, both the vehicle and the particle move along ϵ. At $t = t_2 > t_1$, the vehicle will precede the particle and reach the point $\overrightarrow{R}(t)$ sufficiently ahead by a distance d. This distance is obtained by $d = V_wT_d$, where T_d is the minimum time required to steer the vehicle from $\overrightarrow{R}(t_2) = \overrightarrow{q_0q_i}$ to the same position with flight path angle of γ_f and a heading angle of χ_f. Then, the aerial vehicle intercepts the particle at $\overrightarrow{P}(T) = \overrightarrow{R}(T)$ with $T \triangleq t_2 + T_d$. The previous discussion is illustrated in Figure (3.6).

On the other side, if $V_w \geq \|\overrightarrow{V}\|_{max}$, the particle travels with a speed greater than or equal to the vehicle's speed. So, there would be a reachable set for which an interception can take place[10].

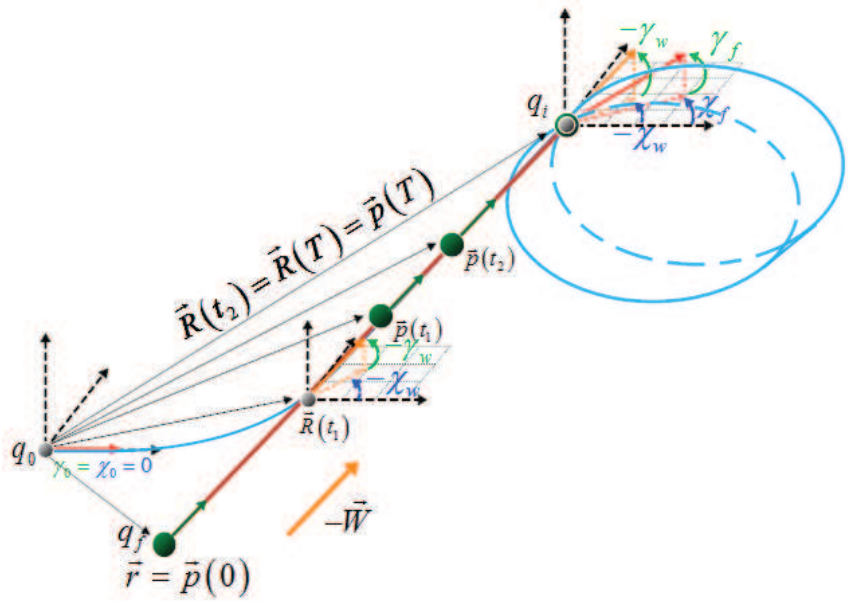

FIGURE 3.6: Sufficient condition for controllability.

The vehicle, which is assumed to be initially in q_0, can move only in the direction of its inertial velocity $\overrightarrow{V}_I = \overrightarrow{V} + \overrightarrow{W}$ with $\overrightarrow{V} = [V_0 \cos\chi_0 \cos\gamma_0,\ V_0 \sin\chi_0 \cos\gamma_0,\ V_0 \sin\gamma_0]$. After some small time δt (sufficiently small to reach V_{max}) the aerial vehicle reaches the configuration q_0' with a flight path angle of $\gamma = \gamma_0 + u_1\delta t$ and a heading angle of $\chi = \chi_0 + u_2\delta t$ where $u_i \in \{-U_{imax}, U_{imax}\}; i = 1, 2$. From the Figure (3.7), it follows that the aircraft is constrained to move along a direction that lies within the cone with vertex q_0' and angle $\widehat{\overrightarrow{V}_I^+ q_0' \overrightarrow{V}_I^-}$, where $\overrightarrow{V}_I^{\pm} = \overrightarrow{V}^{\pm} + \overrightarrow{W}$ with $\overrightarrow{V}^{\pm} = [V_{max} \cos\chi^{\pm} \cos\gamma^{\pm},\ V_{max} \sin\chi^{\pm} \cos\gamma^{\pm},\ V_{max} \sin\gamma^{\pm}]$; $\gamma^{\pm} = \gamma_0 \pm U_{1max}\delta t$ and $\chi^{\pm} = \chi_0 \pm U_{2max}\delta t$.

3.3.2 Trajectory Generation

The aim of this section is to compute an optimal trajectory under acceleration, heading and flight path rates limits together with the associated control demand. The obtained trajectories are analyzed from the point of view of the solution structure e.g. type of control demand number and duration of active constraints.

Since the presence of an obstacle in the air flow generates meteorological turbulence such as wind shear, Venturi effect, Karman vortex, etc... It is important to include such elements in the trajectory generation phase. This procedure can contribute to saving time and energy if the wind come in the direction of motion for example. This problem is a type of Zermelo's problem introduced in [174]. It can be formulated as follows (like its described in [29]):

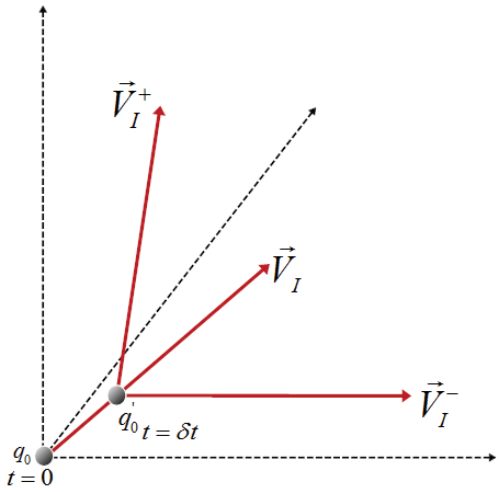

FIGURE 3.7: Reachability analysis when $V_w \geq V_{max}$

A ship must travel through a region of strong currents. The magnitude and direction of the currents are known as functions of position:

$$u = u(x, y) \quad \text{and} \quad v = v(x, y) \tag{3.31}$$

where, (x, y) are rectangular coordinates, and (u, v) are the velocity components of the current in the x and y directions respectively. The magnitude of the ship's velocity relative to the water is V, a constant. The problem is to steer the ship in such a way as to minimize the time necessary to go from a point A to a point B.

This problem can be extended to an aerial vehicle flying at a constant altitude. A large number of publications dealing with Zermelo's problem ant its extensions can be found in the literature, e.g. [22, 70, 83, 84, 88, 159].

The originality of the work is twofold: Firstly, planning in three dimension with varying velocity, heading angle and path angle, i.e. guarantee the continuity of the velocity time profile along all the trajectory. Secondly, taking into account the wind effect.

3.3.2.1 Optimal Control

In this section an optimal control problem is considered in which the objective is to steer the autonomous aircraft in a wind field from one point to another in minimum time. By using the realistic assumption that the system is completely controllable in a weak wind, at least one feasible path exists for every pair of initial and final states.

Thus, an admissible control trajectory $u : [0, T] \mapsto \Omega \subseteq \Re^m$ must be found such that the aerial vehicle kinematic system described by the system of nonlinear differential equations given in Equation (2.70) and Equation (2.100), which can be expressed as

$$\dot{X}(t) = f\left(X(t), U(t), t\right); \quad \begin{matrix} X(t) \in \mathbb{R}^6 \\ U(t) \in \mathbb{R}^3 \end{matrix} \tag{3.32}$$

is transferred from the initial configuration

$$x(0) = x_0, y(0) = y_0, z(0) = z_0, \chi(0) = \chi_0, \gamma(0) = \gamma_0, V(0) = V_0 \tag{3.33}$$

into an admissible final configuration

$$x(T) = x_f, y(T) = y_f, z(T) = z_f, \chi(T) = \chi_f, \gamma(T) = \gamma_f, V(T) = V_f \tag{3.34}$$

and such that the corresponding state trajectory satisfies the following limitations on the control inputs and states

$$|u_1| \leq u_{1max} \tag{3.35a}$$
$$|u_2| \leq u_{2max} \tag{3.35b}$$
$$|u_3| \leq u_{3max} \tag{3.35c}$$
$$|\gamma| \leq \gamma_{max} \tag{3.35d}$$
$$|V| \leq V_{max} \tag{3.35e}$$
$$(x - x_c)^2 + (y - y_c)^2 + z^2 \geq r_p^2 \tag{3.35f}$$

at all times $t \in [0, T]$ and such that the cost functional

$$min \int_0^T dt \tag{3.36}$$

is minimised.

The initial point can be required to be the origin of the i^{th} way point and the final point $(i + 1)^{th}$ way point or the final destination point.
Note that Equation (3.35d) and Equation (3.35e) represents the physical limitations on the vehicle while Equation (3.35f) describes a pillar of radius r_p and situated in (x_c, y_c) which is considered as an obstacle.
For points that are reachable, the resolution is based on the Pontryagin Minimum Principle which constitutes a generalization of Lagrange problem of the calculus of variations. The Pontryagin Minimum Principle to obtain optimal control for the nonlinear system

3.32, without taking state constraints, Equation (3.35d), Equation (3.35e) and Equation (3.35f), into account, can be summarized by the following different stages [19–21]

- Define the Hamiltonian function as

$$\mathcal{H}(x(t), u(t), \lambda(t), t) = L(x(t), u(t), t) + \lambda^T f(t, x(t), u(t)) \tag{3.37}$$

  where, $\lambda(t)$ is the co-state variable.
- Use the necessary condition of Weierstrass along $x^*(t)$. Minimize $\mathcal{H}$ with respect to $|u(t)| \leq u_{max}$ as

$$\mathcal{H}(x^*(t), u^*(t), \lambda^*(t), t) \leq \mathcal{H}(x^*(t), u(t), \lambda^*(t), t) \tag{3.38}$$

  where $*$ denote the optimal solution. This inequality must hold for any t for $0 \leq t \leq T$.
- Solve the following set of $2n$ state and co-state equations:

$$\dot{x}^* = \left(\frac{\partial \mathcal{H}}{\partial \lambda^*}\right) \qquad \text{and} \qquad \dot{\lambda}^* = -\left(\frac{\partial \mathcal{H}}{\partial x^*}\right) \tag{3.39}$$

  such that the boundary conditions, Equation(3.33) and Equation (3.34), be satisfied. This kind of problem is called Two Point Boundary Value Problem.
- Verify the ***transversality condition***: If the final time T is free or not a priori specified and the Hamiltonian does not depend explicitly on time t, then the Hamiltonian must be identically zero when evaluated along the optimal trajectory; that is

$$\mathcal{H}(x^*(t), u^*(t), \lambda^*(t)) = 0 \quad \forall t \in [0, T] \tag{3.40}$$

Let us now apply Pontryagin Minimum Principle to our system (3.19). The Hamiltonian is defined as:

$$\begin{aligned} \mathcal{H} = 1 + \lambda_1(V\cos\chi\cos\gamma + W_x) + \lambda_2(V\sin\chi\cos\gamma + W_y) \\ +\lambda_3(V\sin\gamma + W_z) + \lambda_4 u_1 + \lambda_5 u_2 + \lambda_6 u_3 \end{aligned} \tag{3.41}$$

The optimal control input must satisfy the following set of necessary conditions

$$\dot{X}^* = \frac{\partial \mathcal{H}}{\partial \lambda^*}; \qquad \dot{\lambda}^* = -\frac{\partial \mathcal{H}}{\partial X^*} \tag{3.42}$$

The co-state variables are free, i.e. unspecified, at both the initial and final times because the corresponding state variables of the system are specified. A first interesting result is the determination of a sufficient family of trajectories, i.e. a family of trajectories containing an optimal solution for linking any two configurations.
The co-state equations are then obtained in the standard fashion by differentiating the

negative of the Hamiltonian with respect to the states where:

$$\dot{\lambda_1}^* = 0 \quad \Rightarrow \quad \lambda_1^* = constant \tag{3.43a}$$

$$\dot{\lambda_2}^* = 0 \quad \Rightarrow \quad \lambda_2^* = constant \tag{3.43b}$$

$$\dot{\lambda_3}^* = 0 \quad \Rightarrow \quad \lambda_3^* = constant \tag{3.43c}$$

$$\dot{\lambda_4}^* = \lambda_1^* V^* \cos\chi^* \sin\gamma^* + \lambda_2^* V^* \sin\chi^* \sin\gamma^* - \lambda_3^* V^* \cos\gamma^* \tag{3.43d}$$

$$\begin{aligned} \dot{\lambda_5}^* &= \lambda_1^* V^* \sin\chi^* \cos\gamma^* - \lambda_2^* V^* \cos\chi^* \cos\gamma^* \\ &= \lambda_1^*(\dot{y}^* - W_y) - \lambda_2^*(\dot{x}^* - W_x) \end{aligned} \tag{3.43e}$$

$$\begin{aligned} \dot{\lambda_6}^* &= -\lambda_1^* \cos\chi^* \cos\gamma^* - \lambda_2^* \sin\chi^* \cos\gamma^* - \lambda_3^* \sin\gamma^* \\ &= -\lambda_1^* \frac{\dot{x}^* - W_x}{V^*} - \lambda_2^* \frac{\dot{y}^* - W_y}{V^*} - \lambda_3^* \frac{\dot{z}^* - W_z}{V^*} \end{aligned} \tag{3.43f}$$

Defining the Hamiltonian and multiplier dynamics in this way, the minimum principle of Pontryagin states that the control variable must be chosen to minimize the Hamiltonian at every instant [130, 173].

Therefore, applying Equation (3.38) Leads to the following solution:

$$u_i^* = \delta_i u_{imax}; \quad i = 1, 2, 3 \tag{3.44}$$

with δ_i denoting the *-sign* function defined as

$$\delta_i = \begin{cases} 1 & \text{for} \quad \lambda_{i+3}^* < 0 \\ 0 & \text{for} \quad \lambda_{i+3}^* = 0 \\ -1 & \text{for} \quad \lambda_{i+3}^* > 0 \end{cases} \tag{3.45}$$

This type of control called ″bang-bang″ is obtained from saturating inputs at all times, or zero for singular control.

In order to take the states constraints into account we use *penalty function* methods described in [130]. Let the inequality constraints on the states be expressed as

$$S\left(X(t), t\right) \geq 0 \tag{3.46}$$

where, S is a $p \leq n$ vector function of state and assumed to have continuous first and second partial derivatives with respect to state $x(t)$. Thus, Equation (3.35d), Equation

(3.35e) and Equation (3.35f) can be rewritten as follows

$$g_1(\gamma) = \gamma_{max}^2 - \gamma^2 \geq 0 \tag{3.47a}$$

$$g_2(V) = V_{max}^2 - V^2 \geq 0 \tag{3.47b}$$

$$g_3(x, y, z) = (x - x_c)^2 + (y - y_c)^2 + z^2 - r_p^2 \geq 0 \tag{3.47c}$$

The key idea is to convert the inequality constraints into equality constraints. Therefore, let us define a new state variable $x_{n+1}(t)$ by

$$\dot{\Upsilon}(t) = [g_1(\gamma)]^2 H(g_1) + [g_2(V)]^2 H(g_2) + [g_2(x, y, z)]^2 H(g_3) \tag{3.48}$$

where $H(g_i)$ is a unit Heaviside step function defined by

$$H(g_i) = \begin{cases} 0 & \text{for} \quad g_i \geq 0 \\ 1 & \text{for} \quad g_i < 0 \end{cases} \tag{3.49}$$

The Equation (3.48) and Equation (3.49) mean that $\dot{\Upsilon}(t) = 0$ for all t when the constraints described in Equation (3.35d), Equation (3.35e), and Equation (3.35f) are satisfied, whereas, $\dot{\Upsilon}(t) \geq 0$ for all t, due to the square terms in Equation (3.48), if the constraints are violated. Furthermore, let us require that the new variable $\Upsilon(t)$ has the following boundary conditions

$$\Upsilon(t_0) = 0, \quad \text{and} \quad \Upsilon(t_f) = 0 \tag{3.50}$$

Let us now define the new Hamiltonian as

$$\begin{aligned} \mathcal{H} = \quad & 1 + \lambda_1(V\cos\chi\cos\gamma + W_x) + \lambda_2(V\sin\chi\cos\gamma + W_y) + \lambda_3(V\sin\gamma + W_z) + \lambda_4 u_1 \\ & + \lambda_5 u_2 + \lambda_6 u_3 + \lambda_7 \left[[g_1(\gamma)]^2 H(g_1) + [g_2(V)]^2 H(g_2) + [g_2(x, y, z)]^2 H(g_3) \right] \end{aligned} \tag{3.51}$$

Then, the necessary optimality conditions for the state are

$$\begin{aligned} \dot{X}^*(t) &= \frac{\partial \mathcal{H}}{\partial \lambda} \\ \dot{\Upsilon}^*(t) &= \frac{\partial \mathcal{H}}{\partial \lambda_7} \end{aligned} \tag{3.52}$$

and for the co-state

$$\begin{aligned} \dot{\lambda}^*(t) &= -\frac{\partial \mathcal{H}}{\partial X} \\ \dot{\lambda_7}^*(t) &= -\frac{\partial \mathcal{H}}{\partial \Upsilon} = 0 \end{aligned} \tag{3.53}$$

Observe that $\dot{\lambda_7}^*(t) = 0$ because the Hamiltonian does not explicitly contain the new state variable $\Upsilon(t)$.

Applying Pontryagin Minimum Principle to augmented system leads to the following

$$\mathcal{H}(X^*(t), \Upsilon^*(t), u^*(t), \lambda^*(t), \lambda_7^*(t), t) \leq \mathcal{H}(X^*(t), \Upsilon^*(t), u(t), \lambda^*(t), \lambda_7^*(t), t) \tag{3.54}$$

Then, the optimal control is given by

$$u_i^* = -sign(\lambda_{i+3}) u_{imax}; \quad i = 1, 2, 3 \tag{3.55}$$

Obviously, the control variables depend on co-state variables. Thus, these variables are analyzed in the following section, to show their effect on controls.

Co-state Variables Analysis

The co-state variables $\lambda_{i+3}(t); i = 1, 2, 3$ are called "*switching functions*". In fact, when $\lambda_{i+3}(t)$ passes through zero, a switching time of the control $u_i^*(t)$ is indicated. If $\lambda_{i+3}(t)$ is zero for some finite time interval, then the minimal condition, i.e. Equation (3.38), provides no information about how to select $u_i^*(t)$, and the control is *singular*; see Figure (3.8)).

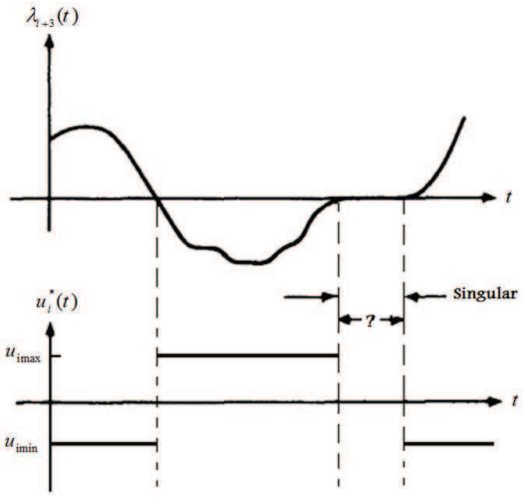

FIGURE 3.8: The relationship between a time-optimal control and its corresponding co-state variables.

From the co-states relation, Equation (3.43), it can be noticed that λ_1, λ_2 and λ_3 are time-invariant variables. In the other side λ_4, λ_5 and λ_6 are time-variant variables. So, let us calculate the time profiles of these multipliers in the time interval $t \in [t_k, t_{k+1}]$ for regular control inputs case. So, by integration of the equations (3.43) the following

relations are obtained:

$$\begin{aligned}\lambda_4 &= \left(\frac{\lambda_1 V}{2(\dot{\chi}+\dot{\gamma})} - \frac{\lambda_2 \dot{V}}{2(\dot{\chi}+\dot{\gamma})^2}\right)\sin(\chi+\gamma) + \left(\frac{\lambda_1 \dot{V}}{2(\dot{\chi}+\dot{\gamma})^2} + \frac{\lambda_2 V}{2(\dot{\chi}+\dot{\gamma})}\right)\cos(\chi+\gamma) + \\ &+ \left(\frac{\lambda_1 V}{2(\dot{\chi}-\dot{\gamma})} - \frac{\lambda_2 \dot{V}}{2(\dot{\chi}-\dot{\gamma})^2}\right)\sin(\chi-\gamma) + \left(\frac{\lambda_1 \dot{V}}{2(\dot{\chi}-\dot{\gamma})^2} + \frac{\lambda_2 V}{2(\dot{\chi}-\dot{\gamma})}\right)\cos(\chi-\gamma) - \\ &- \lambda_3\left(\frac{V\sin\chi}{\dot{\chi}} + \frac{\dot{v}\cos\chi}{\dot{\chi}^2}\right) + \lambda_4' \\ \lambda_5 &= \lambda_1(y-y_k) - \lambda_2(x-x_k) + (\lambda_2 w_x - \lambda_1 w_y)(t-t_k) + \lambda_{5k} \\ \lambda_6 &= \lambda_{6k} + \frac{-\lambda_1(\sin(\chi-\gamma)-\sin(\chi_k-\gamma_k)) + \lambda_2(\cos(\chi-\gamma)-\cos(\chi_k-\gamma_k))}{2(\dot{\chi}-\dot{\gamma})} + \\ &+ \frac{-\lambda_1(\sin(\chi+\gamma)-\sin(\chi_k+\gamma_k)) + \lambda_2(\cos(\chi+\gamma)-\cos(\chi_k+\gamma_k))}{2(\dot{\chi}+\dot{\gamma})} + \frac{\lambda_3(\cos\gamma-\cos\gamma_k)}{\dot{\gamma}}\end{aligned} \tag{3.56}$$

Where:

$$\begin{aligned}\lambda_4' &= \lambda_{4k} - \left(\frac{\lambda_1 V_k}{2(\dot{\chi}+\dot{\gamma})} - \frac{\lambda_2 \dot{V}}{2(\dot{\chi}+\dot{\gamma})^2}\right)\sin(\chi_k+\gamma_k) - \left(\frac{\lambda_1 \dot{V}}{2(\dot{\chi}+\dot{\gamma})^2} + \frac{\lambda_2 V_k}{2(\dot{\chi}+\dot{\gamma})}\right)\cos(\chi_k + \\ &+\gamma_k) - \left(\frac{\lambda_1 V_k}{2(\dot{\chi}-\dot{\gamma})} - \frac{\lambda_2 \dot{V}}{2(\dot{\chi}-\dot{\gamma})^2}\right)\sin(\chi_k-\gamma_k) - \left(\frac{\lambda_1 \dot{V}}{2(\dot{\chi}-\dot{\gamma})^2} + \frac{\lambda_2 V_k}{2(\dot{\chi}-\dot{\gamma})}\right)\cos(\chi_k - \\ &-\gamma_k) + \lambda_3\left(\frac{V_k\sin\chi_k}{\dot{\chi}} + \frac{\dot{V}\cos\chi_k}{\dot{\chi}^2}\right), \\ \dot{\gamma} &= \delta_1 U_{1max}, \quad \dot{\chi} = \delta_2 U_{2max} \quad \text{and} \quad \dot{V} = \delta_3 U_{3max}, \\ \gamma_k &= \gamma(t=t_k), \quad \chi_k = \chi(t=t_k) \quad \text{and} \quad V_k = V(t=t_k)\end{aligned} \tag{3.57}$$

Let us now try to define the cases when singularity on controls occurs.

Singularity on u_1***:*** In this case $\lambda_4 = \dot{\lambda_4} = 0$ and $\gamma = \gamma_k$. So, by substitution the value of γ in the equation $\dot{\lambda_4} = 0$ the singularity occurs when: $V = 0$ or $\tan\gamma_k = \frac{\lambda_3}{\lambda_1\cos\chi + \lambda_2\sin\chi}$

Singularity on u_2***:*** In this case, $\lambda_5 = \dot{\lambda_5} = 0$ and $\chi = \chi_k$. So, from the equation $\dot{\lambda_5} = 0$ the singularity occurs when:
$y = \frac{\lambda_2}{\lambda_1}x + (\lambda_1 w_y - \lambda_2 w_x)t + \lambda_{5c}$, Where λ_{5c} is the integral constant. The previous equation present a straight line in the plane if there is no wind.

Singularity on u_3***:*** In this case $\lambda_6 = \dot{\lambda_6} = 0$ and $v = v_k$. So, from the equation $\dot{\lambda_6} = 0$ that the singularity occurs when:
$z = -\frac{\lambda_1}{\lambda_3}x - \frac{\lambda_2}{\lambda_3}y + (\lambda_1 w_x + \lambda_2 w_y + \lambda_3 w_z)t + \lambda_{6c}$ where λ_{6c} is the integral constant. The previous equation present a straight line in the space if there is no wind. In addition, as there is a limitation on velocity, the condition $V = V_{max}$ is added as an additional case when a singularity on u_3 must occurs.

The remaining problem is to find the initial values of co-state variables such that the two-point boundary value problem formulated by Pontryagin minimum principle is solved, i.e. finding the initial values of co-state variables so that the corresponding switching times allow to steer the system from the given initial configuration point to the desired final one.

Optimal Trajectories Analysis

This paragraph presents the time profiles of x, y, z, γ, χ and V for a time interval $t \in [t_k \quad t_{k+1}]$ under two types of control inputs, i.e. regular ($\delta_i U_{imax} \in \{+1, -1\}$) and singular ($\delta_i U_{imax} = 0$) control. Only the final result of the analytical integrations is given:

– ***Regular arcs:***

For regular control inputs, 8 different arcs exist as shown in the table (3.1).

	u_1	u_2	u_3
Regular arc type 1	$+U_{1max}$	$+U_{2max}$	$+U_{3max}$
Regular arc type 2	$+U_{1max}$	$+U_{2max}$	$-U_{3max}$
Regular arc type 3	$+U_{1max}$	$-U_{2max}$	$+U_{3max}$
Regular arc type 4	$+U_{1max}$	$-U_{2max}$	$-U_{3max}$
Regular arc type 5	$-U_{1max}$	$+U_{2max}$	$+U_{3max}$
Regular arc type 6	$-U_{1max}$	$+U_{2max}$	$-U_{3max}$
Regular arc type 7	$-U_{1max}$	$-U_{2max}$	$+U_{3max}$
Regular arc type 8	$-U_{1max}$	$-U_{2max}$	$-U_{3max}$

TABLE 3.1: The different arc types of regular control

So, the general arc of regular control is given as follows:

$$\begin{aligned} x = x_k + \tfrac{v\sin(\chi+\gamma)-V_k\sin(\chi_k+\gamma_k)}{2(\delta_2 U_{2max}+\delta_1 U_{1max})} + \tfrac{\delta_3 U_{3max}(\cos(\chi+\gamma)-\cos(\chi_k+\gamma_k))}{2(\delta_2 U_{2max}+\delta_1 U_{1max})^2} + \\ + \tfrac{v\sin(\chi-\gamma)-V_k\sin(\chi_k-\gamma_k)}{2(\delta_2 U_{2max}-\delta_1 U_{1max})} + \tfrac{\delta_3 U_{3max}(\cos(\chi-\gamma)-\cos(\chi_k-\gamma_k))}{2(\delta_2 U_{2max}-\delta_1 U_{1max})^2} + w_x(t-t_k) \end{aligned} \tag{3.58}$$

$$\begin{aligned} y = y_k - \tfrac{v\cos(\chi+\gamma)-V_k\cos(\chi_k+\gamma_k)}{2(\delta_2 U_{2max}+\delta_1 U_{1max})} + \tfrac{\delta_3 U_{3max}(\sin(\chi+\gamma)-\sin(\chi_k+\gamma_k))}{2(\delta_2 U_{2max}+\delta_1 U_{1max})^2} - \\ - \tfrac{v\cos(\chi-\gamma)-V_k\cos(\chi_k-\gamma_k)}{2(\delta_2 U_{2max}-\delta_1 U_{1max})} + \tfrac{\delta_3 U_{3max}(\sin(\chi-\gamma)-\sin(\chi_k-\gamma_k))}{2(\delta_2 U_{2max}-\delta_1 U_{1max})^2} + w_y(t-t_k) \end{aligned} \tag{3.59}$$

$$z = z_k - \frac{v\cos\gamma - v_k\cos\gamma_k}{\delta_1 U_{1max}} + \frac{\dot{v}(\sin\gamma - \sin\gamma_k)}{U_{1max}^2} + w_z(t-t_k) \tag{3.60}$$

$$\gamma = \delta_1 U_{1max}(t-t_k) + \gamma_k \tag{3.61}$$

$$\chi = \delta_2 U_{2max}(t-t_k) + \chi_k \tag{3.62}$$

$$v = \delta_3 U_{3max}(t-t_k) + v_k \tag{3.63}$$

– ***Singular arcs:***

In this case,7 different arcs exist as shown in the table (3.2).

The equations of the first type of singular arcs: In this case $u_1 = u_2 = u_3 = 0$. Which lead us to:

$$\begin{aligned} x &= (v_k\cos\chi_k\cos\gamma_k + w_x)(t-t_k) + x_k \\ y &= (v_k\sin\chi_k\cos\gamma_k + w_y)(t-t_k) + y_k \\ z &= (v_k\sin\gamma_k + w_z)(t-t_k) + z_k \\ \gamma &= \gamma_k, \quad \chi = \chi_k, \quad v = v_k \end{aligned} \tag{3.64}$$

	u_1	u_2	u_3
Singular arc type 1	0	0	0
Singular arc type 2	0	0	$\pm U_{3max}$
Singular arc type 3	0	$\pm U_{2max}$	0
Singular arc type 4	0	$\pm U_{2max}$	$\pm U_{3max}$
Singular arc type 5	$\pm U_{1max}$	0	0
Singular arc type 6	$\pm U_{1max}$	0	$\pm U_{3max}$
Singular arc type 7	$\pm U_{1max}$	$\pm U_{2max}$	0

TABLE 3.2: The different arc types of singular control

The equations of the second type of singular arcs: In this case $u_1 = u_2 = 0$ and $u_3 = \delta_3 U_{3max} = \pm U_{3max}$. Which lead us to:

$$\begin{aligned} x &= x_k + \left[\tfrac{\delta_3 U_{3max}}{2}(t^2 - t_k^2) + (v_k - \delta_3 U_{3max} t_k)(t - t_k)\right]\cos\chi_k \cos\gamma_k + w_x(t - t_k) \\ y &= y_k + \left[\tfrac{\delta_3 U_{3max}}{2}(t^2 - t_k^2) + (v_k - \delta_3 U_{3max} t_k)(t - t_k)\right]\sin\chi_k \cos\gamma_k + w_y(t - t_k) \\ z &= z_k + \left[\tfrac{\delta_3 U_{3max}}{2}(t^2 - t_k^2) + (v_k - \delta_3 U_{3max} t_k)(t - t_k)\right]\sin\gamma_k + w_z(t - t_k) \\ \gamma &= \gamma_k, \qquad \chi = \chi_k \\ v &= \delta_3 U_{3max}(t - t_k) + v_k \end{aligned} \tag{3.65}$$

The equations of the third type of singular arcs: In this case $u_1 = u_3 = 0$ and $u_2 = \delta_2 U_{2max} = \pm U_{2max}$. Which lead us to:

$$\begin{aligned} x &= x_k + \tfrac{v_k \cos\gamma_k}{\delta_2 U_{2max}}(\sin\chi - \sin\chi_k) + w_x(t - t_k) \\ y &= y_k - \tfrac{v_k \cos\gamma_k}{\delta_2 U_{2max}}(\cos\chi - \cos\chi_k) + w_y(t - t_k) \\ z &= z_k + (v_k \sin\gamma_k + w_z)(t - t_k) \\ \chi &= \delta_2 U_{2max}(t - t_k) + \chi_k \\ \gamma &= \gamma_k, \qquad v = v_k \end{aligned} \tag{3.66}$$

The equations of the fourth type of singular arcs: In this case $u_1 = 0$, $u_2 = \delta_2 U_{2max} = \pm U_{2max}$ and $u_3 = \delta_3 U_{3max} = \pm U_{3max}$. Which lead us to:

$$\begin{aligned} x &= x_k + \cos\gamma_k \left[\tfrac{v\sin\chi - v_k \sin\chi_k}{\delta_2 U_{2max}} + \tfrac{\delta_3 U_{3max}}{U_{2max}^2}(\cos\chi - \cos\chi_k)\right] + w_x(t - t_k) \\ y &= y_k + \cos\gamma_k \left[-\tfrac{v\cos\chi - v_k \cos\chi_k}{\delta_2 U_{2max}} + \tfrac{\delta_3 U_{3max}}{U_{2max}^2}(\sin\chi - \sin\chi_k)\right] + w_y(t - t_k) \\ z &= z_k + \left[\tfrac{\delta_3 U_{3max}}{2}(t^2 - t_k^2) + (v_k - \delta_3 U_{3max} t_k)(t - t_k)\right]\sin\gamma_k + w_z(t - t_k) \\ \chi &= \delta_2 U_{2max}(t - t_k) + \chi_k \\ v &= \delta_3 U_{3max}(t - t_k) + v_k \\ \gamma &= \gamma_k \end{aligned} \tag{3.67}$$

The equations of the fifth type of singular arcs: In this case $u_1 = \delta_1 U_{1max} = \pm U_{1max}$ and $u_2 = u_3 = 0$. Which lead us to:

$$\begin{aligned} x &= x_k + \tfrac{v_k \cos\chi_k}{\delta_1 U_{1max}}(\sin\gamma - \sin\gamma_k) + w_x(t - t_k) \\ y &= y_k + \tfrac{v_k \sin\chi_k}{\delta_1 U_{1max}}(\sin\gamma - \sin\gamma_k) + w_y(t - t_k) \\ z &= z_k - \tfrac{v_k}{\delta_1 U_{1max}}(\cos\gamma - \cos\gamma_k) + w_z(t - t_k) \\ \gamma &= \delta_1 U_{1max}(t - t_k) + \gamma_k \\ \chi &= \chi_k, \qquad v = v_k \end{aligned} \tag{3.68}$$

The equations of the sixth type of singular arcs: In this case $u_1 = \delta_1 U_{1max} = \pm U_{1max}$, $u_2 = 0$ and $u_3 = \delta_3 U_{3max} = \pm U_{3max}$. Which lead us to:

$$\begin{aligned} x &= x_k + \cos\chi_k \left[\tfrac{v\sin\gamma - v_k\sin\gamma_k}{\delta_1 U_{1max}} + \tfrac{\delta_3 U_{3max}}{U_{1max}^2}(\cos\gamma - \cos\gamma_k)\right] + w_x(t - t_k) \\ y &= y_k + \sin\chi_k \left[\tfrac{v\sin\gamma - v_k\sin\gamma_k}{\delta_1 U_{1max}} + \tfrac{\delta_3 U_{3max}}{U_{1max}^2}(\cos\gamma - \cos\gamma_k)\right] + w_y(t - t_k) \\ z &= z_k - \tfrac{v\cos\gamma - v_k\cos\gamma_k}{\delta_1 U_{1max}} + \tfrac{\delta_3 U_{3max}}{U_{1max}^2}(\sin\gamma - \sin\gamma_k) + w_z(t - t_k) \\ \gamma &= \delta_1 U_{1max}(t - t_k) + \gamma_k \\ v &= \delta_3 U_{3max}(t - t_k) + v_k \\ \chi &= \chi_k \end{aligned} \tag{3.69}$$

The equations of the seventh type of singular arcs: In this case $u_1 = \delta_1 U_{1max} = \pm U_{1max}$, $u_2 = \delta_2 U_{2max} = \pm U_{2max}$ and $u_3 = 0$. Which lead us to:

$$\begin{aligned} x &= x_k + \tfrac{v_k}{2}\left(\tfrac{\sin(\chi-\gamma) - \sin(\chi_k - \gamma_k)}{\delta_2 U_{2max} - \delta_1 U_{1max}} + \tfrac{\sin(\chi+\gamma) - \sin(\chi_k + \gamma_k)}{\delta_2 U_{2max} + \delta_1 U_{1max}}\right) + w_x(t - t_k) \\ y &= y_k - \tfrac{v_k}{2}\left(\tfrac{\cos(\chi-\gamma) - \cos(\chi_k - \gamma_k)}{\delta_2 U_{2max} - \delta_1 U_{1max}} + \tfrac{\cos(\chi+\gamma) - \cos(\chi_k + \gamma_k)}{\delta_2 U_{2max} + \delta_1 U_{1max}}\right) + w_y(t - t_k) \\ z &= z_k - \tfrac{v_k}{\delta_1 U_{1max}}(\cos\gamma - \cos\gamma_k) + w_z(t - t_k) \\ \gamma &= \delta_1 U_{1max}(t - t_k) + \gamma_k \\ \chi &= \delta_2 U_{2max}(t - t_k) + \chi_k \\ v &= v_k \end{aligned} \tag{3.70}$$

Numerical Solution

The basic approach for solving the optimal control problem described above is to transform it into a sequence of nonlinear constrained optimization problems by discretizing the control and/or state variables. This technique is known as ***Direct Collocation Approach*** [156]. In this study, we employ a Nonlinear Programming solver using MATLAB® with respect to the discretized control. The corresponding discretized state variables are determined recursively using a numerical integration scheme (e.g. Euler, Heun, Runge-Kutta, etc.) [30].

Therefore, the time interval $[t_0, t_f]$ is divided into N nodes as follows

$$t_0 = \tau_1 \leq \tau_2 \leq t_3 \leq \cdots \leq \tau_N = t_f \tag{3.71}$$

such that,

$$\tau_k = t_0 + (k-1)\cdot h; \quad h := \frac{t_f - t_0}{N-1}, k = 1, \ldots, N \tag{3.72}$$

Thus, the vector of decision variables is composed of the control inputs over all nodes and the final time t_f as it is shown below

$$\xi = \left[t_f, U_1^T, U_2^T, \ldots, U_N^T\right] \in \mathbb{R}^{N_\xi}; \quad N_\xi = 3N + 1 \tag{3.73}$$

and the state variables are computed recursively using Euler approximation applied to the differential equations (3.32) which yields to the following

$$X_{k+1} = X_k + h \cdot f\left(X_k, U_k\right); \quad k = 1, \ldots, N-1 \tag{3.74}$$

with the initial and final conditions given in Equation (3.33) and Equation (3.34). From the previous, the problem of optimal control can be described as following

$$\begin{aligned}
&\text{Minimize} && J = t_f && \text{(3.75a)}\\
&\text{Subject to} && \dot{X}_{k+1} = f\left(X_k, U_k\right) && \text{(3.75b)}\\
& && X(t_1) = X_0 && \text{(3.75c)}\\
& && X(t_N) = X_f && \text{(3.75d)}\\
& && U_{min} \leq U_k \leq U_{max} && \text{(3.75e)}\\
& && S\left(X_k\right) \leq 0 && \text{(3.75f)}
\end{aligned}$$

where, $k = 1, 2, \ldots, N$, U_{min} and U_{max} are lower and upper bounds on control inputs across all nodes. The k^{th} set of control rate limits, expressed in Equation (3.75e), is given as

$$\begin{bmatrix} \dot{\gamma}_{min} \\ \dot{\chi}_{min} \\ \dot{V}_{min} \end{bmatrix} \leq \begin{bmatrix} \dot{\gamma}_k \\ \dot{\chi}_k \\ \dot{V}_k \end{bmatrix} \leq \begin{bmatrix} \dot{\gamma}_{max} \\ \dot{\chi}_{max} \\ \dot{V}_{max} \end{bmatrix} \tag{3.76}$$

While $S\left(X_k\right)$ denotes the constraints enforced on the path. The k^{th} set of path restrictions, presented in Equation (3.75f), is shown as

$$\begin{bmatrix} \gamma_{min} \\ V_{min} \end{bmatrix} \leq \begin{bmatrix} \gamma_k \\ V_k \end{bmatrix} \leq \begin{bmatrix} \gamma_{max} \\ V_{max} \end{bmatrix} \tag{3.77a}$$

$$r_p^2 - (x_k - x_c)^2 - (y_k - y_c)^2 - z_k^2 \leq 0 \tag{3.77b}$$

Simulation Results

In order to illustrate the performance of the proposed trajectory generation approach, two scenarios for simulations were carried out. Both are Point to Point trajectory

with the same initial and final conditions. The first one presents a trajectory without wind. While the second one take the presence of the wind into consideration. The vehicle constraints and the initial/final configurations are provided in Tab. 3.7 and Tab. 3.8 respectively. The wind is supposed to be a constant having the value of : $W = [W_x, W_y, W_z] = [2, 2, 1]$.

The trajectory in each case is discretized over 70 nodes, and the interior point algorithm included in *fmincon* nonlinear programming solver in MATLAB® is used to compute optimal trajectory. The obtained trajectory is described in Figure (3.9) and Figure (3.10). The red and solid line denotes the case of the presence of the wind. The blue dashed line is the trajectory without wind. The green arrows describe the direction of the wind.

TABLE 3.3: Vehicle's restrictions.

Minimum	Variable	Maximum	Units
-20	γ	20	deg
-5	V	5	m/sec
-5	$\dot{\gamma}$	5	deg/sec
- 15	$\dot{\chi}$	15	deg/sec
- 1.25	$\dot{V}$	1.25	m/sec^2

TABLE 3.4: Initial and final configurations.

	$x\ [m]$	$y\ [m]$	$z\ [m]$	$\gamma\ [Deg]$	$\chi\ [Deg]$	$V\ [m/sec]$
Initial value	0	0	0	0	0	3
Desired value	30	20	10	0	0	4

Notice that the magnitude of the wind in this scenario is $||W|| = 3[m/sec]$ which presents 60% of the vehicle's maximal velocity. Remark that the final configuration is reached in $t_f = 7.21[sec]$ in the presence of the wind, while it requires $t_f = 10.8[sec]$ to be attained if there is no wind. Such results are consistent in the sense where the wind push the vehicle from the back. However, if the wind comes from the front of the vehicle, it is obvious that it will slow down the velocity and consequently the aircraft will require more time to reach its target.

3.3.2.2 Sub-Optimal Trajectory Generation Algorithm

In this section, we introduce a trajectory generation algorithm based on geometrical solution to join two points. The provided trajectory is near-optimal, and in some cases,

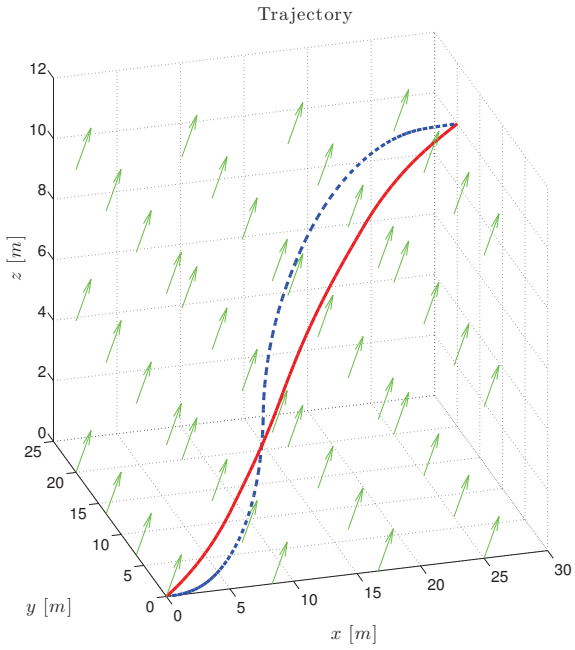

FIGURE 3.9: 3D view of the trajectory.

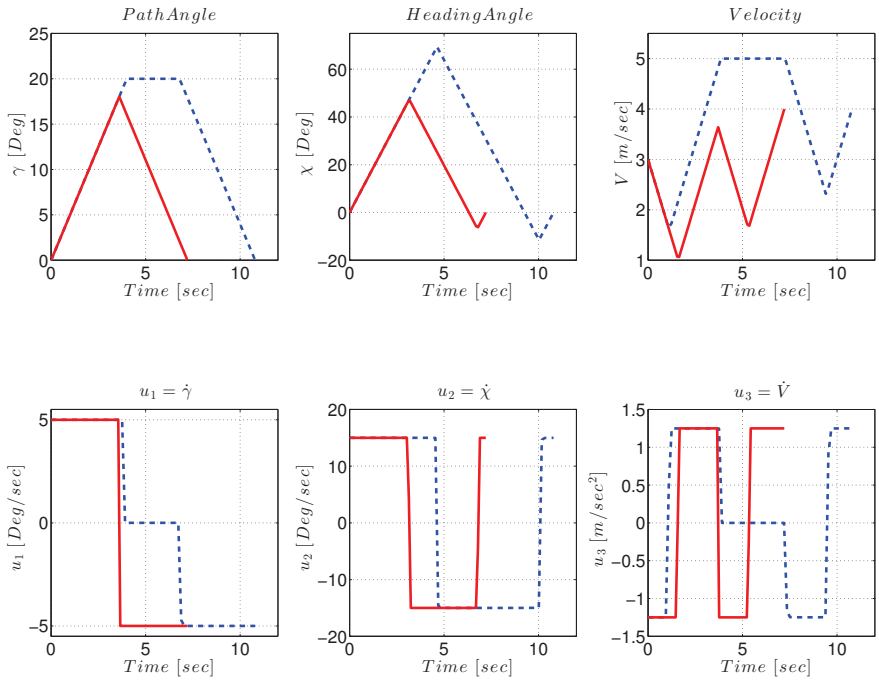

FIGURE 3.10: Time profile of flight path angle γ, heading χ, velocity V and their rate of change $\dot{\gamma}$, $\dot{\chi}$, $\dot{V}$ respectively.

it can be optimal. This problem can be seen as a variant of Dubins problem introduced in [51] and mentioned previously in Sec. 3.2.1. In fact, many publications on Dubins-based solutions were published in the last years. For instance, the authors in [168, 172] base upon Dubins solutions to generate 2-D paths satisfying kinematic and tactical constraints. While an extension is proposed in [5] to guarantee the passage of the path through (or close) assigned waypoints. A generalization of Dubins approach to 3D space was also proposed in [4, 78]. In all these papers, the vehicle was considered to move with a constant velocity and only a geometric path was considered.

Our approach consists of finding a trajectory to steer an aerial vehicle flying at a fixed altitude from its initial point to a predefined target point in minimal time. The trajectory includes time profile of x, y coordinates, heading angle (χ), velocity (V), rate of change of the heading angle ($\dot{\chi}$), and the acceleration ($\dot{V}$). It has been shown in [51] that for each two points separated by a distance greater than twice of turning radius, the shortest path for a vehicle moving with a constant velocity is composed by the union of an arc of circumference, a segment, and again an arc of circumference. In our case, the velocity is not a constant, thus the circumference arc is replaced by a spiral.

Let $S_i \in \{S_{i_{acc}}, S_{i_{dec}}\}$ denotes the initial spiral so that it starts from the initial configuration q_0. Where, $S_{i_{acc}}$ presents the acceleration phase (from V_0 to V_{max}) and $S_{i_{dec}}$ the deceleration phase (from V_0 to V_{min}). Similarly, assume that $S_f \in \{S_{f_{acc}}, S_{f_{dec}}\}$ describes the final spiral with $S_{f_{acc}}$ for V: $V_{min} \rightarrow V_f$, and $S_{f_{dec}}$ for V: $V_{max} \rightarrow V_f$. In addition, consider that Σ_{i_j} represents a circumference containing the end configuration of S_i while Σ_{f_j} is a circumference containing the start configuration of S_f. The index j defines the radius of the circumference with respect to the velocity (max for V_{max} and min for V_{min}). Then, the proposed algorithm is

Algorithm 3.1 Sub-Optimal Trajectory Generation

Require: initial and final configurations (q_0 and q_f).
1: Determine the turning direction for S_i and S_f.
2: Compute S_i and S_f.
3: **for** Each couples of $(S_{i_{acc}}, S_{f_{acc}})$, $(S_{i_{acc}}, S_{f_{dec}})$, $(S_{i_{dec}}, S_{f_{acc}})$, and $(S_{i_{dec}}, S_{f_{dec}})$ **do**
4: Compute L = common tangent.
5: **if** L is empty. **then**
6: Introduce Σ_{i_j} and Σ_{f_j}.
7: Compute L = common tangent.
8: **end if**
9: Store required time t_{total}, rate of change of heading $\dot{\chi}$, and acceleration $\dot{V}$ in a lookup table Π.
10: **end for**
11: Select the minimal time in Π and its corresponding trajectory.

Example 3.1. *2D fast trajectory generation for UAV*

In this example, we illustrate the functioning of the previous algorithm step by step. Considering an aerial vehicle with constraints on heading angle, velocity and control inputs as shown in Tab. 3.5. While the initial and final configurations are presented in Tab. 3.6.

TABLE 3.5: Example 3.1: Restrictions on state and control variables.

Minimum	Variable	Maximum	Units
0.01	V	1	m/sec
- 10	$\dot{\chi}$	10	deg/sec
- 0.12	$\dot{V}$	0.12	m/sec^2

TABLE 3.6: Example 3.1: Initial and desired orientation and velocity values.

	$x\ [m]$	$y\ [m]$	$\chi\ [Deg]$	$V\ [m/sec]$
Initial value	0	0	0	0.1
Desired value	5	5	0	0.02

Step 1: *Determine the turning direction for S_i and S_f*

In order to determine the turning direction we introduce a new coordinate system $\mathcal{R}_{\mathcal{N}}(A_{RP}, x_{\mathcal{N}}, y_{\mathcal{N}})$ related to the aerial vehicle. So that, the $y_{\mathcal{N}}$-axis coincide with the heading of the aircraft. Then x_n-axis is perpendicular to y_n-axis and points to the right. Consequently, the coordinate space can be decomposed into two sub-spaces separated by the y_n-axis. Finally the projection of the final (or initial) point on x_n-axis defines direction of the turn: to the right for a positive value and to the left for a negative one as shown in Figure (3.11).

In our case the projection of x_f on x_n-axis (according to q_0) is negative which means that the turning direction of the initial Spiral is to the left. On the other side, the projection of x_0 on x_n-axis (with respect to q_f) is positive, so the turning direction of S_f is to the right. The transformation relationship between $R_{\mathcal{I}}$ and $R_{\mathcal{N}}$ is detailed in appendix A.

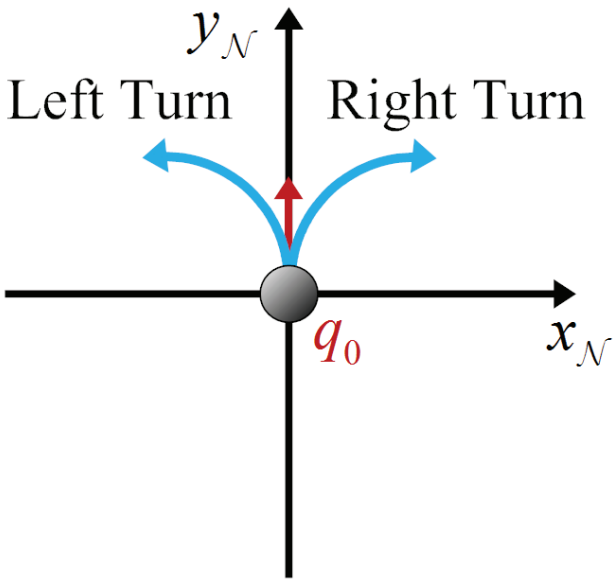

FIGURE 3.11: $R_{\mathcal{N}}$ coordinate system.

Step 2: *Compute S_i and S_f*

The spiral arcs are obtained through the following equations:

$$x = \frac{V\sin\chi - V_0\sin\chi_0}{\dot{\chi}} + \frac{\dot{V}}{\dot{\chi}^2}(\cos\chi - \cos\chi_0) + x_0 \tag{3.78a}$$

$$y = -\frac{V\cos\chi - V_0\cos\chi_0}{\dot{\chi}} + \frac{\dot{V}}{\dot{\chi}^2}(\sin\chi - \sin\chi_0) + y_0 \tag{3.78b}$$

$$\chi = \dot{\chi}t + \chi_0 \tag{3.78c}$$

$$V = \dot{V}t + V_0 \tag{3.78d}$$

where, $\dot{\chi}$ is defined through the turning direction, $\dot{V}$ depends on the phase (acceleration/deceleration), and the required time is obtained by Equation (3.78d). Figure (3.12) shows the case for $S_{i_{acc}}$ and $S_{f_{dec}}$.

Step 3: *Compute the common tangent*

The problem of finding common tangent(s) between two functions is equivalent to be determining two abscissa so that the two coefficients which describe the tangent at respective points are matched. This is like finding an intersection of two parametric curves in $\mathbb{R}^2$, where each axis of the plane corresponds to a coefficient of the tangent.

Let the x and y coordinates of S_i and S_f, calculated in Step 2, be characterized

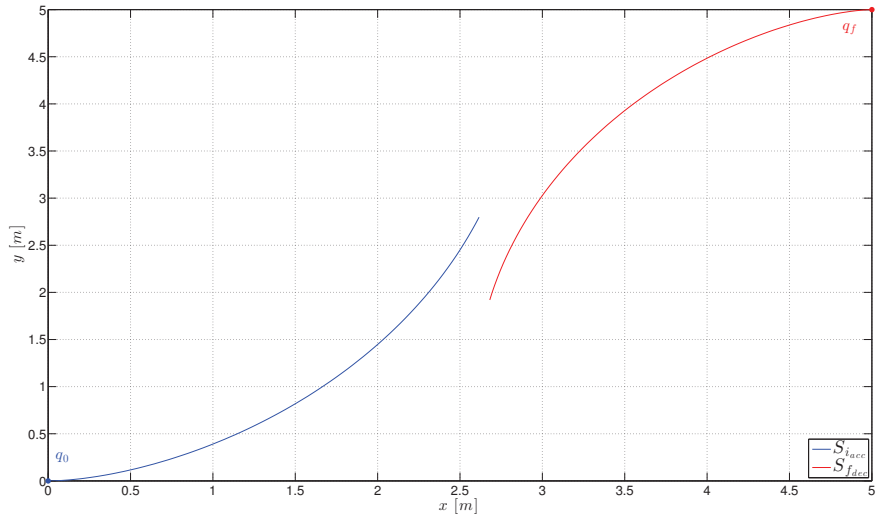

FIGURE 3.12: Computing initial and final spirals: the case of $S_{i_{acc}}$ and $S_{f_{dec}}$.

by a cubic-spline as follows:

$$y_{S_i} = G_1(x_1) = \varsigma_3 x_1^2 + \varsigma_2 x_1^2 + \varsigma_1 x_1 + \varsigma_0 \tag{3.79a}$$

$$y_{S_f} = G_2(x_2) = \iota_3 x_2^2 + \iota_2 x_2^2 + \iota_1 x_2 + \iota_0 \tag{3.79b}$$

where $x_1 \in \{x_0 \rightarrow x_{S_i}(end)\}$, and $x_2 \in \{x_{S_f}(start) \rightarrow x_f\}$. Then the set of tangents to S_i can be defined by the equation

$$y_{G_1} = a_{G_1} x + b_{G_1} \tag{3.80}$$

where, a_{G_1} is the slope of the line and b_{G_1} is the constant term, called also y-intercept. In addition, the slope of the tangent line is given by

$$a_{G_1} = \frac{df(x_1)}{x_1} \tag{3.81}$$

Therefore, from Equation (3.80) and Equation (3.81), we state the first parametric curve as bellow

$$a_{G_1} = 3\varsigma_3^2 x_1^2 + 2\varsigma_2 x_1 + \varsigma_1 \tag{3.82a}$$

$$b_{G_1} = y_{G_1} - a_{G_1} x_1 \tag{3.82b}$$

In a similar way, the parametric curve describing the set of all tangents to S_f is given by

$$a_{G_2} = 3\iota_3^2 x_1^2 + 2\iota_2 x_1 + \iota_1 \tag{3.83a}$$

$$b_{G_2} = y_{G_2} - a_{G_2} x_2 \tag{3.83b}$$

Consequently, the common tangent is defined by the intersection of the two curves defined by Equation (3.82a), Equation (3.82b), Equation (3.83a), and Equation (3.83b). This process is illustrated in Figure (3.13).
Once the tangent is obtained, it is important to define the velocity variation depending on the velocity values at the end of selected S_i and the beginning of S_f, as well as the traveled distance along the tangent. More details on this procedure can be found in appendix B.

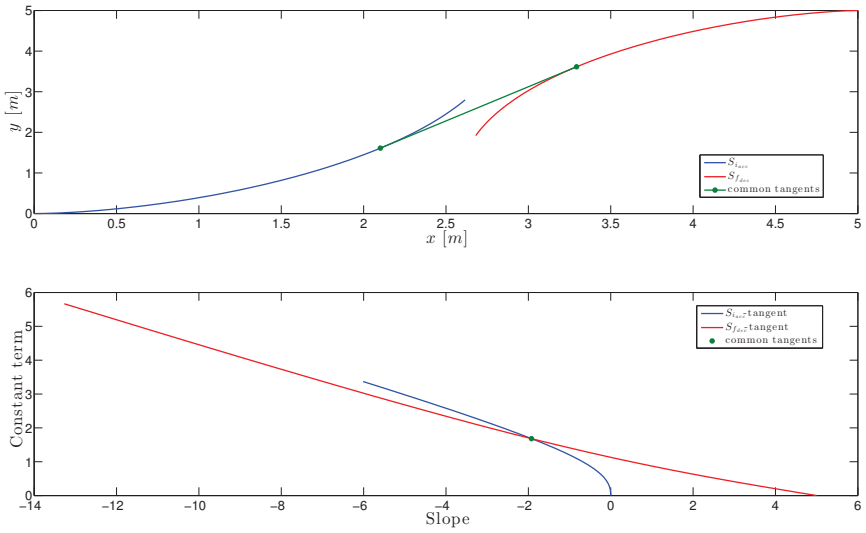

FIGURE 3.13: Computing common tangent between $S_{i_{acc}}$ and $S_{f_{dec}}$.

After repeating Steps 2 and 3 for all possible combinations of S_i and S_f we obtain an optimal trajectory illustrated in Figure (3.14) while the corresponding heading angle, velocity, and control inputs are presented in Figure (3.15). Note that in this example we didn't need to include circumference arcs to the trajectory. Computing circular arcs and finding common tangent between them is presented in appendix C.

A generalization of the proposed algorithm to include the presence of a constant wind and in 3-dimension space can be done by using the equations (3.58 - 3.70) developed previously in *"Optimal Trajectory Analysis"* paragraph.

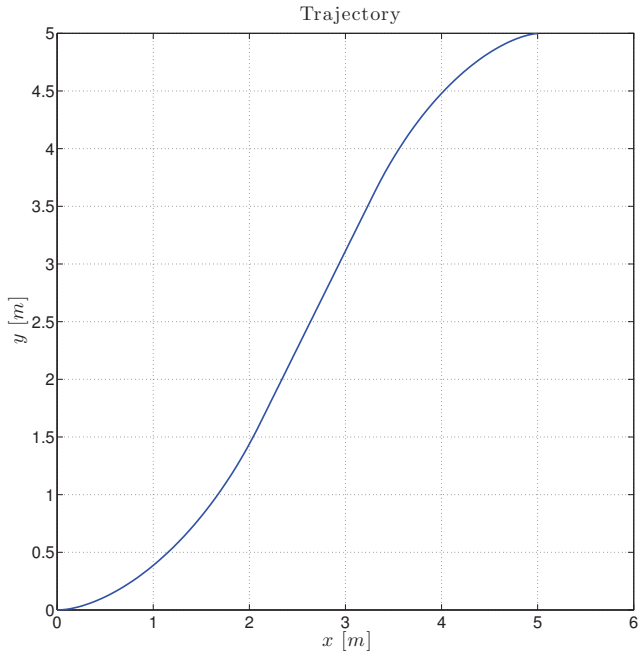

FIGURE 3.14: 2D-optimal trajectory.

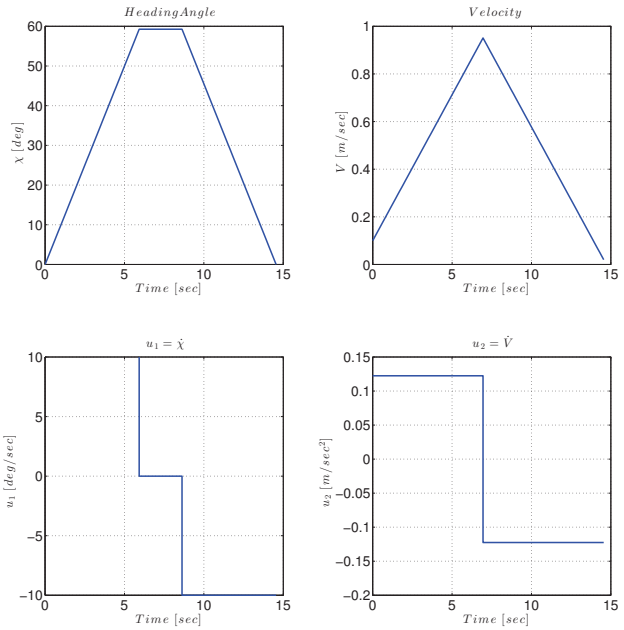

FIGURE 3.15: Time profile of heading, velocity and control inputs along the trajectory.

3.3.3 Updated Flight Planning

Flight planning involves creating a plan to guide an aerial vehicle from its initial position to a destination (way-)point. A mission describes the operation of a UAV in a given region, during a certain period of time while pursuing a specific objective. A flight plan is defined as the ordered set of movements executed by the aerial vehicle during a mission. It can be decomposed in phases. Each phase is described by the coordinates of a pair of way-points and by the speed at which the airship is to fly between these way-points. A phase is completed when the second way-point is reached by the aerial vehicle. Along the way, there may be a set of regions to visit and a set of regions to avoid. In addition, the traveling object may have certain motion constraints. The mission planning strategy could be either static or dynamic depending on whether the mission planning problem is to create a path in static or in dynamic environment. Flight planning routines attempt to create paths that are fully consistent with the physical constraints of the vehicle, the obstacle avoidance, shortest and optimum flight path and weighed regions. Weighed regions are regions with abnormally low or high pressure, wind speeds or any other factor affecting flight. Mission planning in an autonomous vehicle provides the level of autonomy by having minimal ground control. Vehicle autonomy is a discipline fertilized by the robotics and computer science fields.
The evolution of air traffic is subject to many natural sources of uncertainty, roughly classified into two classes.

- Nominal uncertainty, that affects all flights, result of generic perturbations due to weather, variability in the mass of the airship, variability in the settings of the Flight Management System. Nominal uncertainty gives rise to quantitative differences between the actual air traffic and the model used to predict its evolution.
- Non nominal uncertainty that affects certain flights, result of things as malfunctions, errors of the human operators, extreme weather conditions. Non nominal uncertainty gives rise to qualitative structural differences between the actual air traffic and the model used to predict its evolution.

Flight planning requires an awareness of the environment in which it is operating. Maps and charts are available and can be encoded into digital forms which are then stored onboard the lighter than air robot. The rapid advances in computing technology have made it possible to construct a digital representation of the airspace environment which incorporates terrain, airspace boundaries, weather and other aircrafts. This digital world is the key to constructing the situational awareness required by the mission planner to perform onboard mission planning. The assumption is made that the information required will be available. Consideration is not given in this section to the sensing side of the problem. The vehicle must be aware of the location of other aircrafts and weather

(e.g. storms). A number of methods exists for obtaining this information, as a radar based system of the airborne dependent surveillance broadcast can provide the location and speed of nearby aircraft.

Similarly, weather radar can also provide information for adverse weather condition. In addition to the location of the entities within the world, their dimensions must be known. Once the digital representation of the world (i.e. situational awareness) has been created, high level activities such as mission planning can be performed.

In this study, the idea is simple: as uncertainty always exists due to numerous reasons, a way of taking care of it is to update periodically the reference paths. If due to the wind effect, the aerial vehicle has overshoot the way point it was supposed to go through, then the next way point should be considered. The immediate measurements of the position and orientation are taken as initial conditions for the next reference trajectories.

3.3.3.1 Basic Problem Statement

The basic problem is to move the vehicle from an initial configuration to the next configuration. This question was solved in the previous section. The actual measured wind velocity is assumed to be constant between two updates of the reference trajectories. The constraints are first expressed in terms of thrust and velocities and then transformed into limitations on flight path and heading angles.

3.3.3.2 Hierarchical Planning Structure

The structure necessary for the update of the reference trajectories is described in this paragraph. A hierarchical structure appears because of the choice of the predicted arrival time at each update. There exists an upper level of decision making and a lower level where the time optimal problem is solved. One clock, associated with the lower level, allows the reference trajectories computing. It is reset at each update. The following concepts must be introduced to describe this hierarchical structure operation

- Periodic Updates : First, periodic updates are introduced. The period Δ may be 10 to 100 times greater than the integration step of the differential equation. This depends on the nature of the next way-point: fly-by, fly-over or destination point.
- Anticipated Updates : An important perturbation may occur between two periodic updates. To handle this situation, a new reference trajectory is required. The concept of anticipated update is thus important. To decide whether an important perturbation occurs the System needs a supervision level.
- Supervision level: The supervision system is based on a spatiotemporal criterion.

- The temporal part: It depends on the value of a parameter called e_{ct} computed continuously in the lower level. The parameter e_{ct} gives an evaluation of the arrival time predicted in the k^{th} update and the arrival time predicted continuously on the basis of the measured configuration, T_c:

$$e_{ct} = |T^k - T_c - h| \tag{3.84}$$

  where h is the time given by the clock. If $e_{ct} > \varepsilon_{min}$ then an update occurs; ε_{min} is user-fixed relatively to the environment
- The spatial part: The spatial parameter is defined as $e_s = |X_r - X|$ where X_r is the reference configuration and X the measured one. If $e_x > \delta_{min}$ then an update must take place; ε_{min} as δ_{min} influence the number of updates.

Simulation results

In order to illustrate the updated flight planning idea, some simulations were carried out. For this purpose, we take just the spatial part of supervision level shown in the previous section. The proposed scenario consist of updating point to point trajectory with respect to wind variation. The vehicle's limitations and the initial/final configurations are provided in Tab. 3.7 and Tab. 3.8 respectively.

TABLE 3.7: Vehicle's restrictions.

Minimum	Variable	Maximum	Units
-0.35	γ	0.35	rad
-4	V	4	m/sec
-0.26	$\dot{\gamma}$	0.26	rad/sec
- 0.52	$\dot{\chi}$	0.52	rad/sec
- 1.25	$\dot{V}$	1.25	m/sec^2

TABLE 3.8: Initial and final configurations.

	$x\ [m]$	$y\ [m]$	$z\ [m]$	$\gamma\ [Rad]$	$\chi\ [Rad]$	$V\ [m/sec]$
Initial value	0	0	0	0.25	0.5	2
Desired value	16	14	20	0	0	0.5

The wind is supposed to be equal to zero on y and z axes and piecewise constant function (composed of three parts) on x axis. At the start of the simulation the value of the wind is considered to be $W_x = 0.3[m/sec]$ which is relatively weak (it presents about 7.5% of the vehicle's maximal velocity). At $t = 10[sec]$ the magnitude of the wind changes and becomes $W_x = 1.5[m/sec]$ producing 37.5% of V_{max}. Finally, at $t = 25[sec]$ wind speed

slow down to $W_x = 0.5[m/sec]$ (12.5% of V_{max}). Note that this value is equal to V_f. The previous is presented in Figure (3.16).

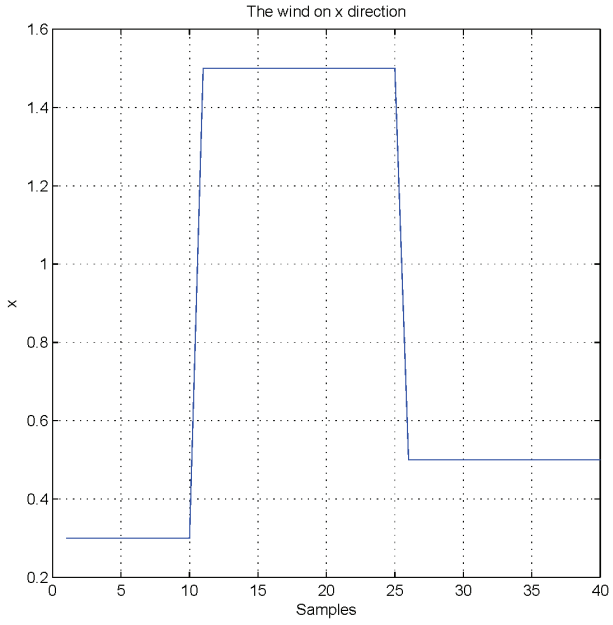

FIGURE 3.16: The wind variation on x axis.

The resulting updated reference trajectory is illustrated in Figure (3.17) - Figure (3.19). In those figures, the red line marked with triangle denotes the reference trajectories given for the UAV while the solid blue line describe the computed trajectory from the initial/actual configuration to the desired one.

A 3D view of the updated trajectory is shown in Figure (3.17). Whereas a side views of this trajectory in the plane $x.y$ and $x.z$ are shown in the Figure (3.18) and Figure (3.19) respectively.

Note that discontinuities on the reference trajectories are present. This is due to the updates, when the real situation is taken into account.

3.3.4 UAV Routing Problem for Bridge Inspection

In this part, we consider the problem of bridge inspection using a small UAV. This problem has been addressed in [70, 71]. In those papers, the authors have proposed two solutions based on a hybridization between Zermelo's navigation problem and TSP/VRP approaches. The wind was supposed to vary linearly and the pillar was assumed as points of interest to be visited by the vehicle. In other words, the configuration space was considered to be obstacle-free. In addition, each point was characterized by only

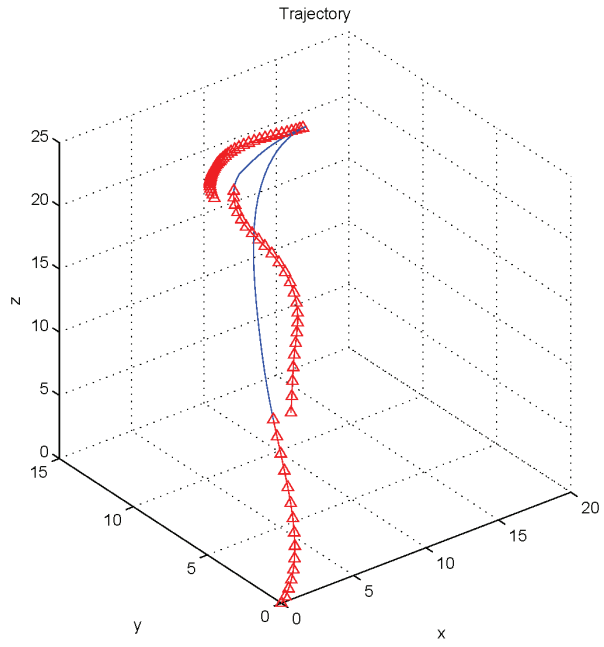

FIGURE 3.17: 3D view of the updated Trajectory.

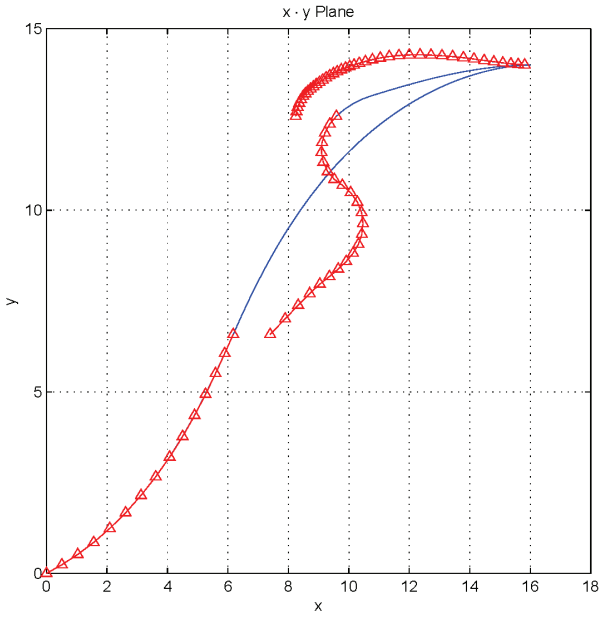

FIGURE 3.18: Side view of the updated Trajectory on the $x.y$ plane.

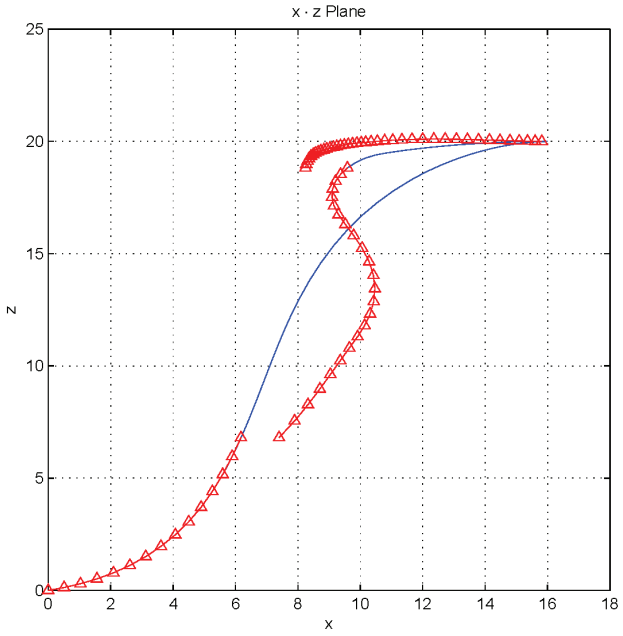

FIGURE 3.19: Side view of the updated Trajectory on the $x.z$ plane.

its position without including its orientation and velocity into account. Our work is a complement to those papers. In our approach, the Venturi effect (wind acceleration) is implemented and we take the presence of obstacles into consideration. Also, each point is defined by its position, orientation and velocity.

3.3.4.1 Problem Statement

The UAV routing problem for structures inspection can be formulated as follows

Let $q = \{q_1, q_2, ..., q_n\}$ be a set of points such that $q_2, ..., q_n$ are the points, situated on the bridge, which must be inspected and q_1 is the departing position representing the ground base.
Suppose that the travel cost matrix C represents the required flight time between every pair q_i, q_j; $i \neq j$.
In addition, assume that T_{Req}, called inspection vector, is a vector specifying the required time to inspect the point q_i.
Finally, consider that T_{Max} is the maximum time allowed for the vehicle to fly. Then, the UAV routing problem for structure inspection is to plan a set of tours in such a way that:

- Each point in $\{q_2, ..., q_n\}$ is visited once.
- All tours start and end at the ground base q_1.

- The maximum time allowed to fly is never violated.
- The total time required to all tours is the minimum.

It is important to state here that the travel cost matrix is asymmetric. So, the required time to fly from a point q_i with an orientation of γ_i, χ_i and a velocity of V_i and arriving at another point q_j with an orientation of γ_j, χ_j and a velocity of V_j may not be equal to the required time to fly in the inverse direction.

3.3.4.2 Capacitated Vehicle Routing Problem

The Capacitated Vehicle Routing Problem is an interesting approach for structures monitoring tasks. In fact, the limited carrying capacity of goods are replaced by an allowed maximum flight time, and the number of vehicle can be regarded as the number of vehicles required in UAVs fleet; **Or** the number of flights (for one aerial vehicle) needed to cover the points of interest. To solve such a problem, a wide variety of methods have been developed. These approaches are, frequently, classified as follows

- **Constructive Methods:** savings and insertion
- **Improvement Methods:** 2-change, 3-change, 2-relocate, 3-relocate, sweep, GENI.

In the following section, we present the **savings method** developed for solving CVRP.

Savings Method

The savings approach is an heuristic algorithm which was first proposed in 1964 by Clarke and Wright [36] to solve a CVRP in which the number of vehicles is free. Its basic idea is very simple. Considering a depot D and n demand points. Supposing that initially the solution to the VRP consists of using n vehicles and dispatching one vehicle to each one of the n demand points. Then, obviously, the total tour length of a such solution is $2\sum_{i=1}^{n} C(D,i)$.

Now, if we use a single vehicle to serve two points, e.g. i and j, on a single trip, the total distance traveled is reduced by the following amount

$$\begin{aligned} S(i,j) &= 2C(D,i) + 2C(D,j) - [C(D,i) + C(i,j) + C(D,j)] \\ &= C(D,i) + C(D,j) - C(i,j) \end{aligned} \tag{3.85}$$

The quantity $S(i,j)$ is known as the ***savings*** resulting from combining points i and j into a single tour. Whenever the value of $S(i,j)$ larger, combining i and j in a single tour becomes more desirable. However, i and j cannot be combined if the resulting tour violates one or more of the constraints of the VRP.

Before describing the algorithm, we introduce the following definition

Definition 3.8. A point i is said to be interior to a route if it is not adjacent to the depot D in the order of the traversal of points.

The savings algorithm of Clarke and Wright can now be presented (as expressed in [109]) as follows

Step 1 Compute the savings $S(i,j) = C(D,i) + C(D,j) - C(i,j)$ for $i,j = 2,..,n$, and $i \neq j$.

Step 2 Rank the savings in descending order of magnitude.

Step 3 For the savings $S(i,j)$ under consideration, include the arc (i,j) in a route if the constraints imposed on the route will not be violated, and if

a. Neither i nor j have already been assigned to a route. In this case, a new route is initiated including both i and j.

b. Exactly one of the two points (i or j) has already been included in an existing route and that point is not interior to that route. In this case, the arc (i,j) is added to the same route.

c. Both i and j have already been included in two different existing routes and neither point is interior to its route. In this case, both routes are merged.

Step 4 If the savings list $S(i,j)$ has not been exhausted, return to Step 3 and shift to the next entry in the list;
otherwise, stop.

Note that, any points that have not been assigned yet to a route during Step 3 must be served by a vehicle route that begins at the depot D visits the unassigned point and returns to D.

The Clarke-Wright algorithm can be programmed to run very efficiently and, since it involves very simple manipulations of the data set, it can be used with large-scale problems. Since nodes are added to routes once or twice at a time, an additional advantage of the algorithm is that it is possible to check whether each addition would violate any set of constraints, even when that set is quite complicated. For example, besides the constraints on maximum capacity and maximum distance, other constraints might be included, such as a maximum number of points that any vehicle may visit.

A number of variants of this method was proposed, e.g. [67, 135]. The Clarke and Wright algorithm suppose that the cost matrix C is symmetric, i.e. $C(i,j) = C(j,i)$. In addition it implicitly ignores vehicle fixed costs and fleet size. Vehicle costs f can easily be taken into account by adding this constant to every C_j $(j = 2,...,n)$. Solutions with a fixed number of vehicles can be obtained by repeating Step 3 until the required number of routes has been reached, even if the savings become negative.

TABLE 3.9: Base station and way-points to be visited during the traveled trajectory.

	$x\ [m]$	$y\ [m]$	$\chi\ [deg]$	$V\ [m/sec]$
q_1	0	0	0	0.1
q_2	10	2	90	0.1
q_3	14	6	180	0.1
q_4	10	10	-90	0.1
q_5	6	6	0	0.1
q_6	10	18	90	0.1
q_7	14	22	180	0.1
q_8	10	26	-90	0.1
q_9	6	22	0	0.1

Simulation Results

In this section we consider a scenario of bridge inspection mission. The vehicle is assumed to start from a base station (q_1) and to visit eight points situated on two pillars. These points are provided in Tab. 3.9, the vehicle's constraints in Tab. 3.7, and the entire environment is described in Figure (3.20). The wind is composed of two parts: the first is constant covering the whole environment with a magnitude of 0.2 $[m/sec]$, and the second characterizes Venturi effect, as it is described in Sec. 2.5.2, with a maximum magnitude of 0.07 $[m/sec]$. At first glance, it seems that the wind is very weak with a magnitude of 0.27 $[m/sec]$ (about 5.4% of V_{max}). But, remark that the desired velocity at each point is equal to 0.1 $[m/sec]$ which makes gives the wind speed an important weight with respect to vehicle's velocity (2.7 times bigger).

The required time for data collection at each point is estimated to be $T_{Req_i} = 5$ $[unit\ of\ time]$. While the vehicle is capable to flight during $T_{Max} = 68$ $[unit\ of\ time]$.

The cost matrix C is defined to be the required minimal time to connect all possible pair (q_i, q_j); $i \neq j$ which gives a matrix of dimension 9×9 with a 72 elements to be determined (excluding the diagonal items). For this end, the trajectory generation approach presented in 3.3.2, is employed for each case of C. Therefore, as in the previous simulations scenario, each trajectory is discretized over 70 nodes, and the interior point algorithm included in *fmincon* nonlinear programming solver in MATLAB® is used to find the optimal trajectory.

The resulted flight plan consists of three flights shown in Figure (3.21) - Figure (3.23).

The obtained solution includes three flight routes:
The first route, illustrated in Figure (3.21), is the cycle $q_1 \rightarrow q_9 \rightarrow q_8 \rightarrow q_1$ with:
Required time from q_1 to $q_9 = 10.96$ $[Unit\ of\ time]$.
Required time from q_9 to $q_8 = 19.08$ $[Unit\ of\ time]$.

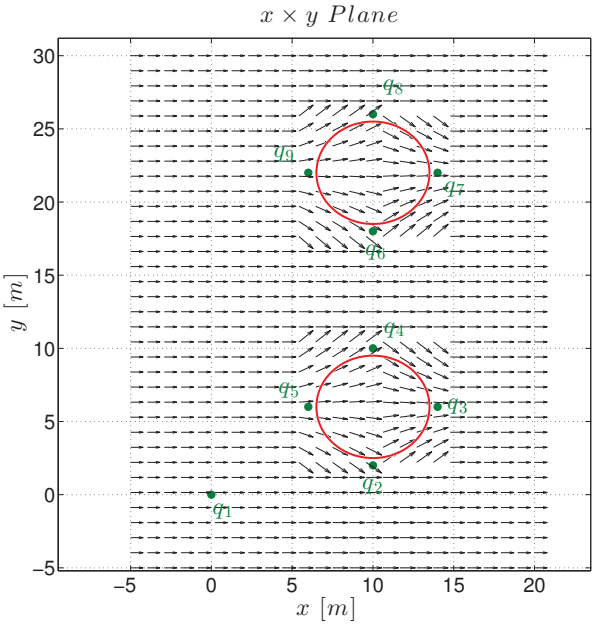

FIGURE 3.20: Simulation's environment.

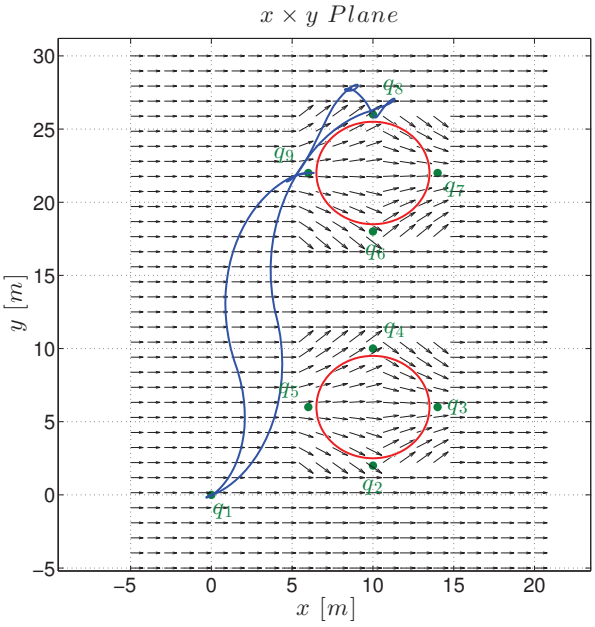

FIGURE 3.21: Flight number 1: $q_1 \rightarrow q_9 \rightarrow q_8 \rightarrow q_1$.

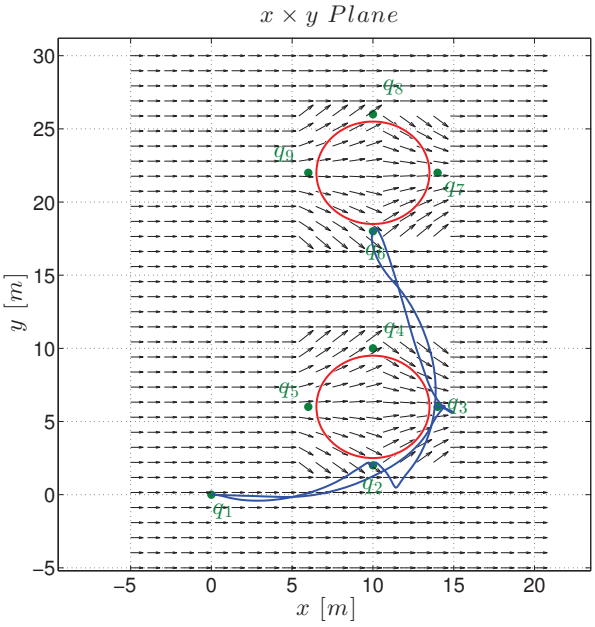

FIGURE 3.22: Flight number 2: $q_1 \rightarrow q_2 \rightarrow q_6 \rightarrow q_3 \rightarrow q_1$.

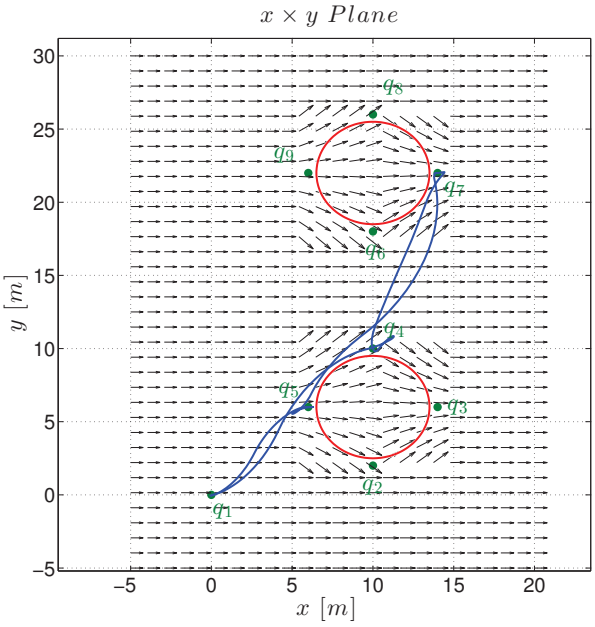

FIGURE 3.23: Flight number 3: $q_1 \rightarrow q_5 \rightarrow q_7 \rightarrow q_4 \rightarrow q_1$.

Required time from q_8 to $q_1 = 13.57$ [$Unit\ of\ time$].
Total flight time on the route $T = 53.62$ [$Unit\ of\ time$].

The second route, presented in Figure (3.22), is the cycle $q_1 \to q_2 \to q_6 \to q_3 \to q_1$ with:
Required time from q_1 to $q_2 = 6.42$ [$Unit\ of\ time$].
Required time from q_2 to $q_6 = 10.43$ [$Unit\ of\ time$].
Required time from q_6 to $q_3 = 8.72$ [$Unit\ of\ time$].
Required time from q_3 to $q_1 = 8.91$ [$Unit\ of\ time$].
Total flight time on the route $T = 49.49$ [$Unit\ of\ time$].

The third route, described in Figure (3.23), is the cycle $q_1 \to q_5 \to q_7 \to q_4 \to q_1$ with:
Required time from q_1 to $q_5 = 6.86$ [$Unit\ of\ time$].
Required time from q_5 to $q_7 = 12.97$ [$Unit\ of\ time$].
Required time from q_7 to $q_4 = 7.55$ [$Unit\ of\ time$].
Required time from q_4 to $q_1 = 10.00$ [$Unit\ of\ time$].
Total flight time on the route $T = 52.39$ [$Unit\ of\ time$].

Notice that the total time for each route respects the vehicle's capacity.

3.4 Conclusion

In this chapter, we have investigated the motion planning and trajectory generation question for small UAV-based structure inspection missions. First of all, the accessibility and controllability issues were considered. It has been proved that a kinematic aerial vehicle is strongly accessible under the condition that its velocity is not equal to zero and its flight path angle do not reach ninety degrees. Even-more, it has been shown that the vehicle could be completely controllable if the wind velocity is strictly less than the vehicle's velocity.

Next, the problem of trajectory generation was addressed. An analysis of the time-optimal trajectories of a kinematic aerial vehicle considering second-order constraints, with a piecewise constant wind and the presence of obstacles was presented. The minimum-time flight problem is formulated as a problem of the calculus of variations. This formulation is converted to nonlinear constrained optimization problems and the solution is obtained through a direct collocation approach. Geometric characterization of the candidate trajectories satisfying the necessary conditions for time optimality was also described. This characterization is used to develop a sub-optimal fast trajectory generation algorithm. Besides, an updated flight planning is presented using the receding horizon control as wind is considered to be piecewise constant.

Likewise, a strategy for trajectory planning was developed. The proposed method is based on operational research approaches, especially capacitated vehicle routing problem. In fact, the solution is a modified CVRP algorithm that includes the maximum flight time and loitering in optimization objective.

Chapter 4

Trajectory Tracking

Once a feasible and flyable trajectory is generated, it becomes necessary to move one step down into low-level control design, that means, to deal with trajectory tracking question. Such problem consists in stabilizing the state, or an output function of the state, to a desired reference value, possibly time-varying [123]. The control strategies proposed for this aim include two categories: traditional control methods and advanced control [118]. The classical control algorithms (like PID) have the advantage to be easily implemented and to provide reliable control performance. On the other hand, advanced control methods are mainly developed to improve its control performance in a complex and unstable flight environment.

4.1 Introduction

In order to make an unmanned aerial vehicle asymptotically track a desired reference trajectory, several control methods have been developed in literature such as backstepping control, robust control, model-prediction control and other intelligent control methods [17, 33].

In [167], a robust control method for an airship is introduced. The control strategy consists of an inner H_∞ controller for the dynamics and an outer Single Input Single Output Proportional or Proportional-Integral controller for the remaining states. Similarly different solutions are also proposed, namely one-loop-at-a-time PID [125] and Proportional-Integral control [54], sliding modes techniques [45, 171], vision-based [8, 64, 73, 95, 149, 150, 163], fuzzy logic [138] and fuzzy logic improved with genetic algorithms [137]. Other approaches include dynamic inversion for lateral and longitudinal control with desired dynamics given by a linear optimal compensator[124] and input-output linearization with a nonlinear controller [18, 170]. In the same way in [153], a

nonlinear control strategy based on extended linearization is proposed to stabilize the aerodynamic speed, altitude and heading of a small sized airship in cruise flight phase.

A popular technique used to control lighter than air vehicles is the backstepping methodology. In [14], the authors propose a time-varying controller, based on this approach, to stabilize both the orientation and the position of this aerial vehicle to a fixed value. In [13], a backstepping tracking feedback control for ascent and descent flight maneuvers is also introduced. In this work the desired trajectories were constrained to be trimming. In a similar way, in [139] and [11] the authors introduce a trajectory tracking controller for the airship based on this approach. Furthermore, a control strategy based on combination of backstepping and Exact Tracking Error Dynamics Passive Output Feedback (ETDPOE) is proposed in [129].

Artificial neural network is another approach currently applied to airship control designs. For example in [136], a neural network is applied to compensate the underlying model errors and to control velocity, pitch and yaw angles for an airship. Likewise in [86], a Particle Swarm Optimization algorithm and an updated neural network model to approximate a tri-turbofan airship are presented. On the other hand, robust controllers to stabilize airships have been also proposed. In [175], a robust adaptive controller using Lyapunov method and Matrosov theorem are presented. Similarly in [66], a model predictive control is introduced to stabilize the airship system taking into account inequality constraints on position, velocity and thrust. In addition in [65], the inverse optimal control approach was employed to design a trajectory tracking controller to stabilize an airship in ideal environment. The proposed method takes only the uncertainties into account without paying any attention to the potential perturbations applied on the vehicle such wind gusts.

Even though there are a lot of modern control techniques which can be adopted for unmanned aerial vehicle control, robust controllers remain interesting for real applications of these vehicles when they are exposed to unknown environment (like wind gusts). In this chapter, we focus on introducing a robust control strategy to make autonomous trajectory following of lighter than air/quad-rotors vehicles in the presence of wind gust. The control algorithm is based on Robust Control Lyapunov Function and Sontag's universal stabilizing feedback. This method has been successfully applied to the PVTOL aircraft and quad-rotors craft exposed to crosswind with a constant set point [127, 128], neverthless it has never been applied to trajectory tracking for 3DOF translational model of lighter than air vehicle and quad-rotors craft.

The rest of this chapter is organized as follows: The section 4.2 is dedicated to introduce the concept of the Robust Control Lyapunov Function. While, the section 4.4 presents a

nonlinear controller to stabilize kinematic airship/quad-rotors vehicles. The section 4.5 addresses the stabilization problem for a lighter than air vehicle tacking its dynamics, i.e. Equation (2.95), into consideration. Finally, the section 4.6 describes autopilot design strategy to a quad-rotors craft, based on its rigid body model, i.e. Equation (2.63).

4.2 Robust Control Lyapunov Function

Mathematical models of actual systems contain uncertainty terms representing the designer's lack of knowledge about parameter values and disturbances. These quantities and inaccuracies in mathematical modeling itself, contribute to performance degradation of feedback control system [176]. Self-tuning and adaptive control methods are based on estimating the unknown system's parameters and uncertainties through on-line identification algorithms. Then, this estimated parameters are used to provide a suitable adaptive globally stable controller. The main disadvantage of these approaches is their complexity.

On the other side, the deterministic control approaches provide a fixed nonlinear feedback control law which can be used directly to stabilize the system over a specified range of system variations and perturbations. One of the main methods used in deterministic control of uncertain systems is the *Lyapunov control design technique*.

In [119] Aleksandr Lyapunov proved that the stability of a dynamical system is guaranteed by the existence of a scalar valued function (called a *Lyapunov function*) of the system state which has a strict minimum at the equilibrium state and whose value decrease along every trajectory of the system. Since Lyapunov stability theory was developed for systems without inputs , it has been classically employed to test the stability of closed loop control systems for which the control inputs has been replaced through the predetermined feedback control. Anyway, an extension to the Lyapunov theory has been made by using a candidate Lyapunov functions in the feedback control designing procedure itself [6, 154]. The idea is to stabilize a control system by choosing a control law in order to make the Lyapunov derivative negative. Such objective led to the introduction of the *control Lyapunov function* (CLF). Therefore, considering the following control system

$$\dot{x} = f(x, u) \tag{4.1}$$

a control Lyapunov function (CLF) for a such system is a $\mathcal{C}^1$ *, positive definite, radially unbounded † function $\mathscr{L}(x)$ such that

$$x \neq 0 \quad \Rightarrow \quad \inf_{u \in \mathscr{U}} \nabla \mathscr{L}(x).f(x,u) < 0 \tag{4.2}$$

where $\mathscr{U}$ is a convex set of admissible values of the control variable u.

Freemann and Kokotović had generalized the CLF methodology to systems having both control and disturbance inputs. Their work resulted in introducing the ***Robust Control Lyapunov Function*** (RCLF). In addition, they proved that the existence of a RCLF is necessary and sufficient to guarantee the robust stabilizability [62].

Considering the following three finite-dimensional Euclidean spaces $\mathscr{X}$, $\mathscr{U}$, $\mathscr{W}$ to be the *state*, *control* and *disturbances* spaces respectively. Assuming a continuous function $f : \mathscr{X} \times \mathscr{U} \times \mathscr{W} \times \Re \to \mathscr{X}$, we define the nonlinear control system Σ as

$$\dot{x} = f(x,u,w,t) \tag{4.3}$$

where $x \in \mathscr{X}$ is the state variables, $u \in \mathscr{U}$ is the control inputs, $w \in \mathscr{W}$ is the disturbance input, and $t \in \Re$ is the time variable.

Definition 4.1. Suppose that φ_1 and φ_2 are class $\mathcal{K}_\infty$ ‡ functions. The set of all $\mathcal{C}^1$ functions $\mathscr{L} : \mathscr{X} \times \Re \to \Re_+$ such that

$$\varphi_1(||x||) \leq \mathscr{L}(x,t) \leq \varphi_2(||x||) \tag{4.4}$$

is called the set of all candidate Lyapunov functions for testing the robust stability of the system Σ, and denoted by $\mathscr{V}(\mathscr{X})$.

Definition 4.2. The set $\mathcal{P}(\mathscr{X}) \supset \mathscr{V}(\mathscr{X})$ is the set of all continuous functions $\varphi : \mathscr{X} \times \Re \to \Re_+$ such that there exist $\mathscr{F}_1, \mathscr{F}_2 \in \mathcal{K}$ ‡ satisfying

$$\mathscr{F}_1(||x||) \leq \varphi(x,t) \leq \mathscr{F}_2(||x||) \tag{4.5}$$

Definition 4.3. The Lyapunov derivative of $\mathscr{L} \in \mathscr{V}(\mathscr{X})$ is given by the following relationship

$$L_f \mathscr{L}(x,u,w,t) := \mathscr{L}_t(x,t) + \mathscr{L}_x(x,t).f(x,u,w,t) \tag{4.6}$$

where $\mathscr{L}_t$ and $\mathscr{L}_x$ denote the partial derivatives of $\mathscr{L}$ with respect to time (t) and states (x) consequently.

∗. A function $f(x)$ is said to be $\mathcal{C}^1$ if it is differentiable and its derivative is continuous.

†. A function $f(x) : \Re^n \to \Re$ is said to be radially unbounded if it satisfies the condition: $||x|| \to \infty \Rightarrow f(x) \to \infty$.

‡. A continuous function $f : [0,a[\to [0,\infty[$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $f(0) = 0$. It is said to belong to class $\mathcal{K}_\infty$ if $a = \infty$ and $f(r) \to \infty$ as $r \to \infty$.

Definition 4.4. Let $\mathscr{Y}$ to be a finite-dimensional Euclidean space presenting the output space, $y \in \mathscr{Y}$ to describe an output, $t \in \Re$ to present a time, and $c \in \Re_+$ to denote a constant. We define the set $Q(y, c, t)$ to be

$$Q(y, c, t) := \{x \in \mathscr{X} : y \in \mathscr{Y}(x, t) \text{ and } \mathscr{L}(x, t) \geq c\} \tag{4.7}$$

In other words, $Q(y, c, t)$ is the set of all states $x \in \mathscr{X}$ which lie outside the c-level set of $\mathscr{L}$ and which are consistent with the output y.

Based on the previous definitions, a function $\mathscr{L} \in \mathscr{V}(\mathscr{X})$ is said to be a ***Robust Control Lyapunov Function*** (RCLF) for the system Σ, Equation (4.3), when there exist $c_v \in \Re_+$ and $\varphi_v \in \mathcal{P}(\mathscr{X})$ such that

$$\inf_{u \in \mathscr{U}} \sup_{x \in Q} \sup_{w \in \mathscr{W}} [L_f \mathscr{L}(x, u, w, t) + \varphi_v(x, t)] < 0 \tag{4.8}$$

for all $y \in \mathscr{Y}$, all $t \in \Re$ and all $c > c_v$ [62].

Remark Supposing that Σ_1 is a time-invariant system with full-state feedback, i.e. $\mathscr{X} = \mathscr{Y}$ and $y(x) = \{x\}$ for all $x \in \mathscr{X}$. Then, Equation (4.3) becomes

$$\dot{x} = f(x, u, w) \tag{4.9}$$

Thus, the time-invariant function $\mathscr{L} \in \mathscr{V}(\mathscr{X})$ is said to be a RCLF for Σ_1 if and only if there exist $c_v \in \Re_+$ and a time-invariant function $\varphi_v \in \mathcal{P}(\mathscr{X})$ such that

$$\inf_{u \in \mathscr{U}} \sup_{w \in \mathscr{W}} [L_f \mathscr{L}(x, u, w) + \varphi_v(x)] < 0 \tag{4.10}$$

whenever $\mathscr{L}(x) > c_v$.

After defining a robust control Lyapunov function, the remaining task is to construct a feedback control inputs which guarantee the stability of the system. This task is the subject of the next section.

4.3 Inverse Optimality Design

Optimal feedback controllers are very interesting because, above ensuring the stability of the closed-loop system, they guarantee many desired characteristics through the associated cost functional which depends on the state and control variables. Therefore, the optimality property plays an important role in the selection of the control inputs among

the entire set of stabilizing control laws [62]. In order to benefit of such an advantage, it is required to solve the steady-state Hamilton-Jacobi-Issacs (HJI) equation, which is the optimality condition for the robust stabilization problem. The complexity of these controllers comes from the fact that solving the HJI equation is an extremely difficult task for a high order nonlinear system. Therefore, the robust optimal feedback stabilization problem can be summarized as follows: Given a general time-invariant nonlinear system and cost functional

$$\dot{x} = f(x, u, w), \qquad J = \int_0^\infty \mathscr{K}(x, y)\, dt \tag{4.11}$$

find the feedback control u which achieves asymptotic stability of the equilibrium $x = 0$, and minimizes the cost functional J. The steady-state HJI equation for this problem is written as

$$0 = \min_u \quad \max_w \quad [\mathscr{K}(x, u) + \nabla \mathscr{L}.f(x, u, w)] \tag{4.12}$$

where the *value function* $\mathscr{L}$ is the unknown. However, it is possible to avoid solving the steady-state HJI equation through many methods proposed in literature. One of these approaches is the *inverse optimality design* methodology. The idea is to find a meaningful cost functional such that the given RCLF is the corresponding value function, hence, the solution of the steady-state HJI equation is obtained indirectly which allows to compute the stabilizing optimal control law.

In [62] Freemann and Kokotović showed that every RCLF solves the steady-state HJI equation associated with a meaningful game. Moreover, they proposed an explicit mathematical relationship to generate an optimal feedback control law, called *pointwise min-norm*, which involves only the RCLF, the system equations and design parameters without solving the HJI equation nor to construct the cost functional. Hence, their approach is described as follows:

Assume that the system Σ_1 given in Equation (4.9) can be written as a nonlinear control system affine in controls and disturbances

$$\dot{x} = f_0(x) + f_1(x)u + f_2(x)w \tag{4.13}$$

where $x \in \mathscr{X}$ describes the state variables, $u \in \mathscr{U}$ denotes the control inputs, $w \in \mathscr{W}$ represents the disturbances, and $f_0(x)$ $f_1(x)$, $f_2(x)$ are continuous functions.

In addition, suppose that the disturbance is subjected to a constraint $\mathcal{W} : \mathscr{X} \rightsquigarrow \mathscr{W}$ which is continuous with nonempty compact [§] values, and convex in u [62]. Then, a disturbance w is said to be admissible when $w \in \mathcal{W}(x)$ for all $x \in \mathscr{X}$.

§. A set $\mathscr{A}$ is said to be compact if it is *closed* and *bounded*.

Given a Robust Control Lyapunov Function $\mathscr{L}$ for the system Σ_1, we define $D : \mathcal{X} \times \mathcal{U} \to \Re$ and $K : \mathcal{X} \rightsquigarrow \mathcal{U}$ by

$$D(x,u) := \max_{w \in \mathcal{W}(x)} [L_f \mathscr{L}(x,u,w) + \varphi_v(x)] \tag{4.14}$$

$$K(x) := \{u \in \mathscr{U}(x) : D(x,u) < 0\} \tag{4.15}$$

Then the above relationship implies that

$$D(x,u) = \varpi_0(x) + \varpi_1^T(x)u \tag{4.16}$$

$$K(x) = \left\{u \in \mathscr{U} : \varpi_0(x) + \varpi_1^T(x)u < 0\right\} \tag{4.17}$$

where,

$$\varpi_0(x) = \nabla\mathscr{L}(x).f_0(x) + \|\nabla\mathscr{L}(x).f_2(x)\| + \varphi_v(x) \tag{4.18}$$

$$\varpi_1(x) = [\nabla\mathscr{L}(x).f_1(x)]^T \tag{4.19}$$

and $\varphi_v(x) > 0$.

Remark that D is continuous and convex in u. Moreover K is lower semi continuous (lsc) [¶] with nonempty convex values on $\mathscr{L}^{-1}(c_v, \infty)$. A such set has a unique element of minimum norm which can be defined using the projection theorem described in [101].

Theorem 4.5. [The projection theorem] *Let $\mathcal{G} \subset \mathcal{H}$ be a Hilbert subspace [‖] and let $x \in \mathcal{H}$.*

Then,

1. *There exist a unique element $\hat{x} \in \mathcal{G}$ (called the **projection of** x **onto** $\mathcal{G}$) such that*

$$\|x - \hat{x}\| = \inf_{y \in \mathcal{G}} \|x - y\| \tag{4.20}$$

   where $\|\cdot\|$ is the norm generated by the inner product associated with $\mathcal{H}$.

2. *$\hat{x}$ is (uniquely) characterized by*

$$(x - \hat{x}) \in \mathcal{G}^{\perp} \tag{4.21}$$

   *with, $\mathcal{G}^{\perp}$ denotes the **orthogonal complement** to $\mathcal{G}$ in $\mathcal{H}$.*

Therefore the control law that stabilize the system (4.13) is:

$$u(x) := \operatorname{argmin}\{\|u\| : u \in K(x)\} \tag{4.22}$$

¶. A *set-valued function* $\mathcal{F} : \mathcal{X} \rightsquigarrow \mathcal{Z}$ is said to be ***lower semi continuous*** when $\forall x \in \mathcal{X}$, and for every open set $\mathcal{U} \subset \mathcal{Z}$ such that $x \in \mathcal{F}^{-1}(\mathcal{U})$ there exists a neighborhood $\mathcal{V}$ of x such that $\mathcal{V} \subset \mathcal{F}^{-1}(\mathcal{U})$.

‖. A Hilbert space is a pair $(\mathcal{H}, (\cdot,\cdot))$ such that: $\mathcal{H}$ is a vector space, $(\cdot,\cdot)$ is an inner product, and the normed space $(\mathcal{H}, \|\cdot\|)$ is complete, where $\|\cdot\|$ is the norm generated by $(\cdot,\cdot)$.

From Equation (4.16), Equation (4.17), and the projection theorem (Theorem 4.5), the Equation (4.22) becomes

$$u = \begin{cases} \frac{-\varpi_0(x)\varpi_1(x)}{\varpi_1^T(x)\varpi_1(x)} & ; \quad \varpi_0(x) > 0 \\ 0 & ; \quad \varpi_0(x) \leq 0 \end{cases} \tag{4.23}$$

Observe that the control law, Equation (4.23) depends on φ_v through the ϖ_0 function. Notice also that, there is never division by zero because the set $K(x)$ is nonempty for all $x \in \mathscr{L}^{-1}(c_v, \infty)$.

Remark: The function φ_v represents the desired negativity of the Lyapunov derivative, and it can be adjusted to achieve a tradeoff between the control effort and the rate of convergence of the state to zero.

4.4 Robust nonlinear controller for the kinematic aerial vehicles

In chapter 2, we modeled a kinematic aerial vehicle by the Equation (2.70) and Equation (2.100). Remark that these equations can be rewritten in the form of (4.13) as it is shown below [90, 91]

$$\dot{X} = \begin{bmatrix} V\cos\chi\cos\gamma \\ V\sin\chi\cos\gamma \\ V\sin\gamma \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} U + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} W \tag{4.24}$$

with $X = [x\ y\ z\ \gamma\ \chi\ V]^T$, $U = [u_1\ u_2\ u_3]^T$, and $W = [W_x\ W_y\ W_z]^T$.

Considering the following Lyapunov function

$$\mathscr{L}(X) = \frac{1}{2} X^T P X \tag{4.25}$$

where $P_{6\times 6}$ is a symmetric and positive definite matrix defined as the solution of the following algebraic Riccati equation

$$A^T X + XA - XBR^{-1}B^T X + Q = 0 \tag{4.26}$$

with, A and B denoting the state and input matrices of the linearized nominal system (about an operation point $X_{\mathcal{O}}$, $U_{\mathcal{O}}$) respectively. As f_1 and f_2 are constant matrices, then A and B are given by the following relationships

$$A = \left.\frac{\partial f_0(X)}{\partial X}\right|_{X_{\mathcal{O}}} \tag{4.27}$$

$$B = f_1 \tag{4.28}$$

Thus, from previous it follows

$$A = \begin{bmatrix} 0 & 0 & 0 & -V\cos\chi\sin\gamma & -V\sin\chi\cos\gamma & \cos\chi\cos\gamma \\ 0 & 0 & 0 & -V\sin\chi\sin\gamma & V\cos\chi\cos\gamma & \sin\chi\cos\gamma \\ 0 & 0 & 0 & V\cos\gamma & 0 & \sin\gamma \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.29a}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.29b}$$

Then , the derivative of the Lyapunov function becomes

$$\nabla\mathscr{L}(X) = \frac{1}{2}\frac{\partial[X^T P X]}{\partial X} = PX = \begin{bmatrix} \nabla_{\mathscr{L}_1} & \nabla_{\mathscr{L}_2} & \nabla_{\mathscr{L}_3} & \nabla_{\mathscr{L}_4} & \nabla_{\mathscr{L}_5} & \nabla_{\mathscr{L}_6} \end{bmatrix}^T$$

with

$$\nabla_{\mathscr{L}_i} = P_i X \quad ; \quad P_i \text{ is the } i-th \text{ row of } P.$$

In addition, from Equation (4.18) and Equation (4.19), $\varpi_0(X)$ and $\varpi_1(X)$ takes the following form

$$\varpi_0(X) = \nabla_{\mathscr{L}_1} V\cos\chi\cos\gamma + \nabla_{\mathscr{L}_2} V\sin\chi\cos\gamma + \nabla_{\mathscr{L}_3} V\sin\gamma + \varphi_v(X) + \sqrt{\nabla^2_{\mathscr{L}_1} + \nabla^2_{\mathscr{L}_2} + \nabla^2_{\mathscr{L}_3}}$$

and

$$\varpi_1(X) = \begin{bmatrix} \nabla_{\mathscr{L}_4} \\ \nabla_{\mathscr{L}_5} \\ \nabla_{\mathscr{L}_6} \end{bmatrix}$$

where $\varphi_v(X) = X^T M X$ and $M_{6\times 6}$ is a diagonal positive matrix, such that,

$$\varphi_v(X) = M_{11}x^2 + M_{22}y^2 + M_{33}z^2 + +M_{44}\gamma^2 + M_{55}\chi^2 + M_{66}V^2$$

Consequently, the robust control laws yields

$$\begin{aligned}
u_1 &= \begin{cases} \frac{-\varpi_0(X).\nabla_{\mathscr{L}_4}}{\nabla^2_{\mathscr{L}_4}+\nabla^2_{\mathscr{L}_5}+\nabla^2_{\mathscr{L}_6}} & ; \quad \varpi_0(X) > 0 \\ 0 & ; \quad \varpi_0(X) \leq 0 \end{cases} \\
u_2 &= \begin{cases} \frac{-\varpi_0(X).\nabla_{\mathscr{L}_5}}{\nabla^2_{\mathscr{L}_4}+\nabla^2_{\mathscr{L}_5}+\nabla^2_{\mathscr{L}_6}} & ; \quad \varpi_0(X) > 0 \\ 0 & ; \quad \varpi_0(X) \leq 0 \end{cases} \\
u_3 &= \begin{cases} \frac{-\varpi_0(X).\nabla_{\mathscr{L}_6}}{\nabla^2_{\mathscr{L}_4}+\nabla^2_{\mathscr{L}_5}+\nabla^2_{\mathscr{L}_6}} & ; \quad \varpi_0(X) > 0 \\ 0 & ; \quad \varpi_0(X) \leq 0 \end{cases}
\end{aligned} \tag{4.30}$$

4.4.1 Simulation Results

In order to validate the performance of the closed-loop system, two scenarios for simulations were carried out. The first one presents a trajectory with a line form while the second one a trajectory with a curved form. The wind employed in these simulations consists of an average value on each axis, and two gusts. The first begins at $time = 400[sec]$ while the second take place at $time = 700[sec]$. This wind is introduced in the Figure (4.1). On the other hand, the constraints on flight path angle, velocity and control variables are provided in Table 4.1.

Moreover, solving algebraic Riccati Equation (4.26) requires the identification of the weight matrices Q and R. The matrix Q has a dimension of $n \times n$; $n = 6$ and it must be symmetric, positive semi-definite. Beside, the matrix R is a symmetric positive definite and has a dimension of $m \times m$; $m = 3$. The selection of the elements of Q and R matrices play an important role in the closed loop response. Choosing Q to be large means that the error variables ($e = X - X_{ref}$) must be smaller which implies a large control efforts. On the contrary, taking large values for R matrix means that the control inputs must be smaller resulting in large values of error variables. The selected values for Q and R are

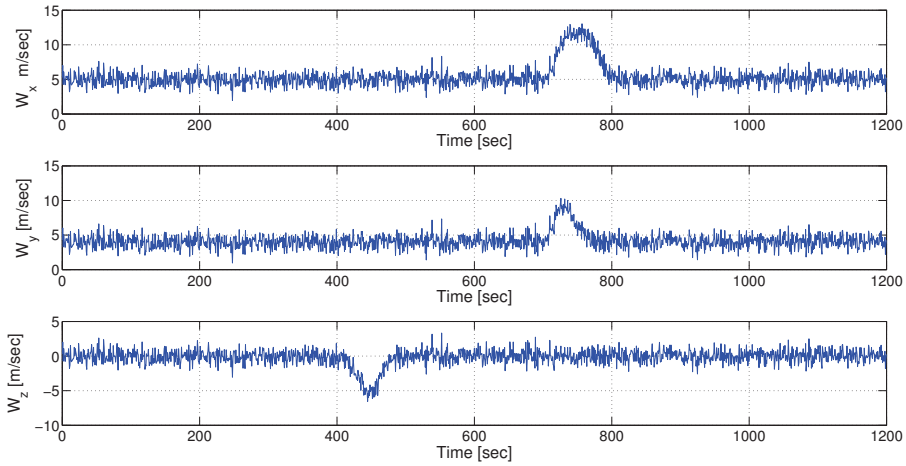

FIGURE 4.1: Wind gust applied to the aerial vehicle.

TABLE 4.1: Restrictions on state and control variables

Minimum	Variable	Maximum	Units
- 45	γ	45	deg
0.1	V	20	m/sec
- 10	$\dot{\gamma}$	10	deg/sec
- 20	$\dot{\chi}$	20	deg/sec
- 1.25	$\dot{V}$	1.25	m/sec^2

$$Q = \begin{bmatrix} 100 & 0 & 0 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10^4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix} \tag{4.31}$$

$$R = 10^4 * \begin{bmatrix} 1/u_{1max}^2 & 0 & 0 \\ 0 & 1/u_{2max}^2 & 0 \\ 0 & 0 & 1/u_{3max}^2 \end{bmatrix} \tag{4.32}$$

The M matrix is chosen to be symmetric positive semi-definite. In addition (like Q and

R matrices) it is adjusted to achieve a tradeoff between the control effort and the rate of convergence of the error variables to zero. In general, smaller values of M leads to smaller control magnitudes and slower convergence. The selected values are

$$M = \begin{bmatrix} 0.1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.001 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^6 \end{bmatrix} \tag{4.33}$$

4.4.1.1 Trajectory with a line form

Fixing the orientation of the vehicle (flight path and heading angles) to a constant value leads to a trajectory with a line form. Hence, our objective in this section is to stabilize the aerial vehicle around the desired orientation and velocity (which will be also constant) without forgetting to maintain the spatial motion of the vehicle at the desired one.

The initial position of the vehicle is $x_0 = y_0 = z_0 = 0$ while the initial and the desired orientation and velocity values are shown in Table 4.2. Remark that the velocity of the first and the second gusts is up to 46 % and 120 % of the aerial vehicle velocity respectively.

TABLE 4.2: Initial and desired orientation and velocity values

	$\gamma\ [Deg]$	$\chi\ [Deg]$	$V\ [m/sec]$
Initial value	0	0	5
Desired value	15	60	13

Basing on Q, R, M, the desired values of γ, χ, V and from the Equation (4.26), we have

$$P = \begin{bmatrix} 260 & 80 & 20 & -560 & -2480 & 390 \\ 80 & 350 & 40 & -960 & 1430 & 680 \\ 20 & 40 & 1700 & 17770 & 0 & 490 \\ -560 & -960 & 17770 & 384850 & 0 & 4170 \\ -2480 & 1430 & 0 & 0 & 76850 & 0 \\ 390 & 680 & 490 & 4170 & 0 & 3320 \end{bmatrix} \tag{4.34}$$

The response of the system to the proposed control strategy is illustrated in Figure (4.2) - Figure (4.9). The position errors along x, y and z axes are presented in the Figure (4.2), Figure (4.3) and Figure (4.4) respectively. Also, the time profile of the vehicle's orientation (presented by its flight path and heading angles) and its velocity are shown in Figure (4.5), Figure (4.6) and Figure (4.7). A three dimensional view of the spatial trajectory is displayed in the Figure (4.9). In these figures, the red line represents the desired signal while the blue line represents the system response. Additionally, the control inputs are depicted in Figure (4.8).

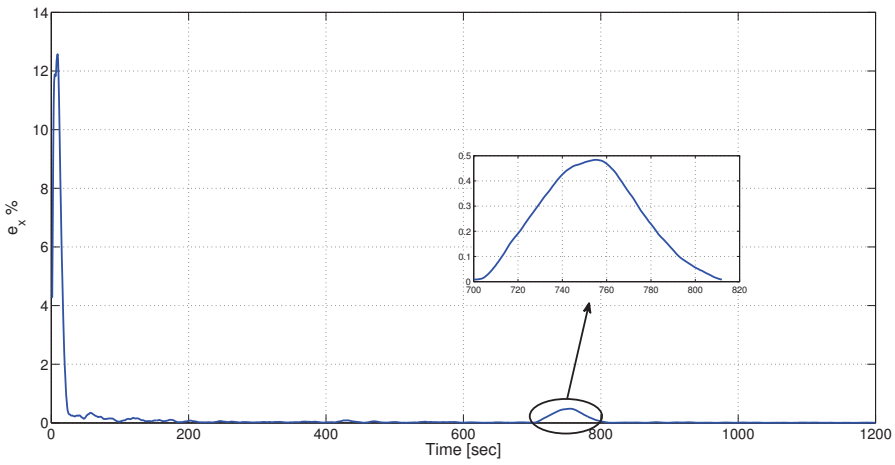

FIGURE 4.2: Relative error on x position (trajectory with a line form).

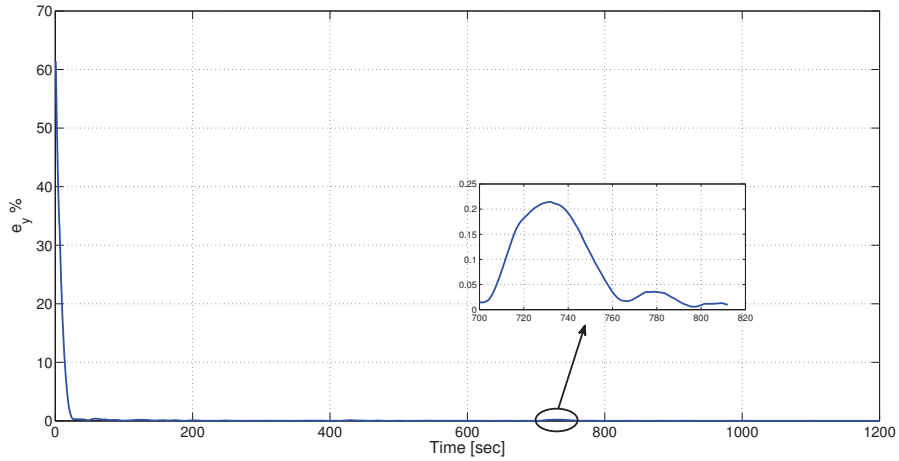

FIGURE 4.3: Relative error on y position (trajectory with a line form).

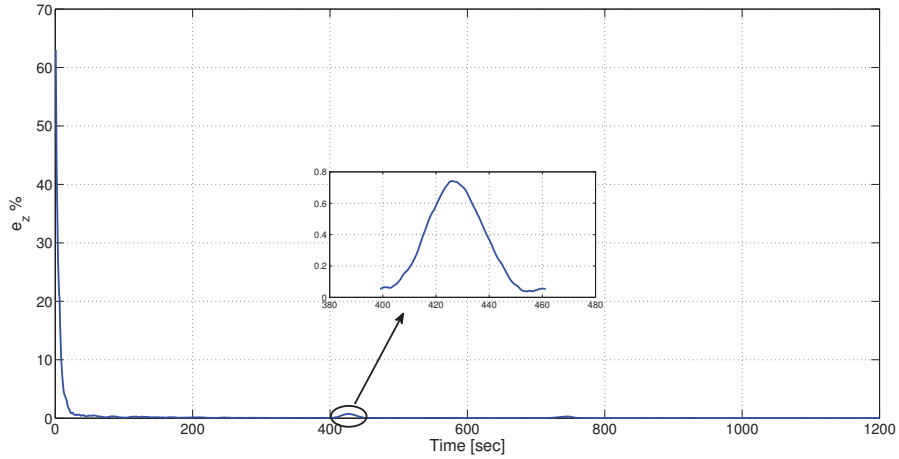

FIGURE 4.4: Relative error on z position (trajectory with a line form).

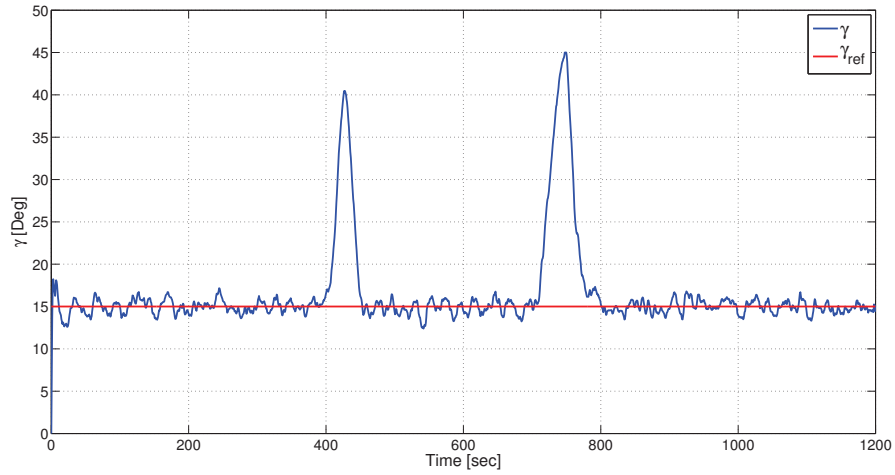

FIGURE 4.5: Flight Path Angle response in closed-loop (trajectory with a line form).

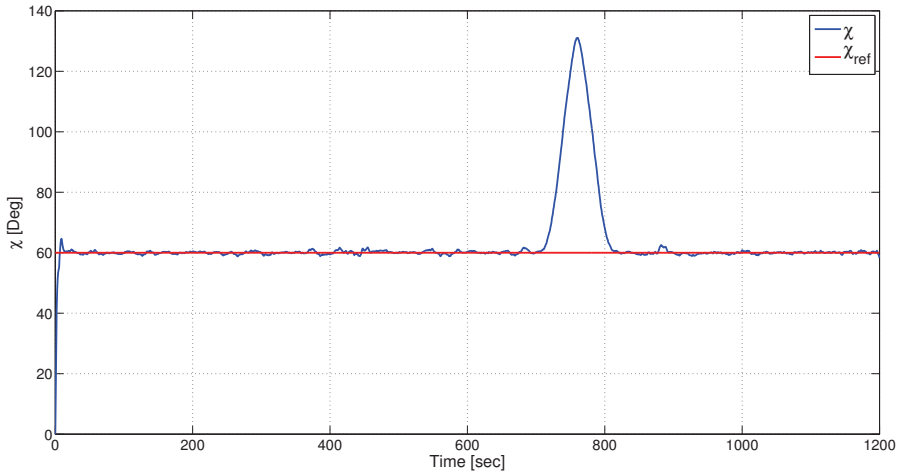

FIGURE 4.6: Heading Angle response in the closed-loop (trajectory with a line form).

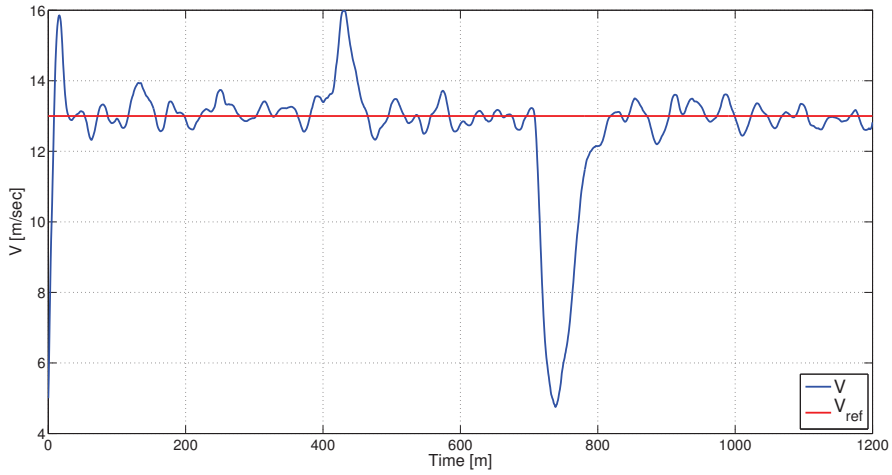

FIGURE 4.7: Velocity response in the closed-loop (trajectory with a line form).

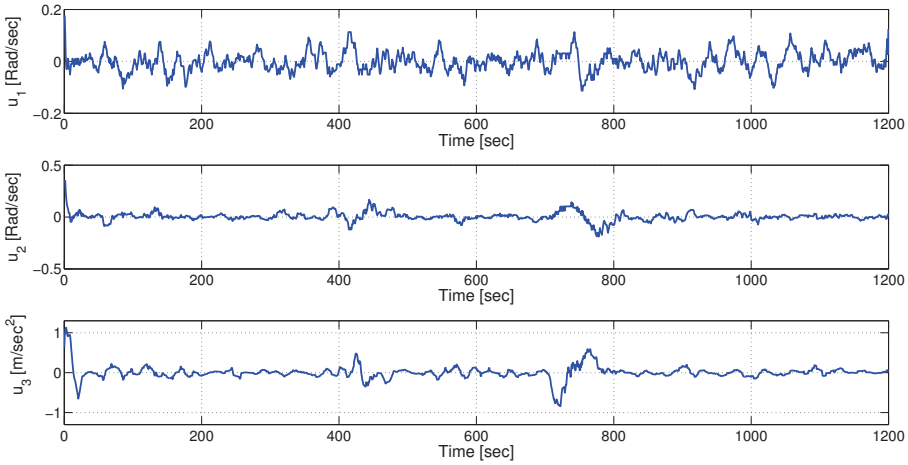

FIGURE 4.8: Control inputs responses u_1, u_2 and u_3 (trajectory with a line form).

The previous figures reveal the system reaction facing unknown perturbations. For the first wind gust, the control inputs increase the velocity value to 16$[m/sec]$ and the flight path angle to 40$[deg]$. Observe that as the first wind burst affects only the z axis, then the system will not need to change heading angle. Next, when the second wind gust appear, the control inputs increase the flight path angle to 45$[deg]$ and the heading angle to 130$[deg]$, while the vehicle's velocity is decreased to 5$[m/sec]$. Further, from Figure (4.2), Figure (4.3) and Figure (4.4) remark that the largest relative error on x, y and z positions in the presence of disturbances (wind gusts) remains reasonable (about 0.5%, 0.22% and 0.75% respectively). Thus, the closed-loop system has a good performance even in the presence of unknown wind gusts.

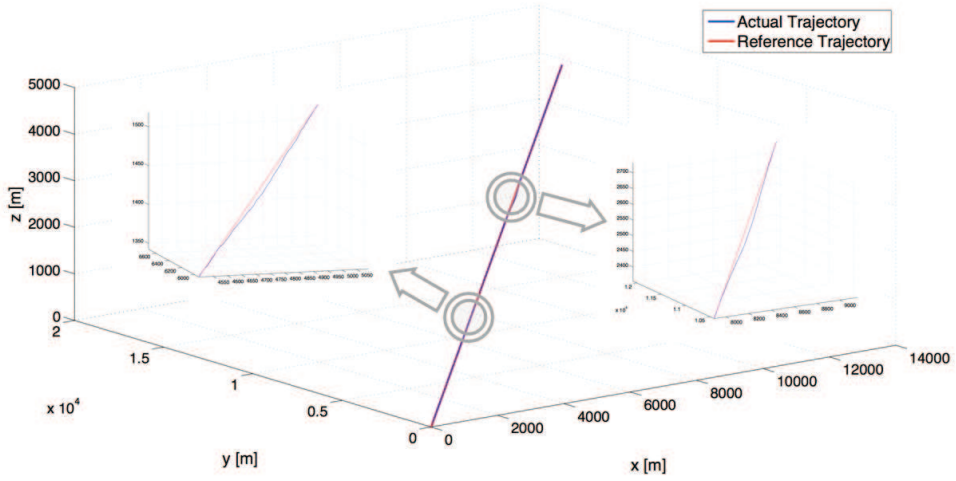

FIGURE 4.9: Three dimensional view of the trajectory.

4.4.1.2 Trajectory with a curved form

By definition, a curve is a line which gradually deviates from being straight for some or all of its length. This deviation implies that the orientation and the velocity of the aerial vehicle must vary with time. Thus, our objective in this section is to stabilize the aerial vehicle around the desired trajectory (spatial motion, orientation and velocity) which is also time variant.

The proposed simulation scenario consists of two phases. In the first one, which lasts 600 seconds, the aircraft starts at the position ($x_0 = 0$, $y_0 = 0$, $z_0 = 0$) with an orientation of ($\gamma_0 = 0$, $\chi_0 = 0$) and a velocity of $V_0 = 7[m/sec]$. Then, it performs a turning flight of 360 degrees (heading angle) and in the same time it climbs with an angle of 10 degrees (flight path angle) and maintains the velocity at V_0. Once the first phase is achieved, the second one begins and lasts from $t = 600[sec]$ to $t_f = 1200[sec]$. In this stage, the aircraft flies in a straight line at a constant altitude and a velocity of 13 [m/sec].

In order to maintain the stability of the aerial vehicle along the trajectory, four operating point had been chosen. These points are presented in Table 4.3. Thus, the Lyapunov function (4.25) and consequently the control laws will change four times during the trajectory depending on the actual operating point. The selected P for each operating point is

$$P_1 = \begin{bmatrix} 210 & 0 & 0 & 0 & -2860 & 0 \\ 0 & 400 & 40 & -620 & 0 & 800 \\ 0 & 40 & 9440 & 56960 & 0 & 860 \\ 0 & -620 & 56960 & 690750 & 0 & 7070 \\ -2860 & 0 & 0 & 0 & 77590 & 0 \\ 0 & 800 & 860 & 7070 & 0 & 3320 \end{bmatrix} \tag{4.35}$$

$$P_2 = \begin{bmatrix} 400 & 0 & 40 & 620 & 0 & 800 \\ 0 & 210 & 0 & 0 & 2860 & 0 \\ -40 & 0 & 9440 & 56960 & 0 & 860 \\ -620 & 0 & 56960 & 690750 & 0 & 7070 \\ 0 & -2860 & 0 & 0 & 77590 & 0 \\ 800 & 0 & 860 & 7070 & 0 & 3320 \end{bmatrix} \tag{4.36}$$

$$P_3 = \begin{bmatrix} 210 & 0 & 0 & 0 & 2860 & 0 \\ 0 & 400 & -40 & 620 & 0 & -800 \\ 0 & -40 & 9440 & 56960 & 0 & 860 \\ 0 & 620 & 56960 & 690750 & 0 & 7070 \\ 2860 & 0 & 0 & 0 & 77590 & 0 \\ 0 & -800 & 860 & 7070 & 0 & 3320 \end{bmatrix} \tag{4.37}$$

$$P_4 = \begin{bmatrix} 400 & 0 & 40 & -620 & 0 & 800 \\ 0 & 210 & 0 & 0 & 2860 & 0 \\ 40 & 0 & 9440 & 56960 & 0 & 860 \\ -620 & 0 & 56960 & 690750 & 0 & 7070 \\ 0 & 2860 & 0 & 0 & 77590 & 0 \\ 800 & 0 & 860 & 7070 & 0 & 3320 \end{bmatrix} \tag{4.38}$$

Table 4.3: The four selected operating points

	$\gamma\ [Deg]$	$\chi\ [Deg]$	$V\ [m/sec]$
OP_1	10	90	13
OP_2	10	180	13
OP_3	10	270	13
OP_4	10	360	13

The response of the system to the proposed control strategy is illustrated in Figure (4.10) - Figure (4.17). The position errors along x, y and z axes are presented in the Figure (4.10), Figure (4.11) and Figure (4.12) respectively. Also, the time profile of the vehicle's orientation (presented by its flight path and heading angles) and its velocity are shown in Figure (4.13), Figure (4.14) and Figure (4.15). A three dimensional view of the spatial trajectory is displayed in the Figure (4.17). In these figures, the red line represents the desired signal while the blue line represents the system response. Additionally, the control inputs are depicted in Figure (4.16).

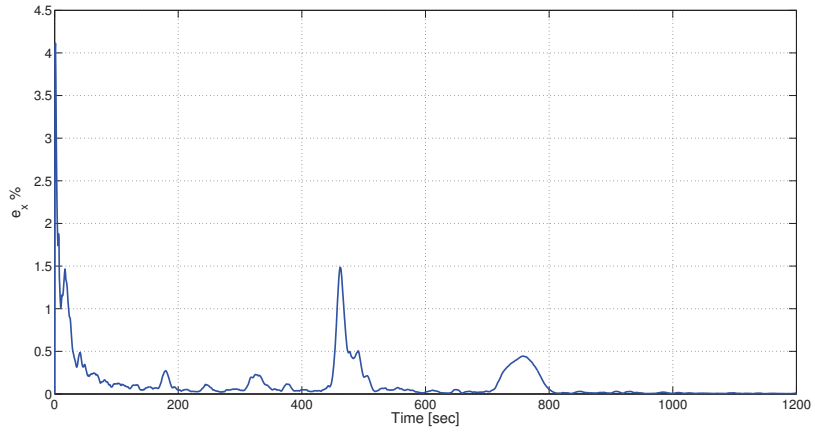

FIGURE 4.10: Relative error on x position (trajectory with a curve form).

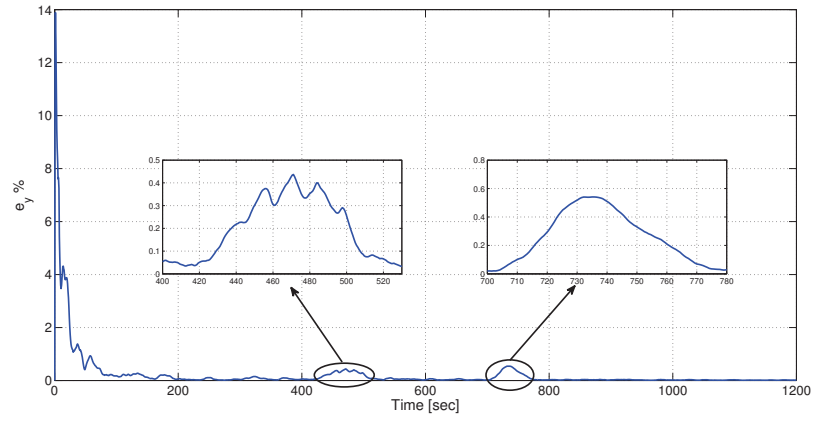

FIGURE 4.11: Relative error on y position (trajectory with a curve form).

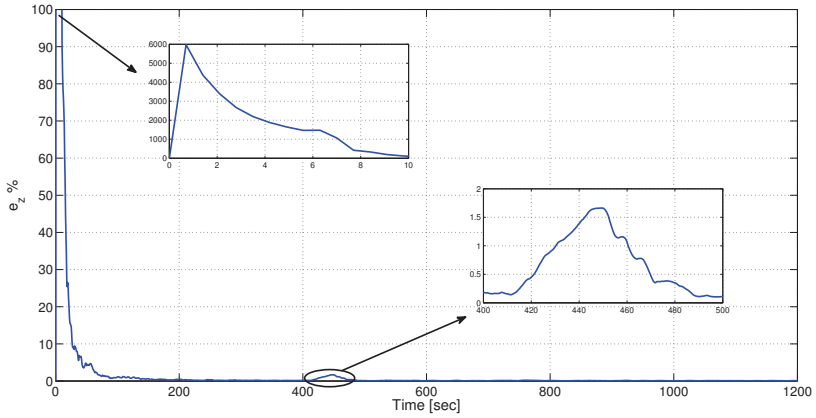

FIGURE 4.12: Relative error on z position (trajectory with a curve form).

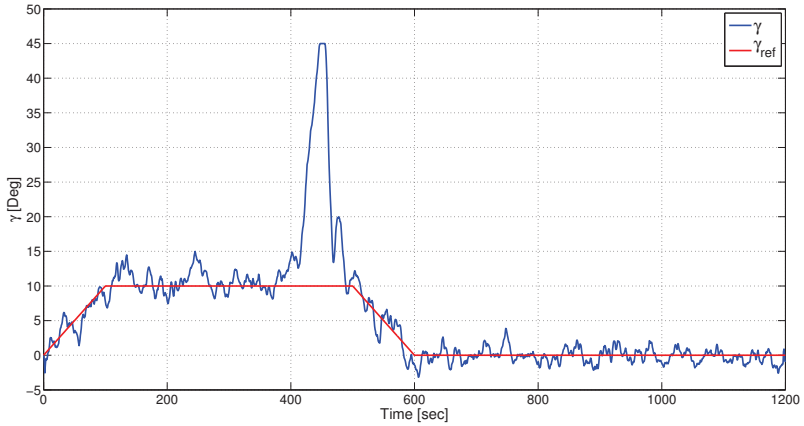

FIGURE 4.13: Flight Path Angle response in closed-loop (trajectory with a curve form).

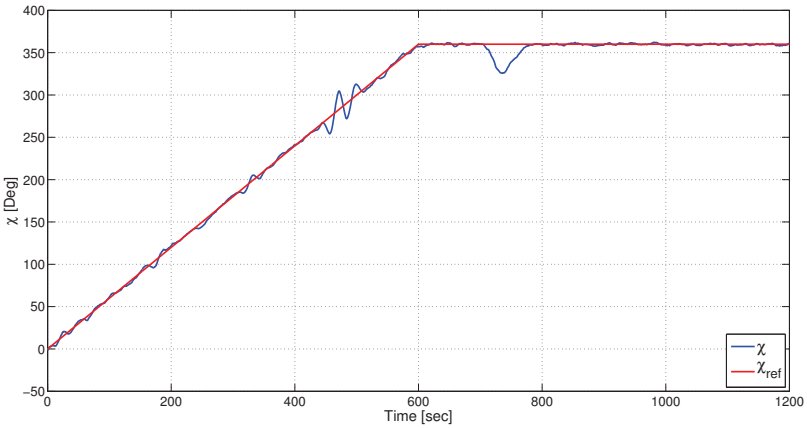

FIGURE 4.14: Heading Angle response in the closed-loop (trajectory with a curve form).

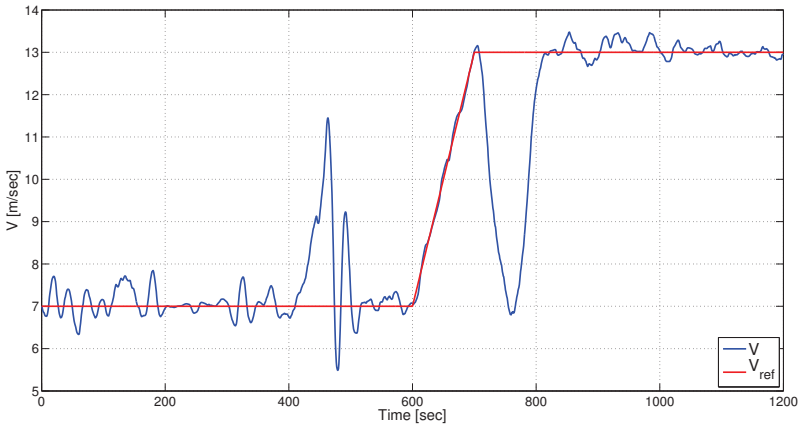

FIGURE 4.15: Velocity response in the closed-loop (trajectory with a curve form).

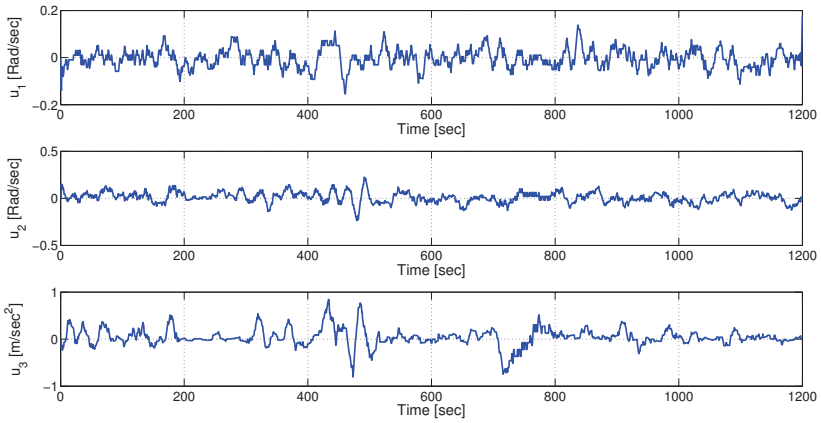

FIGURE 4.16: Control inputs responses u_1, u_2 and u_3 (trajectory with a curve form).

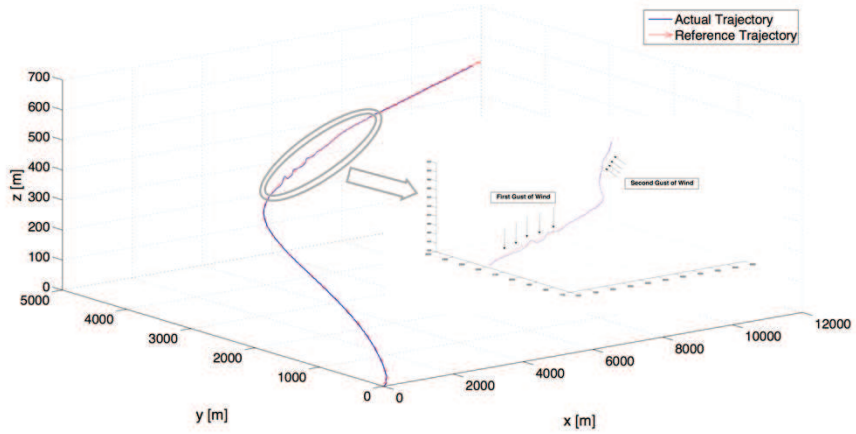

FIGURE 4.17: Three dimensional view of the trajectory.

As for the trajectory with a line form, the Figure (4.10) - Figure (4.17) reveal the system reaction facing unknown perturbations. For the first wind gust, the control inputs increase the velocity value to 11.6$[m/sec]$ and the flight path angle to 45$[deg]$. Observe that, in contrast to the case of a trajectory with a line form, the heading angle here is disturbed. Hence, control efforts is exerted to ensure the stability of χ. Next, when the second wind gust appear, the control inputs decrease the heading angle to 320$[deg]$ and the velocity to 7$[m/sec]$, while the flight path angle does not change because the aircraft flies at a constant altitude, *i.e.* γ is a constant, and the second burst affects only x and y axes. Further, from Figure (4.2), Figure (4.3) and Figure (4.4) remark that the largest relative error on x, y and z positions in the presence of disturbances (wind gusts) remains reasonable (about 1.5%, 0.55% and 1.65% respectively). Thus, the closed-loop system has a good performance even in the presence of unknown wind gusts.

4.5 Robust nonlinear controller for lighter than air vehicle

Translational motion of airships is characterized by six nonlinear differential equations (three for kinematics and three for dynamics) detailed in chapter (2). This modeling often employs angle of attack α, bank angle σ and thrust T as three control variables, which yields the model to be non affine in control inputs.

To overcome this obstacle, the nonlinear state equation is transformed into an augmented state space model in which the new control appears in a linear fashion [115, 152]. For this purpose, the control variables are added to the state vector and regarded as extra state variables, while its derivatives are used as virtual control inputs. So, the new state vector becomes $[x\ y\ z\ \gamma\ \chi\ V\ T\ \sigma\ \alpha]$. Whilst, the virtual control inputs turn into $[\dot{T}\ \dot{\sigma}\ \dot{\alpha}]$ [89].

From the previous description and Equation (2.70 and Equation 2.95), translational motion of a lighter than air vehicle can be mathematically represented by the following nonlinear differential equations affine in control and disturbances inputs:

$$\dot{X} = f_0 + f_1 U + f_2 W \tag{4.39}$$

where,

$$X = \left[\underbrace{x \quad y \quad z}_{X_1} \quad \underbrace{\gamma \quad \chi \quad V}_{X_2} \quad \underbrace{T \quad \sigma \quad \alpha}_{X_3} \right]^T \tag{4.40}$$

$$U = \dot{X}_3 = \begin{bmatrix} \dot{T} & \dot{\sigma} & \dot{\alpha} \end{bmatrix}^T \tag{4.41}$$

$$W = \left[\underbrace{W_x \quad W_y \quad W_z}_{W_1} \quad \underbrace{\dot{W}_x \quad \dot{W}_y \quad \dot{W}_z}_{W_2} \right]^T \tag{4.42}$$

$$f_0 = \begin{bmatrix} \Xi_a(X_2) & \Xi_b(X_2, X_3) & 0_{3\times 1} \end{bmatrix}^T \tag{4.43}$$

$$f_1 = \begin{bmatrix} 0_{3\times 3} & 0_{3\times 3} & I_{3\times 3} \end{bmatrix}^T \tag{4.44}$$

$$f_2 = \begin{bmatrix} I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & B_w \\ 0_{3\times 3} & 0_{3\times 3} \end{bmatrix} \tag{4.45}$$

with,

$$\Xi_a(X_2) = \begin{bmatrix} V\cos\chi\cos\gamma \\ V\sin\chi\cos\gamma \\ V\sin\gamma \end{bmatrix} \tag{4.46}$$

Likewise

$$\Xi_b\left(X_2, X_3\right) = \begin{bmatrix} \frac{(T\sin\alpha+L)\cos\sigma-\cos\gamma(B-mg)}{(m+m_{33})V} \\ \frac{(T\sin\alpha+L)\sin\sigma}{(m+m_{22})V\cos\gamma} \\ \frac{T\cos\alpha-D-\sin\gamma(B-mg)}{m+m_{11}} \end{bmatrix} \tag{4.47}$$

Indeed,

$$B_w = \begin{bmatrix} \frac{m\cos\chi\sin\gamma}{(m+m_{33})V} & \frac{m\sin\chi\sin\gamma}{(m+m_{33})V} & \frac{m\cos\gamma}{(m+m_{33})V} \\ \frac{m\sin\chi}{(m+m_{22})V\cos\gamma} & \frac{-m\cos\chi}{(m+m_{22})V\cos\gamma} & 0 \\ -\frac{m\cos\gamma\cos\chi}{m+m_{11}} & -\frac{m\sin\chi\cos\gamma}{m+m_{11}} & \frac{m\sin\gamma}{m+m_{11}} \end{bmatrix} \tag{4.48}$$

Considering the following Lyapunov function

$$\mathscr{L}\left(X\right) = \frac{1}{2}X^T P X \tag{4.49}$$

where $P_{9\times 9}$ is the solution of algebraic Riccati equation (4.26). Thus, the derivative of Lyapunov function is given by

$$\nabla\mathscr{L}\left(X\right) = \frac{1}{2}\frac{\partial\left[X^T P X\right]}{\partial X} = PX = \begin{bmatrix} \nabla_{\mathscr{L}_1} & \nabla_{\mathscr{L}_2} & \nabla_{\mathscr{L}_3} \end{bmatrix}^T \tag{4.50}$$

with $dim\left(\nabla_{\mathscr{L}_i}\right) = 3$; $i = 1, 2, 3$. Then, $\varpi_0\left(X\right)$ and $\varpi_1\left(X\right)$ defined in equations (4.18 and 4.19) respectively can be written as

$$\varpi_0\left(X\right) = \nabla\mathscr{L}\cdot f_0\left(X\right) + \| \begin{bmatrix} \nabla_{\mathscr{L}_1} & \nabla_{\mathscr{L}_2}\cdot B_w \end{bmatrix} \| + \varphi_v\left(X\right) \tag{4.51}$$

$$\varpi_1\left(X\right) = \nabla_{\mathscr{L}_3} \tag{4.52}$$

with $\varphi_v\left(X\right) = X^T M X$ and $M_{9\times 9}$ defines a diagonal positive matrix, which leads to

$$\varphi_v\left(X\right) = M_{11}x^2 + M_{22}y^2 + M_{33}z^2 + M_{44}\gamma^2 + M_{55}\chi^2 + M_{66}V^2 + M_{77}T^2 + M_{88}\sigma^2 + M_{99}\alpha^2 \tag{4.53}$$

Hence, the control inputs take the following form

For $\varpi\left(X\right) > 0$

$$u_1 = \dot{T} = \frac{-\varpi_0\left(X\right)\cdot\nabla_{\mathscr{L}_3}(1)}{\nabla^2_{\mathscr{L}_3}(1) + \nabla^2_{\mathscr{L}_3}(2) + \nabla^2_{\mathscr{L}_3}(3)} \tag{4.54a}$$

$$\tag{4.54b}$$

$$u_2 = \dot{\sigma} = \frac{-\varpi_0\left(X\right)\cdot\nabla_{\mathscr{L}_3}(2)}{\nabla^2_{\mathscr{L}_3}(1) + \nabla^2_{\mathscr{L}_3}(2) + \nabla^2_{\mathscr{L}_3}(3)} \tag{4.54c}$$

$$\tag{4.54d}$$

$$u_3 = \dot{\alpha} = \frac{-\varpi_0\left(X\right)\cdot\nabla_{\mathscr{L}_3}(3)}{\nabla^2_{\mathscr{L}_3}(1) + \nabla^2_{\mathscr{L}_3}(2) + \nabla^2_{\mathscr{L}_3}(3)} \tag{4.54e}$$

whilst, for $\varpi_0\left(X\right) \leq 0$, $u_1 = u_2 = u_3 = 0$.

Remark: Solving algebraic Riccati equation (4.26) requires the linearized nominal system which is given by the equation

$$\dot{X} = AX + BU \tag{4.55}$$

where,

$$A = \begin{bmatrix} 0_{3\times3} & A_a & 0_{3\times3} \\ 0_{3\times3} & A_b & A_c \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{4.56}$$

$$B = \begin{bmatrix} 0_{3\times3} \\ 0_{3\times3} \\ I_{3\times3} \end{bmatrix} \tag{4.57}$$

with,

$$A_a = \begin{bmatrix} -V\cos\chi\cos\gamma & -V\sin\chi\cos\gamma & \cos\chi\cos\gamma \\ -V\sin\chi\sin\gamma & V\cos\chi\cos\gamma & \sin\chi\cos\gamma \\ V\cos\gamma & 0 & \sin\gamma \end{bmatrix} \tag{4.58}$$

likewise,

$$A_b = \begin{bmatrix} \frac{\sin\gamma(B-mg)}{(m+m_{33})V} & 0 & \frac{L_v\cos\sigma}{(m+m_{33})} - \frac{\cos\sigma(T\sin\alpha+L)-\cos\gamma(B-mg)}{(m+m_{33})V^2} \\ \frac{\sin\gamma\sin\sigma(T\sin\alpha+L)}{(m+m_{22})V\cos^2\gamma} & 0 & \frac{L_v\sin\sigma}{(m+m_{22})V\cos\sigma} - \frac{\sin\sigma(T\sin\alpha+L)}{(m+m_{22})V^2\cos\gamma} \\ \frac{-\cos\gamma(B-mg)}{m+m_{11}} & 0 & \frac{-D_v}{m+m_{11}} \end{bmatrix} \tag{4.59}$$

Indeed,

$$A_c = \begin{bmatrix} \frac{\sin\alpha\cos\sigma}{(m+m_{33})V} & \frac{-\sin\sigma(T\sin\alpha+L)}{(m+m_{33})V} & \frac{L_\alpha\cos\sigma+T\cos\alpha}{(m+m_{33})V} \\ \frac{\sin\alpha\sin\sigma}{(m+m_{22})V\cos\gamma} & \frac{\cos\sigma(T\sin\alpha+L)}{(m+m_{22})V\cos\gamma} & \frac{L_\alpha\sin\sigma+T\cos\alpha}{(m+m_{22})V\cos\gamma} \\ \frac{\cos\alpha}{m+m_{11}} & 0 & \frac{-D_\alpha+T\sin\alpha}{m+m_{11}} \end{bmatrix} \tag{4.60}$$

4.5.1 Simulation Results

The simulation scenario proposed in this section consists of an initial configuration, a final one and three way-configurations. Hence, the trajectory is divided into four segments. The first sector presents a maneuver in which the aerial vehicle makes a left turn with ascent to reach the first way-configuration q_1. Next, in the second section, the airship flies in a climbing straight line to attain q_2. Then, it performs a right turn with descent until it arrives to q_3. Finally, in the forth segment, the lighter than air vehicle moves in a straight line with a constant altitude to its final configuration. These configurations are presented in Table 4.4.

TABLE 4.4: Configurations to be visited during the traveled trajectory

	$x\ [m]$	$y\ [m]$	$z\ [m]$	$\gamma\ [deg]$	$\chi\ [deg]$	$V\ [m/sec]$
q_0	0	0	0	0	0	10
q_1	1800	750	100	5	45	10
q_2	4620	3560	445	5	45	10
q_3	6400	4312	533	0	0	10
q_f	13406	4312	533	0	0	10

Besides, along the previous trajectory, the aerial vehicle is exposed to two wind gusts introduced in Figure (4.18). The first blast begin at $t = 330[sec]$, while the second one take place at $t = 800[sec]$. Their velocity is up to 70 % and 122 % of the vehicle velocity respectively.

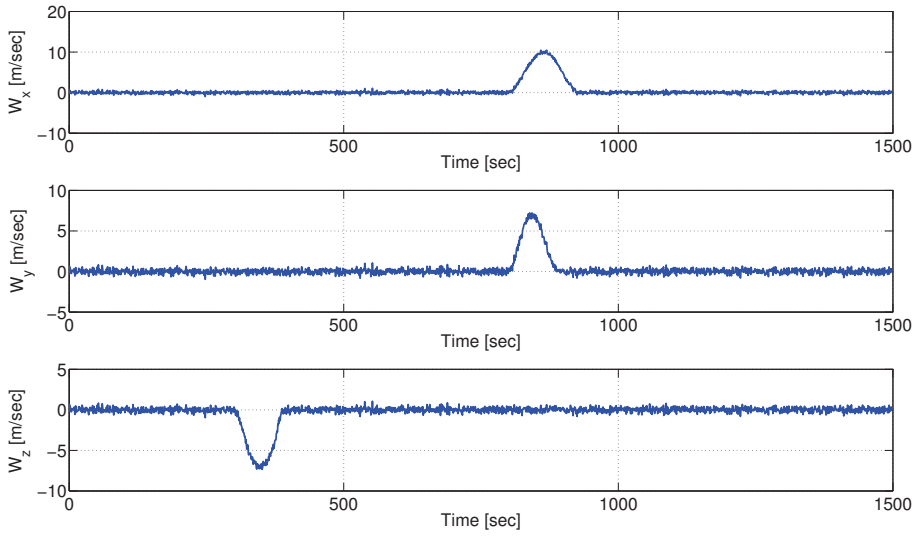

FIGURE 4.18: Wind gusts applied to the airship.

Also, the vehicle is subjected to different limitations and restrictions due to its actuators and structure. These constraints are provided in table (4.5). On the other hand, table (4.6) determines the different baseline airship design parameters used in this study.

From table (4.4), notice that each one of fligt path angle, heading angle and velocity vary between two values along the proposed trajectory. Then, in order to calculate the

TABLE 4.5: Restrictions on the lighter than air vehicle

Minimum	Variable	Maximum	Units
-45	γ	45	deg
0.1	V	20	m/sec
-15	$\dot{\gamma}$	15	deg/sec
-20	$\dot{\chi}$	20	deg/sec
-1.25	$\dot{V}$	1.25	m/sec^2
0	T	10000	N
-15	σ	15	deg
-15	α	15	deg
-50	$\dot{T}$	50	N/sec
-5	$\dot{\sigma}$	5	deg/sec
-2	$\dot{\alpha}$	2	deg/sec

TABLE 4.6: Baseline lighter than air vehicle design parameters

Design parameter	Symbol	Value
Volume	U_h	2181.7 $[m^3]$
Mass	m	1750 $[kg]$
Added mass coefficient, x axis	k_x	0.082
Added mass coefficient, y axis	k_y	0.860
Added mass coefficient, z axis	k_z	0.860
Lift coefficient	C_{L0}	0.3118
Drag coefficient	C_{D0}	0.37

P matrix (defined in Equation (4.26)) only one operating point is considered. This point is defined as $OP \Leftrightarrow \gamma_{OP} = 5\ [deg]$, $\chi_{OP} = 45\ [deg]$ and $V = 10\ [m/sec]$. Furthermore, the selected Q and R matrices for this case are

$$Q = \begin{bmatrix} 10^3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10^3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10^4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix} \tag{4.61}$$

$$R = 10^4 \cdot \begin{bmatrix} 1/u_{1max}^2 & 0 & 0 \\ 0 & 1/u_{2max}^2 & 0 \\ 0 & 0 & 1/u_{3max}^2 \end{bmatrix} \tag{4.62}$$

Therefore, the resulting P matrix takes the following form

$$P = 10^6 \cdot \begin{bmatrix} 0.0096 & 0.0054 & 0.0510 & 0.0662 & -0.0622 & 0.0532 & 0 & -0.0256 & -0.0577 \\ 0.0054 & 0.0096 & 0.0510 & 0.0662 & 0.0622 & 0.0532 & 0 & 0.0256 & -0.0577 \\ 0.0510 & 0.0510 & 0.4920 & 0.9377 & 0 & 0.4983 & 0.0002 & 0 & 0.1240 \\ 0.0662 & 0.0662 & 0.9377 & 2.9910 & 0 & 0.9384 & 0.0003 & 0 & 2.3457 \\ -0.0622 & 0.0622 & 0 & 0 & 2.7642 & 0 & 0 & 1.5174 & 0 \\ 0.0532 & 0.0532 & 0.4983 & 0.9384 & 0 & 0.5167 & 0.0002 & 0 & 0.2013 \\ 0 & 0 & 0.0002 & 0.0003 & 0 & 0.0002 & 0 & 0 & 0 \\ -0.0256 & 0.0256 & 0 & 0 & 1.5174 & 0 & 0 & 1.2496 & 0 \\ -0.0577 & -0.0577 & 0.1240 & 2.3457 & 0 & 0.2013 & 0 & 0 & 8.3352 \end{bmatrix} \tag{4.63}$$

Furthermore, the chosen M matrix is

$$M = \begin{bmatrix} 10^{-2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10^{-2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10^{-2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10^6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10^5 \end{bmatrix} \tag{4.64}$$

The response of the system to the proposed control strategy is illustrated in Figure (4.19)

- Figure (4.29). The tracking position errors along x, y and z axes are presented in the Figure (4.19), Figure (4.20) and Figure (4.21) respectively. Also, the time profile of the vehicle's orientation (presented by its flight path and heading angles) and its velocity are shown in Figure (4.22), Figure (4.23) and Figure (4.24). Further, the thrust, bank angle and the angle of attack variables are demonstrated in Figure (4.25), Figure (4.26) and Figure (4.27) respectively. In addition, a three dimensional view of the spatial trajectory is displayed in the Figure (4.29). In these figures, the red line represents the desired signal while the blue line represents the system response. Finally, the control inputs are depicted in Figure (4.28).

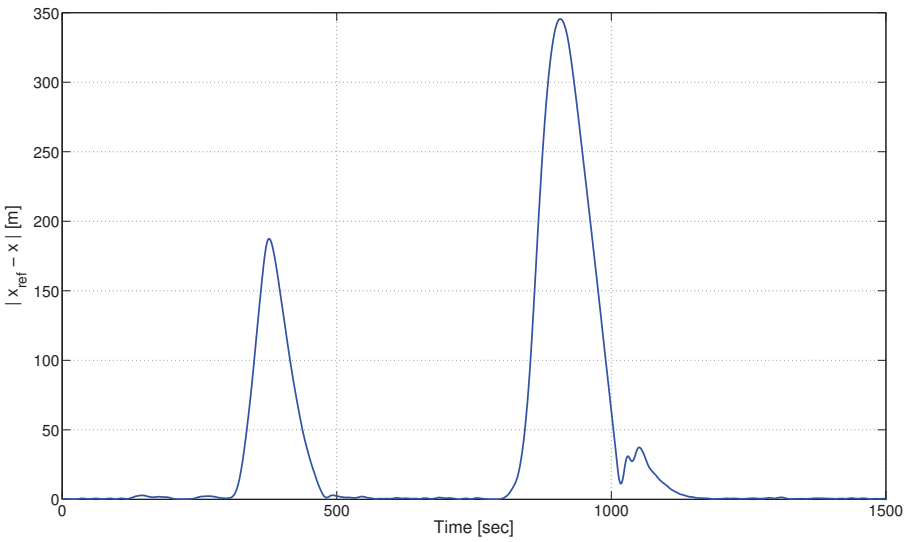

FIGURE 4.19: Relative error on x position (lighter than air vehicle).

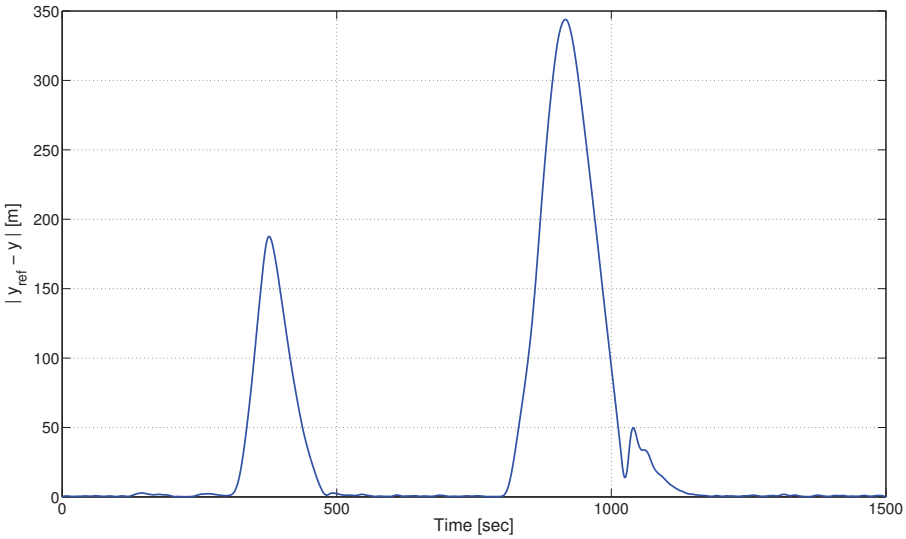

FIGURE 4.20: Relative error on y position (lighter than air vehicle).

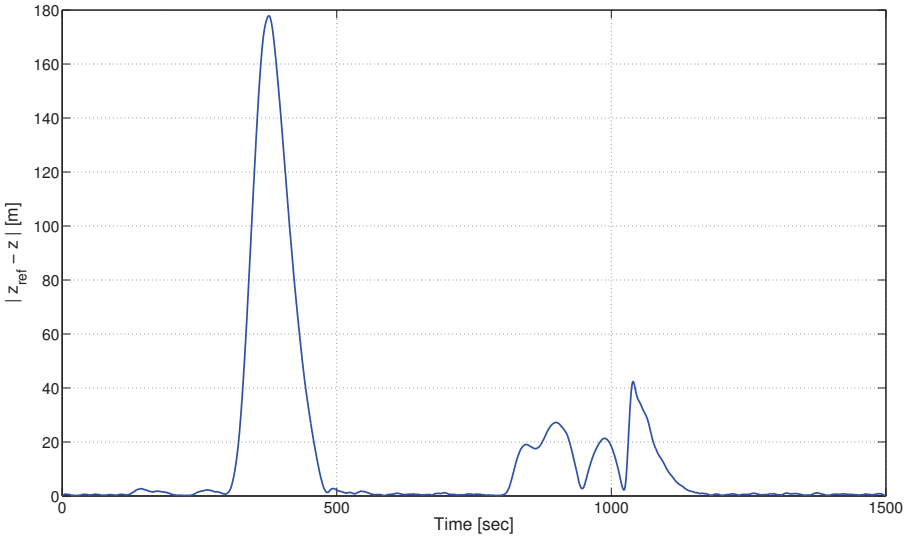

FIGURE 4.21: Relative error on z position (lighter than air vehicle).

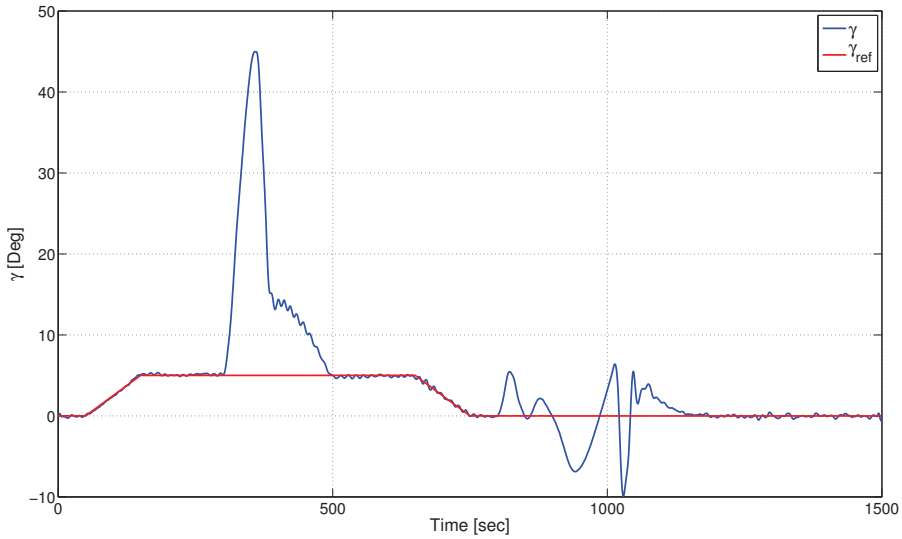

FIGURE 4.22: Flight Path Angle response in closed-loop (lighter than air vehicle).

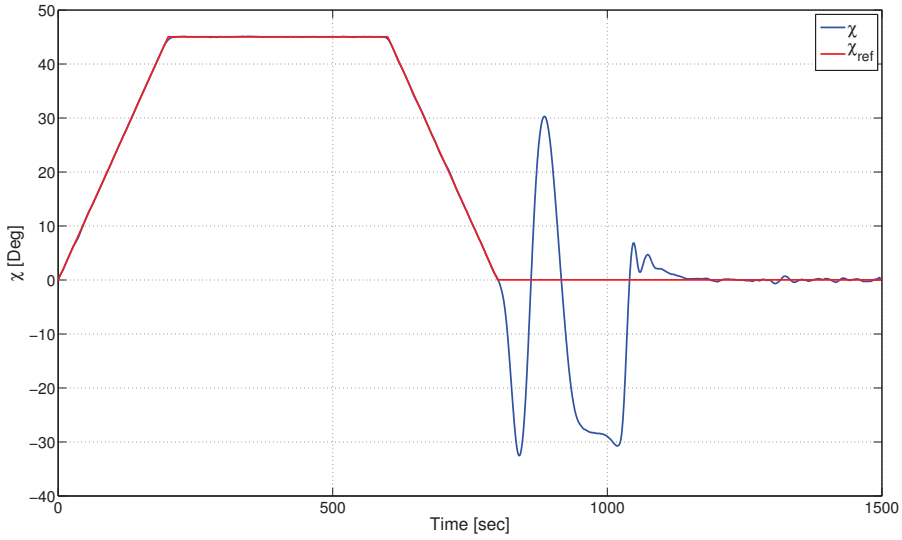

FIGURE 4.23: Heading Angle response in the closed-loop (lighter than air vehicle).

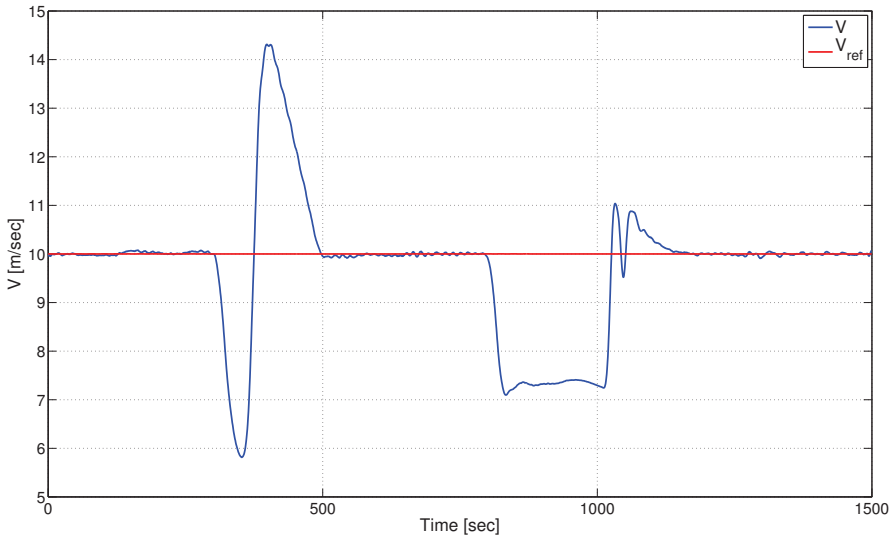

FIGURE 4.24: Velocity response in the closed-loop (lighter than air vehicle).

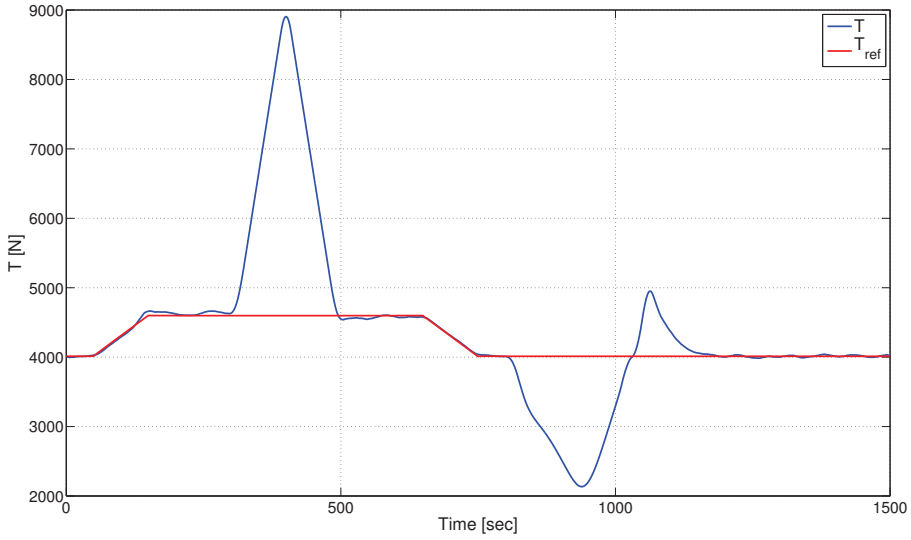

FIGURE 4.25: Thrust response in the closed-loop (lighter than air vehicle).

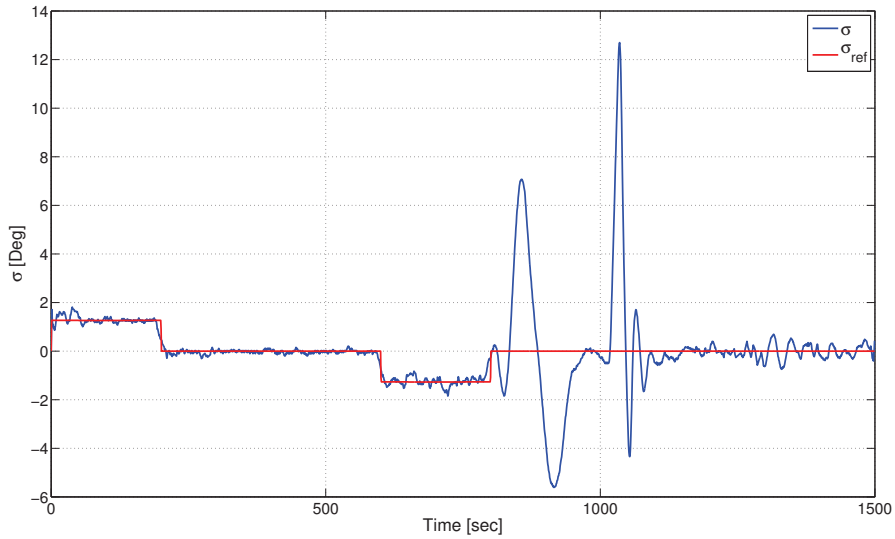

FIGURE 4.26: Bank angle response in the closed-loop (lighter than air vehicle).

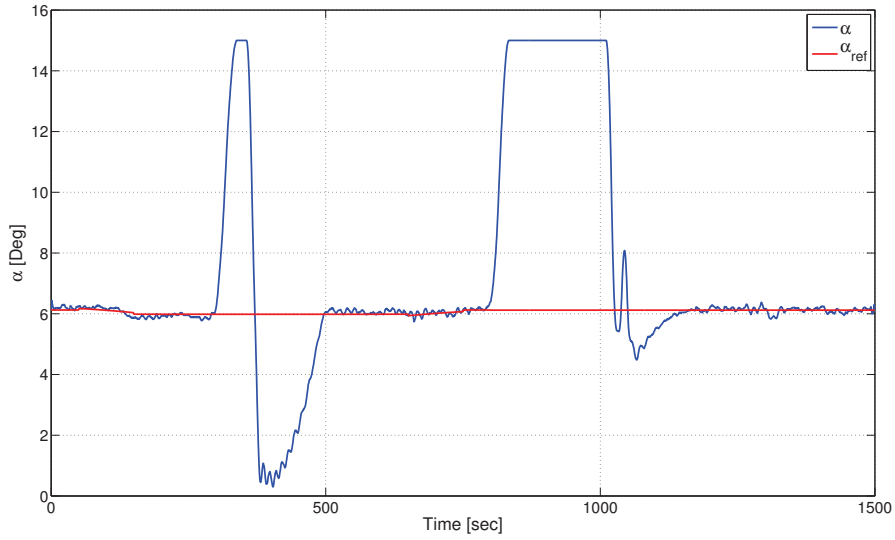

FIGURE 4.27: Angle of attack response in the closed-loop (lighter than air vehicle).

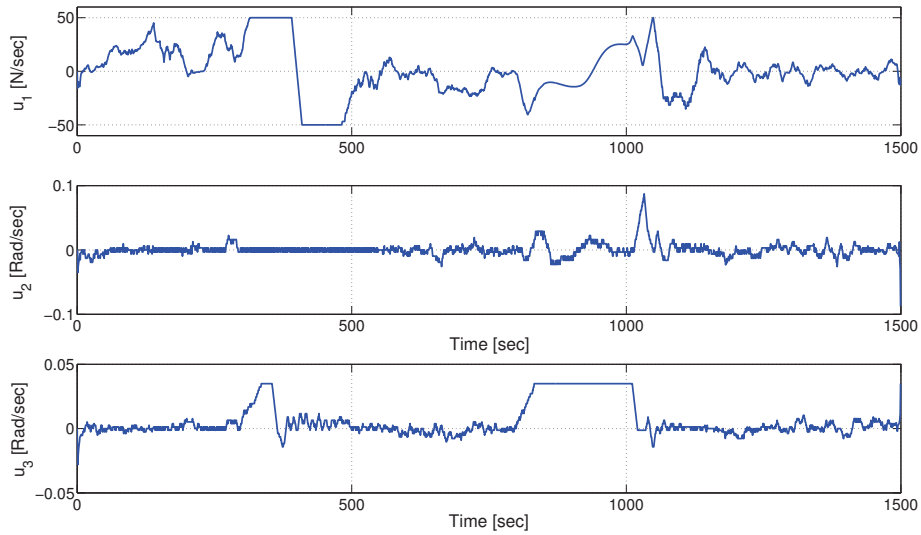

FIGURE 4.28: Control inputs responses u_1, u_2 and u_3 (lighter than air vehicle).

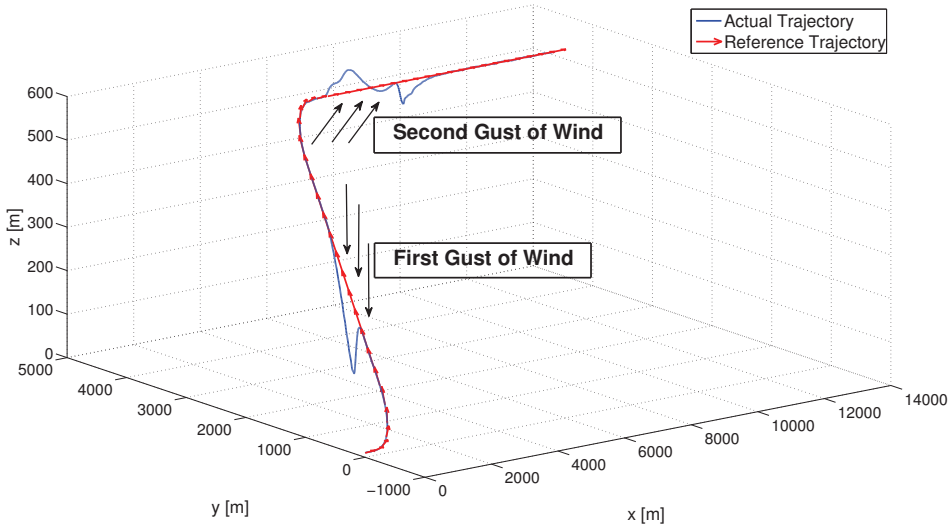

FIGURE 4.29: Three dimensional view of the trajectory.

Figure (4.19) - Figure (4.29) illustrate the attitude of the system tracking a reference trajectory and facing unknown disturbances. Notice that when the first wind gust arrives (at $t = 300$ $[sec]$), the control inputs increase the thrust to 8900 Newton while the angle of attack varies between 0.3 and 15 degrees before it stabilized at 6.12 degrees. Observe also that there is no changes on the bank angle response to the first gust because this blast affects only the z axis and when it arrives, the horizontal orientation of the vehicle is already stabilized at a constant value. On the other hand, when the second gust arrives (at $t = 800$ $[sec]$), the control inputs make the thrust to vary between 2120 and 4950 Newton, the bank angle to change between -5.6 and 12 degrees and the angle of attack to be increased to 15 degrees. These changes affect directly the behavior of each of γ, χ and V as it is demonstrated in Figure (4.22), Figure (4.23) and Figure (4.24). From previous we conclude that the proposed control strategy has a good performance guaranteeing the trajectory tracking even in the presence of unknown wind gusts.

4.6 Trajectory Tracking for Quadrotors

The aim of this section is to examine the applicability and compatibility of the reference trajectories with the dynamics of the quadrator. For this purpose, we employ the trajectory generated by the previous method to the rotorcraft's rigid body model, Equation (2.63) as reference signals; and we design an autopilot to ensure the trajectory tracking. The control strategy used here is based on the control laws introduced in [32] and [33] for stabilizing quadrotor at hover.

The controller regulates each one of the state variables in a sequence according to a predefined priority rule as follows:
First of all, the desired altitude (z) is reached using the control input u. Then, the yaw angle (ψ) is controlled through τ_ψ. Next, the desired values of pitch angle (ϕ) and y-displacement are reached by controlling τ_ϕ. Finally, the control input τ_θ is used to obtain the desired pitch angle (θ) and x-displacement values.

Next, the following definition is fundamental for the control strategy.

Definition 4.6. The function $\sigma(t)$ is said to be a saturation function if and only if

$$|\sigma(t)| \leq M; \forall M \in \mathbb{R}^+ \tag{4.65}$$

4.6.1 Altitude and Yaw Control

The control of z-displacement is ensured through the following control input

$$u = \sec(\theta)\sec(\phi)\bar{r} \tag{4.66}$$

where

$$\bar{r} = -\mathfrak{K}_{z1}\left(\dot{z} - \dot{z}_{ref}\right) - \mathfrak{K}_{z2}\left(z - z_{ref}\right) + mg \tag{4.67}$$

with, $\mathfrak{K}_{z1}$ and $\mathfrak{K}_{z2}$ are positive constant, $\dot{z}_{ref}$ and z_{ref} are the desired vertical velocity and altitude respectively. In addition, θ and ϕ are assumed to be limited such that they can not reach 90 degrees.

On the other side, the yaw angle is controlled by applying

$$\tau_{\psi} = -\sigma_{\psi 1}\left(\mathfrak{K}_{\psi 1}\left(\dot{\psi} - \dot{\psi}_{ref}\right)\right) - \sigma_{\psi 2}\left(\mathfrak{K}_{\psi 2}\left(\psi - \psi_{ref}\right)\right) \tag{4.68}$$

Substituting Equation (4.67) and Equation (4.68) into Equation (2.63c) and Equation (2.63d), it follows that: $\dot{z} \mapsto \dot{z}_{ref}$, $z \mapsto z_{ref}$, $\dot{\psi} \mapsto \dot{\psi}_{ref}$ and $\psi \mapsto \psi_{ref}$.

4.6.2 Roll and Lateral Position Control

Once $\dot{z}$ and z are stabilized, the Equation (2.63a) and Equation (2.63b) can be reduced to

$$\ddot{x} = -g\tan(\theta)\sec(\phi) \tag{4.69a}$$

$$\ddot{y} = g\tan\left(\phi\right) \tag{4.69b}$$

Considering the subsystem (ϕ, y) given by Equation (4.69b) and Equation (2.63f). Then, the control input for this subsystem is given by

$$\begin{aligned} \tau_{\phi} = \quad & -\sigma_{\phi 1}\left(\mathfrak{K}_{\phi 1}\left(\dot{y} - \dot{y}_{ref}\right)\right) - \sigma_{\phi 2}\left(\mathfrak{K}_{\phi 2}\left(y - y_{ref}\right)\right) \\ & -\sigma_{\phi 3}\left(\mathfrak{K}_{\phi 3}\left(\dot{\phi}\right)\right) - \sigma_{\phi 4}\left(\mathfrak{K}_{\phi 4}\left(\phi - \phi_{ref}\right)\right) \end{aligned} \tag{4.70}$$

where the controller gain parameters $\mathfrak{K}_{\phi 1}$, $\mathfrak{K}_{\phi 2}$, $\mathfrak{K}_{\phi 3}$ and $\mathfrak{K}_{\phi 4}$ are positive constant.

4.6.3 Pitch and Forward Position Control

From Equation (4.70) and Equation (4.69b) it follows that $\phi \mapsto 0$. Then, the Equation (4.69a) becomes $\ddot{x} = -g\tan(\theta)$ Finally, the subsystem (θ, x) is controlled through τ_{θ}

which is given by

$$\begin{aligned}\tau_\theta = \quad & \sigma_{\theta 1}\left(\mathfrak{K}_{\theta 1}\left(\dot{x}-\dot{x}_{ref}\right)\right)+\sigma_{\theta 2}\left(\mathfrak{K}_{\theta 2}\left(x-x_{ref}\right)\right) \\ & -\sigma_{\theta 3}\left(\mathfrak{K}_{\theta 3}\left(\dot{\theta}\right)\right)-\sigma_{\theta 4}\left(\mathfrak{K}_{\theta 4}\left(\theta-\theta_{ref}\right)\right)\end{aligned} \tag{4.71}$$

where the controller gain parameters $\mathfrak{K}_{\theta 1}$, $\mathfrak{K}_{\theta 2}$, $\mathfrak{K}_{\theta 3}$ and $\mathfrak{K}_{\theta 4}$ are positive constant.

4.7 Simulation Results

In this section, we consider a scenario of a quadrotor flying at fixed altitude. The trajectory consists of an initial configuration, an initial one, and several way-configurations to be visited by the aerial vehicle. Then, the optimization process described in sec. 3.3.2.1 is applied for every segment separately. Once the reference trajectory is generated, an autopilot is designed to evaluate the validity of a such trajectory.

The constraints on pitch angle, velocity, and control variables are provided in Table 4.7.

TABLE 4.7: Restrictions on state and control variables

Minimum	Variable	Maximum	Units
- 15	θ	15	deg
0.001	V	0.2	m/sec
- 5	$\dot{\theta}$	5	deg/sec
- 10	$\dot{\psi}$	10	deg/sec
- 0.01	$\dot{V}$	0.01	m/sec^2

As the vehicle is moving only in the horizontal plane ($x \times y$), both $\dot{\theta}$ and θ are equal to zero. In addition we assume the lack of wind in the environment. Then, the equations of motion presented in Equation (2.99) and Equation (2.101) are reduced to the following

$$\begin{aligned}\dot{x} &= V\cos\psi \\ \dot{y} &= V\sin\psi \\ \dot{\psi} &= u_1 \\ \dot{V} &= u_2\end{aligned}$$

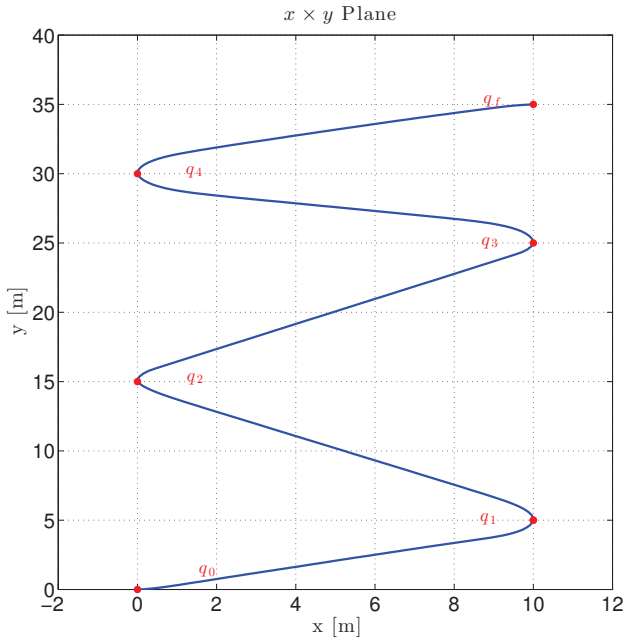

FIGURE 4.30: Flying at a constant altitude trajectory.

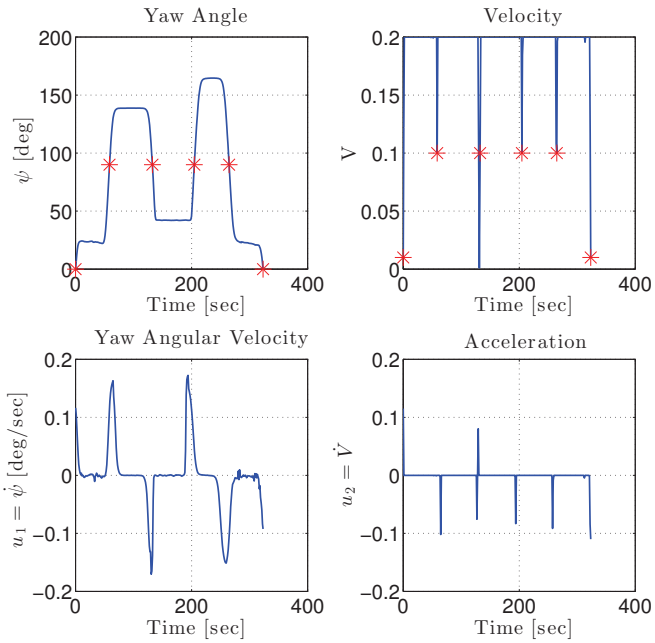

FIGURE 4.31: Time profile of yaw angle ψ, vehicle's velocity and their rate of change $\dot{\psi}$, $\dot{V}$ respectively.

The number of nodes (N) used in each segment is 84, and the interior point algorithm included in *fmincon solver* in MATLAB® is used to compute optimal trajectory. The initial configuration q_0, the final one q_f, and the way-configurations q_i, in this case, are presented in Table 4.8. The position, orientation, velocity, and control inputs of the aerial vehicle are illustrated in Figure (4.30) and Figure (4.31).

TABLE 4.8: Configurations to be visited during the traveled trajectory

	$x\ [m]$	$y\ [m]$	$\psi\ [deg]$	$V\ [m/sec]$
q_0	0	0	0	0.01
q_1	10	5	90	0.1
q_2	0	15	90	0.1
q_3	10	25	90	0.1
q_4	0	30	90	0.1
q_f	10	35	0	0.01

Notice the final configuration is reached in $t_f = 323.2$ [sec].

Once the reference trajectory is generated we go forward to the tracking problem in order to evaluate the applicability and the validity of the calculated reference trajectory with the dynamics of quadrotor. For this end, we take the results found above as references for the full dynamic quadrotor model presented in Equation (2.63). The control parameters and the limitations of saturation functions values were chosen to ensure a stable well-damped response specially for x, y and ψ variables.

The performance of the designed autopilot is illustrated in the Figure (4.32) - Figure (4.35). In these figures, the solid line represents the system response and the dashed line describes the desired value or trajectory. The time profile of the yaw angle and its derivative are shown in Figure (4.32). While, the time profile of the x and y displacement and their absolute error are presented in Figure (4.33) and Figure (4.34) respectively. In Figure (4.35), the time profile of pitch and roll angles are depicted.

Note that the controller has a good performance to tracks the yaw angle, x and y displacements. While, it has not the same efficiency to follow the yaw angular velocity. This behavior is due to the fact that the gain assigned to yaw angle (i.e. $\mathfrak{K}_{\phi 2}$) is more important than the one dedicated to its derivative (i.e. $\mathfrak{K}_{\phi 1}$). On the other hand, observe that the desired pitch and roll angles are set to be zeros but the real θ and

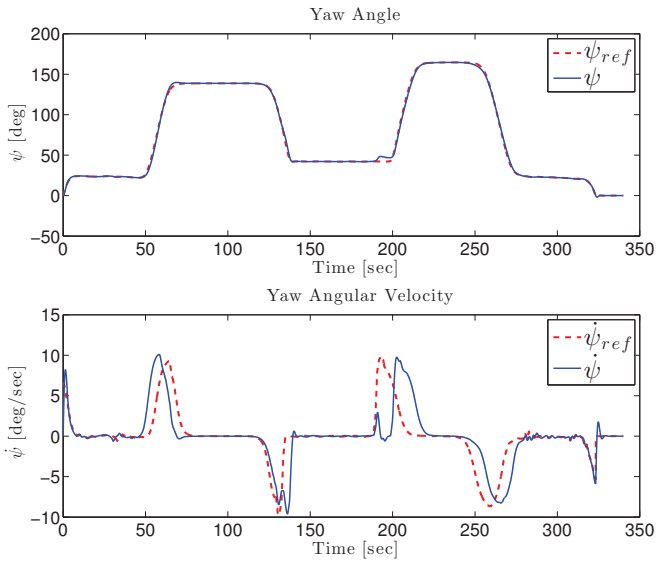

FIGURE 4.32: Yaw angle of quadrotor.

ϕ differs for some periods of time which is due to movements on x and y axis and acceleration/deceleration effects.

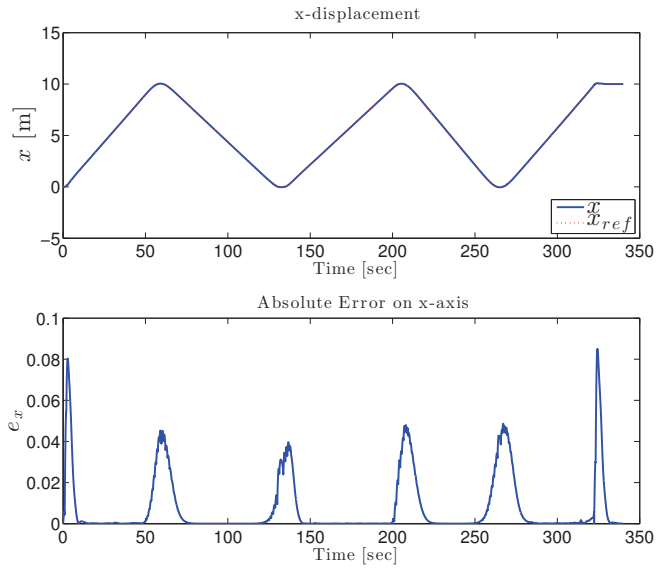

FIGURE 4.33: x-displacement of quadrotor.

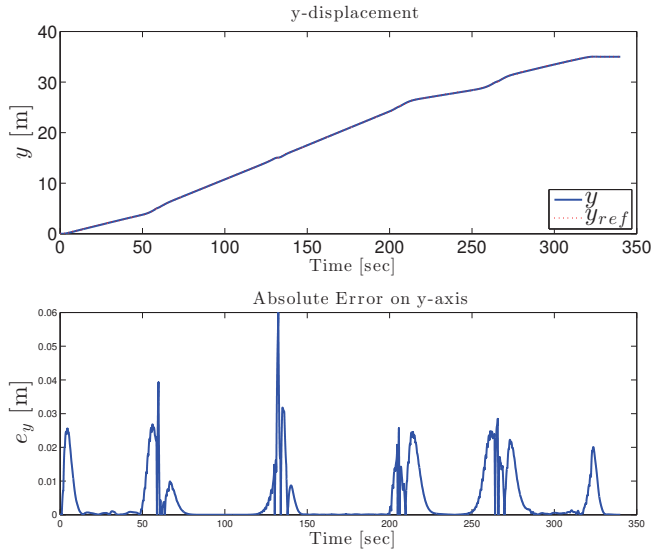

FIGURE 4.34: y-displacement of quadrotor.

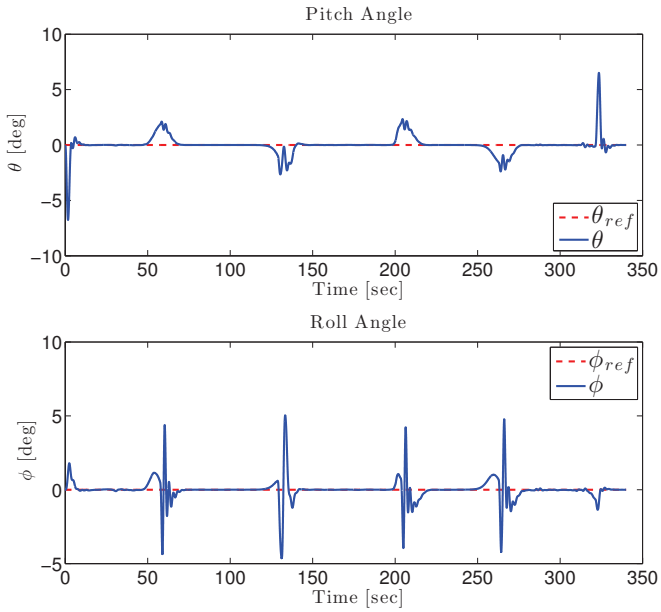

FIGURE 4.35: Time profile of pitch (θ) and roll (ϕ) angles.

4.8 Conclusion

In this chapter, Firstly, we have formulated and solved inverse optimal robust stabilization problem to guarantee the trajectory tracking for a kinematic aerial vehicle and for a dynamic lighter than air vehicle. These systems belongs to a specified class of nonlinear control systems which are affine in control and disturbances inputs. The formulas (4.22) and (4.23) introduced in [62] are used to generate robust control laws which have the desirable properties of optimality and do not require the solution of HJI equation. Some numerical simulations were carried out and some graphs were presented to illustrate the good performance of the closed-loop system even in presence of unknown disturbances.

Secondly, an autopilot for an autonomous quadrotor has been designed. We formulated a minimum time optimal control problem to generate the reference trajectory using the vehicle's point mass model. While the control strategy in the tracking problem was designed through the quadrotor's rigid body model.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

UAV's are a promising solution to the structure inspection problem. In fact the nature of such missions implies the interaction of different research domains including, automatic control, computer vision, image processing, wireless communications, aerial vehicle design, etc... The goal of this thesis is the development of motion planning, trajectory generation and tracking algorithms in the presence of wind. Since inspection missions require an aerial vehicle with a hovering capability, we were interested by two types of unmanned aerial vehicles: Lighter than air vehicle and quadrotors. Thereby, we begin with the mathematical representation for those aerial vehicles. Firstly, a six degree of freedom equations of motion, called rigid body model, was developed for lighter than air vehicle using Newton- Euler approach, and for quadrotor crafts through Euler-Lagrange technique. This model is devoted to treat the stability and control problems. Next, a three degree of freedom translational equations of motion, called point mass model, including the wind was introduced. This model is based on Newton's second law and used to deal with navigation and guidance control systems. Translational kinematics and dynamics for airships was realized, while the kinematic model was adopted to present quadrotors crafts.

First of all, the accessibility and controllability issues were considered. It has been proved that a kinematic aerial vehicle is strongly accessible under the condition that the velocity is not equal to zero and if its flight path angle do not reach $\frac{\pi}{2}$. Moreover, it has been shown that the vehicle could be completely controllable if the wind velocity is strictly less than the vehicle's velocity.

The proposed methods for reference trajectory generation are based on optimality notion. The objective is chosen to be the traveling time. The minimum-time flight problem

is formulated as a problem of the calculus of variations. Then, this formulation is converted to a nonlinear constrained optimization problem and it is solved through a direct collocation approach. Geometric characterization of the candidate trajectories satisfying the necessary conditions for time optimality is also described. Simulations are carried out to validate the proposed algorithm. These results show the wind effect on the computed trajectory. In fact, it can be a benefit in the sense where the wind push the vehicle from the back, and a disadvantage if it comes from the front of the vehicle. Next, a suboptimal fast trajectory generation algorithm based on geometrical solution is proposed, and an illustrative example is carried out. The main advantage of this algorithm is its rapidity, while its main weakness is its applicability when the distance separating the initial and final configurations is too small.

Besides, an updated flight planning strategy is proposed in which, the wind was considered as a constant piecewise function. The basic idea is to supervise the trajectory with a spatiotemporal criterion and to decide when the reference trajectory has to be updated. Simulation results reveal the presence of discontinuities in reference trajectory. This phenomenon is caused by the updates when the real situation is taken into account.

The proposed motion planning method takes into consideration the energy limitation of UAV as a maximal allowed flight time. In fact, a modification of a capacitated vehicle routing problem is proposed. The limited carrying capacity of goods are replaced by an allowed maximum flight time. Simulation results show that the proposed approach is capable of finding time-optimal routes, taking into account the wind, while respecting the maximum flight time of the vehicle.

Trajectory tracking problem is also addressed in this thesis. The proposed control strategy should face the presence of unknown gusts. Then, a robust nonlinear control law based on inverse optimality and robust Lyapunov Control Function is introduced for kinematic aerial vehicle and for a dynamic lighter than air vehicle. The carried out simulations prove the good performance of the controller.

5.2 Future Work

This thesis has highlighted a number of areas that would merit further investigation. Some of these are listed below:

- All algorithms, introduced in this work, have been validated in simulations. Therefore, an experimental validation should be carried out to confirm simulation results.
- The proposed sub-optimal fast algorithm takes into consideration only a planar flight case, i.e. flying at a constant altitude. It will be interesting to improve this algorithm to include a 3D case.
- In this work, all state variables and wind velocity are assumed to be available, which could not be the case in real systems. Then, it becomes necessary to develop a state and wind estimators to overcome such problem.
- The input of the trajectory planner is supposed to be a set of points predefined by the user which is very difficult to handle for large-scale structures. Thus, an automatic selection of points of interest methods is needed to be developed. This can be done through meshing techniques for example.
- All developed methods and algorithms, in this thesis, are dedicated for a translational flight. Consequently, a study on the stability of a hovering flight will be needed to complete our work.

Appendix A

Determining the Direction of Turn

Considering three points O, P, and M. Where, O presents the origin of Earth fixed inertial coordinate system, P is the actual position and orientation of the aerial vehicle, and M denotes the desired destination configuration. Lets $R_{\mathcal{N}}$ to be a new coordinate system having its origin located at P. So that, its $y_{\mathcal{N}}$-axis coincide with the heading of the vehicle while its $x_{\mathcal{N}}$-axis, perpendicular to $y_{\mathcal{N}}$, points to the right direction. Therefore, in order to define in which direction the aircraft must go to reach its final destination, it is sufficient to look at the projection of M in $R_{\mathcal{N}}$, see Figure (3.11).

The projection of M in $R_{\mathcal{N}}$ is obtained through a translation and a rotation.

A.1 Translation

The position of M with respect to P is defined through the vector $\overrightarrow{PM}$. From Figure (A.1) we state

$$\overrightarrow{PM} = \overrightarrow{PO} + \overrightarrow{OM} \tag{A.1}$$

where,

$$\overrightarrow{PO} = (x_O - x_P)\,\overrightarrow{i} + (y_O - y_P)\,\overrightarrow{j} \tag{A.2a}$$

$$\overrightarrow{OM} = (x_M - x_O)\,\overrightarrow{i} + (y_M - y_O)\,\overrightarrow{j} \tag{A.2b}$$

Therefore,

$$\overrightarrow{PM} = (x_M - x_P)\,\overrightarrow{i} + (y_M - y_P)\,\overrightarrow{j} \tag{A.3}$$

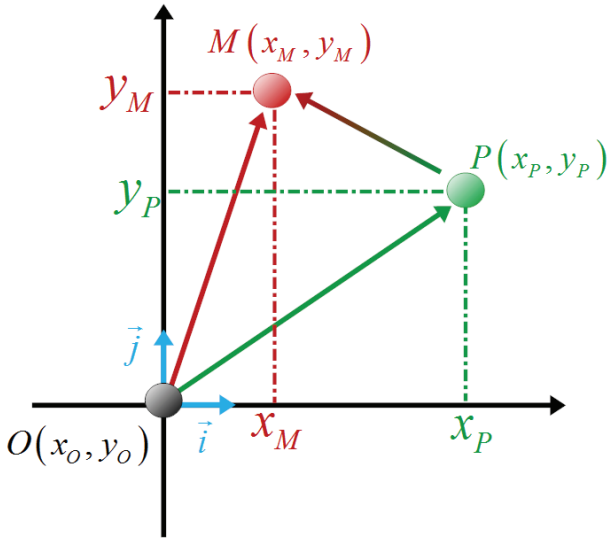

FIGURE A.1: Translation.

A.2 Rotation

The orientation of the vehicle with respect to $R_{\mathcal{I}}$ can be described through a counter-clockwise rotation with an angle θ, see Figure (A.2). Then, we have

$$\begin{bmatrix} \vec{i}' \\ \vec{j}' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \vec{i} \\ \vec{j} \end{bmatrix} \tag{A.4}$$

Therefore,

$$P\left(x_P, y_P\right)|_{\vec{i}', \vec{j}'} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_P \vec{i} \\ y_P \vec{j} \end{bmatrix} \tag{A.5}$$

with, $\theta = \frac{3\pi}{2} + \chi$.

Consequently, the projection of the destination goal M onto $R_{\mathcal{N}}$ is expressed by the following relationship

$$M\left(x_M, y_M\right)|_{\vec{i}', \vec{j}'} = \begin{bmatrix} \cos\left(\frac{3\pi}{2} + \chi\right) & \sin\left(\frac{3\pi}{2} + \chi\right) \\ -\sin\left(\frac{3\pi}{2} + \chi\right) & \cos\left(\frac{3\pi}{2} + \chi\right) \end{bmatrix} \begin{bmatrix} \left(x_M - x_P\right) \vec{i} \\ \left(y_M - y_P\right) \vec{j} \end{bmatrix} \tag{A.6}$$

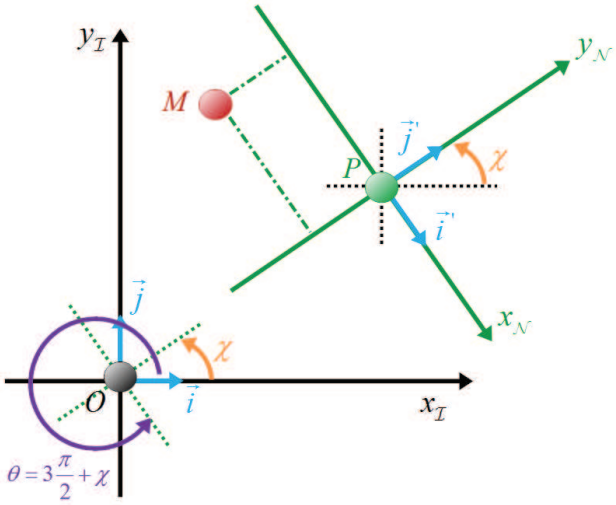

FIGURE A.2: Rotation.

Appendix B

Linear Phase Calculation

Remember that the equations of motion for an aerial vehicle flying at a fixed altitude is given by

$$\dot{x} = V\cos\chi \tag{B.1a}$$
$$\dot{y} = V\sin\chi \tag{B.1b}$$
$$\dot{\chi} = u_1 \tag{B.1c}$$
$$\dot{V} = u_2 \tag{B.1d}$$

Flying in a straight line means that the orientation of the aircraft is fixed, i.e. $\chi = \chi_0$. This implies that $\dot{\chi} = u_2 = 0$. Then, integrating the Equation (B.1) leads to the following

$$x = \frac{1}{2}u_2\cos\chi_0 t^2 + V_0\cos\chi_0 t + x_0 \tag{B.2a}$$
$$y = \frac{1}{2}u_2\sin\chi_0 t^2 + V_0\sin\chi_0 + y_0 \tag{B.2b}$$
$$\chi = \chi_0 \tag{B.2c}$$
$$V = u_2 t + V_0 \tag{B.2d}$$

with $u_2 = \pm u_{2max}$ for acceleration and deceleration phases, while $u_2 = 0$ entails the constant velocity case.

B.1 Problem Statement

Considering two configurations $q_1 = [x_1, y_1, \chi_1, V_1]$ and $q_2 = [x_2, y_2, \chi_2, V_2]$, with $\chi_1 = \chi_2 = cte$. In addition, assuming that reaching q_2 from q_1 can be done through a straight

line. Then, the basic problem is to find a control $u_2 \in \{0, \pm u_{2max}\}$ such that the continuity of the aerial vehicle's velocity is guaranteed. In other words, defining the acceleration, deceleration, and constant velocity segments along the trajectory.

B.2 Proposed Solution

To deal with the previous problem, we propose the following algorithm

Algorithm B.1 Linear Segments Calculation

Require: initial and final configurations (q_1 and q_2).
1: Compute the distance (d) between q_1 and q_2.
2: Determine the required distance for the acceleration/deceleration phase (d_{acc} and d_{dec}).
3: **if** $d > d_{acc} + d_{dec}$ **then**
4: Three linear segments are required to reach q_2:
Compute switching points and traveled time for acceleration, deceleration, and constant velocity segments.
5: **else**
6: Two linear segments are required to reach q_2:
Compute the time and the point to switch between the acceleration and deceleration phases.
7: **end if**
8: **return** Switching time and configurations as well as control inputs along the trajectory.

In the following, we describe in details the previous algorithm.

Step 1: Calculate the distance (d) between q_1 and q_2

This distance is given by:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{B.3}$$

Step 2: Determine the required distance for the acceleration/deceleration phase (d_{acc} and d_{dec})

The required time to acceleration phase is obtained by

$$t_{acc} = \frac{V_{max} - V_1}{u_{2max}} \tag{B.4}$$

Thus, from Equation (B.2a) and Equation (B.2b) we obtain

$$x_{acc} = \frac{1}{2} u_{2max} \cos\chi_0 t_{acc}^2 + V_0 \cos\chi_0 t_{acc} + x_1 \tag{B.5a}$$

$$y_{acc} = \frac{1}{2} u_{2max} \sin\chi_0 t_{acc}^2 + V_0 \sin\chi_0 t_{acc} + y_1 \tag{B.5b}$$

where, (x_{acc}, y_{acc}) presents the end point of acceleration phase. Therefore, the traveled distance during the acceleration phase is given by

$$d_{acc} = \sqrt{(x_{acc} - x_1)^2 + (y_{acc} - y_1)^2} \tag{B.6}$$

In a similar way, we state

$$t_{dec} = \frac{V_f - V_{max}}{-u_{2max}} \tag{B.7}$$

$$x_{dec} = x_2 - \frac{1}{2}(-u_{2max}) \cos \chi_0 t_{dec}^2 - V_{max} \cos \chi_0 t_{dec} \tag{B.8a}$$

$$y_{dec} = y_2 - \frac{1}{2}(-u_{2max}) \sin \chi_0 t_{dec}^2 - V_{max} \sin \chi_0 t_{dec} \tag{B.8b}$$

$$d_{dec} = \sqrt{(x_f - x_{dec})^2 + (y_f - y_{dec})^2} \tag{B.9}$$

with, t_{dec} is the required time to decelerate from V_{max} to V_2, (x_{dec}, y_{dec}) is the start point of the deceleration phase, and d_{dec} is the traveled distance during the deceleration phase.

Step 4: Calculate switching points and traveled time for acceleration, deceleration, and constant velocity segments

The switching points as well as the traveled time for acceleration and deceleration phases has been already calculated in step 2. Then it remains to calculate the traveled time for the constant segment. This one is given by

$$t_{const} = \frac{d_{const}}{V_{max}} \tag{B.10}$$

where, d_{const} denotes the traveled distance which is obtained through the following relationship

$$d_{const} = \sqrt{(x_{dec} - x_{acc})^2 + (y_{dec} - y_{acc})^2} \tag{B.11}$$

Step 6: Calculate the time and the point to switch between the acceleration and deceleration phases

Assuming two particles moving on the same line in an opposite direction. The first one, P_1, located at q_1, while the second one, P_2, based at q_2. In addition, both are in acceleration phase. Then the interception point of these two particles corresponds to the switching time and point between the acceleration and deceleration phases.

Next, we distinguish the following three cases:

Case 1: $V_1 = V_2$

From Equation (B.2a), we state

$$x_{P_1} = \frac{1}{2} u_{2max} \cos\chi_1 t^2 + V_1 \cos\chi_1 t + x_1 \tag{B.12a}$$

$$x_{P_1} = \frac{1}{2} u_{2max} \cos(\chi_2 + \pi) t^2 + V_2 \cos(\chi_2 + \pi) t + x_2 \tag{B.12b}$$

where, x_{P_1} and x_{P_2} denote the time profile of x-coordinate for the first and second particle respectively. Then, at the interception point, we have $x_{P_1} = x_{P_2}$. So,

$$\frac{1}{2} u_{2max} \left(\cos\chi_1 - \cos(\chi_1 + \pi)\right) t^2 + \left(V_1 \cos\chi_1 - V_2 \cos(\chi_1 + \pi)\right) t + x_1 - x_2 = 0 \tag{B.13}$$

Since the time is the only unknown variable in the previous equation, then solving this equation gives the switching time, t_{swt}. Consequently, the switching velocity is given by

$$V_{swt} = u_{2max} t_{swt} + V_1 \tag{B.14}$$

Case 2: $V_1 < V_2$

In this case, we compute the acceleration time t_n required to reach a new configuration q_n such that $V_n = V_2$. Then, we adopt the same strategy described in case 1, after substituting q_1 by q_n, to compute the switching configuration.

Case 3: $V_1 > V_2$

This situation is the opposite of case 2. So, we compute the acceleration time t_n necessary to reach a new configuration q_n such that $V_n = V_1$. Next, we we compute the switching configuration in a similar way to case 1 after replacing q_2 by q_n.

Appendix C

Trajectory Generation and Circular Arcs

Assuming that the condition presented in the line 5 of the algorithm B.1 is not satisfied, then two circumference Σ_{i_j} and Σ_{f_j} containing the end configuration of S_i and the start configuration of S_f respectively must be included to the trajectory. The index j defines the radius of the circumference with respect to the velocity (max for V_{max} and min for V_{min}). Remember that in this case the velocity is a constant.

C.1 Circular arc equations

Flying in a circular motion implies that the velocity of the aerial vehicle is fixed while its orientation is linearly changing. Then, the equations characterizing the circular motion are

$$x = x_0 + \frac{V_0}{u_1}(\sin\chi - \sin\chi_0) \tag{C.1a}$$

$$y = y_0 - \frac{V_0}{u_1}(\cos\chi - \cos\chi_0) \tag{C.1b}$$

$$\chi = \chi_0 + u_1 t \tag{C.1c}$$

$$V = V_0 \tag{C.1d}$$

where, $u_1 = \pm u_{1max}$ for left or right turn, and $q_0 = [x_0, y_0, \chi_0, V_0]$ is the vehicle's initial configuration.

C.2 Common Tangent between Two Circles

In general, two circles have four common tangent (two outer tangents and two inner tangents). To defines these tangents we need, first of all, to find the center and the radius of the circles. Then, we can compute the tangential points on each circle.

C.2.1 Determining the Center of Circle

Defining the center of a circle which contains a configuration $q_1 = [x_1, y_1, \chi_1, V_1]$ can be done by the following four stages.

Step 1: This step consist of determining the radius of turn. Since the velocity of an object traveling the circle is:

$$V_1 = \frac{2\pi r}{T} = \omega r \tag{C.2}$$

where, r is the radius, T is the time required for one rotation, and ω is the angular velocity. So we have:

$$r = \frac{V_1}{\omega} \tag{C.3}$$

Step 2: Defining the straight line, L_1 which passes in (x_1, y_1) in the direction of χ_1.

$$L_1 : \quad y = a_1 x + b_1 \tag{C.4}$$

with $a_1 = \tan^{-1}(\chi_1)$ and $b = y_1 - a_1 x_1$.

Step 3: Characterizing the perpendicular line to L_2.

$$L_2 : \quad y = a_2 x + b_2 \tag{C.5}$$

with $a_2 = -\frac{1}{a_2}$ and $b_2 = y_1 - a_2 x_1$.

Step 4: The intersection of L_2 with the circle having a radius r and centered at (x_1, y_1):

$$y = a_2 x + b_2 \tag{C.6a}$$

$$r^2 = (x - x_1)^2 + (y - y_1)^2 \tag{C.6b}$$

gives the center of turn.

C.2.2 Tangential Points

Considering the following two circles:

$$C_1: \quad (x-a)^2 + (y-b)^2 = r_1^2 \tag{C.7a}$$
$$C_2: \quad (x-c)^2 + (y-d)^2 = r_2^2 \tag{C.7b}$$

where (a, b), r_1, (c, d), and r_2 present the center and the radius of C_1 and C_2 respectively. In addition assuming that $r_1 > r_2$ as it is shown in Figure (C.1). Then, the intersection point of outer tangents P_o is given by

$$x_{P_o} = \frac{r_2 a - r_1 c}{r_2 - r_1} \tag{C.8a}$$
$$y_{P_o} = \frac{r_2 b - r_1 d}{r_2 - r_1} \tag{C.8b}$$

Thus, the tangential points of the outer tangents located at C_1 are:

$$x_{t_{o1,2}} = \frac{r_1^2(x_{P_o} - a) \pm r_1(y_{P_o} - b)\sqrt{(x_{P_o} - a)^2 + (y_{P_o} - b)^2 - r_1^2}}{(x_{P_o} - a)^2 + (y_{P_o} - b)^2} + a \tag{C.9a}$$

$$y_{t_{o1,2}} = \frac{r_1^2(y_{P_o} - b) \mp r_1(x_{P_o} - a)\sqrt{(x_{P_o} - a)^2 + (y_{P_o} - b)^2 - r_1^2}}{(x_{P_o} - a)^2 + (y_{P_o} - b)^2} + b \tag{C.9b}$$

In a similar way, the tangential points located at C_2, i.e. $x_{t_{o3,4}}$, $y_{t_{o3,4}}$, can be computed through Equation (C.9) by replacing $a \to c$, $b \to d$, and $r_1 \to r_2$.

Therefore, the outer tangents are characterized by the equations

$$L_{t_o}1: \quad y = \left(\frac{y_{t_{o3}} - y_{t_{o1}}}{x_{t_{o3}} - x_{t_{o1}}}\right) x + \left(y_{t_{o3}} \left(\frac{y_{t_{o3}} - y_{t_{o1}}}{x_{t_{o3}} - x_{t_{o1}}}\right)\right) x_{t_{o3}} \tag{C.10a}$$

$$L_{t_o}2: \quad y = \left(\frac{y_{t_{o4}} - y_{t_{o2}}}{x_{t_{o4}} - x_{t_{o2}}}\right) x + \left(y_{t_{o4}} \left(\frac{y_{t_{o4}} - y_{t_{o2}}}{x_{t_{o4}} - x_{t_{o2}}}\right)\right) x_{t_{o4}} \tag{C.10b}$$

Besides, the intersection point of the two inner tangents is

$$x_{P_i} = \frac{r_1 c + r_2 a}{r_1 + r_2} \tag{C.11a}$$
$$y_{P_i} = \frac{r_1 d + r_2 b}{r_1 + r_2} \tag{C.11b}$$

Then, the tangential points located at C_1 are defined as

$$x_{t_{i1,2}} = \frac{r_1^2(x_{P_i} - a) \pm r_1(y_{P_i} - b)\sqrt{(x_{P_i} - a)^2 + (y_{P_i} - b)^2 - r_1^2}}{(x_{P_i} - a)^2 + (y_{P_i} - b)^2} + a \tag{C.12a}$$

$$y_{t_{i1,2}} = \frac{r_1^2(y_{P_i} - b) \mp r_1(x_{P_i} - a)\sqrt{(x_{P_i} - a)^2 + (y_{P_i} - b)^2 - r_1^2}}{(x_{P_i} - a)^2 + (y_{P_i} - b)^2} + b \tag{C.12b}$$

while $x_{t_{i3,4}}$ and $y_{t_{i3,4}}$ are obtained from Equation (C.12) by substitution a with c, b with d, and r_1 with r_2. So, the equations of the inner tangents are

$$L_{t_i1}: \quad y = \left(\frac{y_{t_{i3}} - y_{t_{i1}}}{x_{t_{i3}} - x_{t_{i1}}}\right) x + \left(y_{t_{i3}} \left(\frac{y_{t_{i3}} - y_{t_{i1}}}{x_{t_{i3}} - x_{t_{i1}}}\right)\right) x_{t_{i3}} \tag{C.13a}$$

$$L_{t_i2}: \quad y = \left(\frac{y_{t_{i4}} - y_{t_{i2}}}{x_{t_{i4}} - x_{t_{i2}}}\right) x + \left(y_{t_{i4}} \left(\frac{y_{t_{i4}} - y_{t_{i2}}}{x_{t_{i4}} - x_{t_{i2}}}\right)\right) x_{t_{i4}} \tag{C.13b}$$

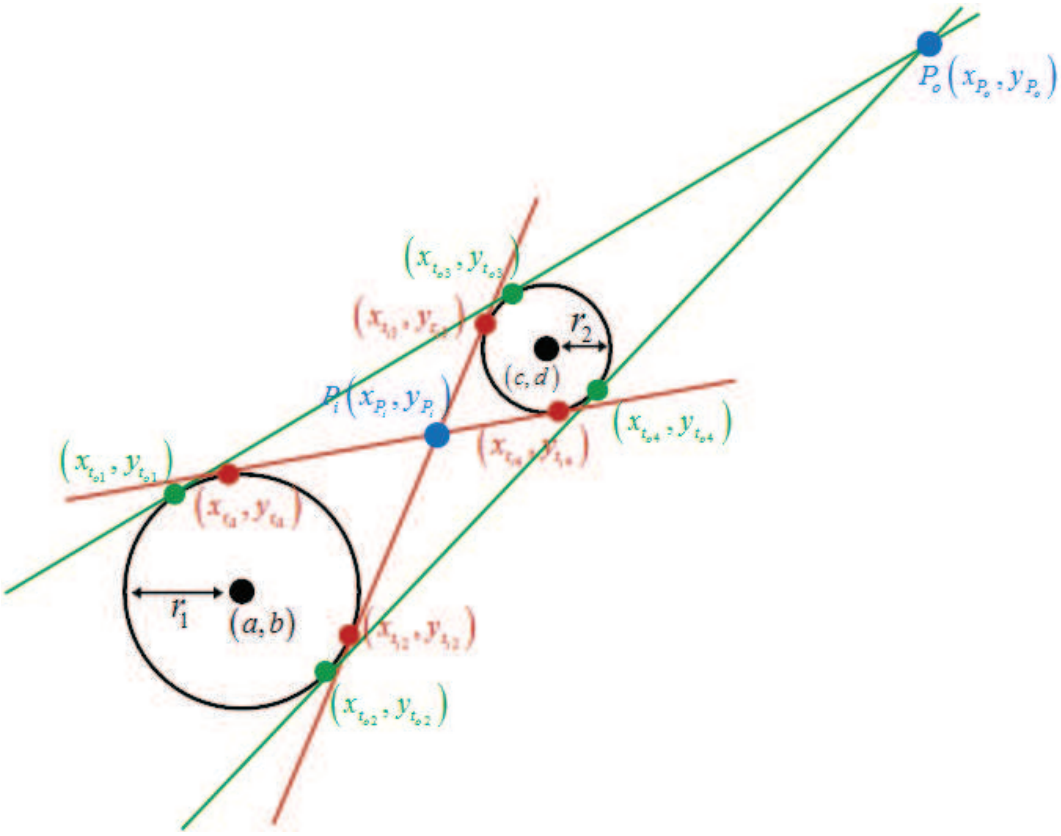

FIGURE C.1: Common Tangents to Two Circles ($r_1 > r_2$).

Note that when $r_1 = r_2$ the Equation (C.9) can not be used to compute tangential points due to the non existance of the intersection point P_o. In fact, in this case, the outer

tangents becomes parallel to each other as well as to the line passes by (a, b) and (c, d) as it is illustrated in Figure (C.2).

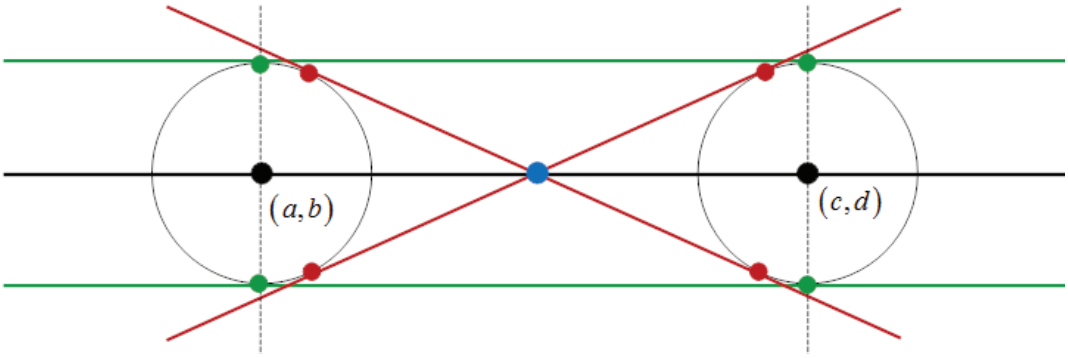

FIGURE C.2: Common Tangents to Two Circles ($r_1 = r_2$).

Let L_C denotes the line passing through the center of C_1 and C_2. Then, the tangential points are the intersections between C_1/C_2 and the perpendicular lines to L_C at (a, b) and (c, d) respectively.

Bibliography

[1] D. Aarno, D. Kragic, and H.I. Christensen. Artificial potential biased probabilistic roadmap method. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 461–466, New Orleans, LA, USA, April 2004.

[2] M. Alighanbari, Y. Kuwata, and J.P. How. Coordination and control of multiple uavs with timing constraints and loitering. In *American Control Conference*, volume 6, pages 5311–5316, Denver, Colorado, USA, June 2003.

[3] N.M. Amato, O.B. Bayazit, L.K. Dale, C. Jones, and D. Vallejo. Obprm: An obstacle-based prm for 3d workspaces. In *Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, Houston, Texas, USA, March 1998.

[4] G. Ambrosino, M. Ariola, U. Ciniglio, F. Corraro, E. De Lellis, and A. Pironti. Path generation and tracking in 3-d for uavs. *IEEE Transactions on Control Systems Technology*, 17(4):980–988, July 2009.

[5] E.P. Anderson and R.W. Beard. An algorithmic implementation of constrained extremal control for uavs. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Monterey, California, USA, August 2002.

[6] Z. Artstein. Stabilization with relaxed controls. *Nonlinear Analysis: Theory, Methods & Applications*, 7(11):1163–1173, November 1983.

[7] Z. Ashraf and M. A. Choudhry. Dynamic modeling of the airship using analytical aerodynamic model. In *International Conference on Emerging Technologies*, pages 188–193, Islamabad, Pakistan, 19-20 October 2009.

[8] J. R. Azinheira, P. Rives, J. R. H. Carvalho, G. F. Silveira, E. C. De Paiva, and S. S. Bueno. Visual servo control for the hovering of an outdoor robotic airship. In *IEEE International Conference on Robotics and Automation*, pages 2787–2792, Washington, DC, USA, May 2002.

[9] L. Bairstow. *Applied Aerodynamics*. Green and Co., London, 1920.

[10] E. Bakolas and P. Tsiotras. Time-optimal synthesis for the zermelo-markov-dubins problem: The constant wind case. In *American Control Conference*, pages 6161–6168, Marriott Waterfront, Baltimore, MD, USA, June–July 2010.

[11] H. Bang, S. Lee, and H. Lee. Nonlinear trajectory tracking using vectorial backsteeping approach. In *International Conference on Control, Automation and Systems*, pages 169–174, COEX, Seoul, Korea, October 2008.

[12] A. Barrientos, P. Gutierrez, and J. Colorado. *Advanced UAV Trajectory Generation: Planning and Guidance, Aerial Vehicles, T. M. Lam (Ed.)*. InTech, 2009.

[13] L. Beji and A. Abichou. Tracking control of trim trajectories of blimp for ascent and descent flight maneuvers. *International Journal of Control*, 78(10):706–719, July 2005.

[14] L. Beji, A. Abichou, and Y. Bestaoui. Stabilization of a nonlinear underactuated autonomous airship - a combined averaging and backstepping approach. In *the 3rd International Workshop on Robot Motion and Control (RoMoCo)*, Bukowy Dworek, Poland, November 2002.

[15] J. Bellingham, M. Tillerson, A. Richards, and J.P. How. *Multi-task allocation and path planning for cooperation UAVs*. In: Cooperative Control: Models, Applications and Algorithms, S. Butenko, R. Murphey, and P.M. Pardalos (Eds.). Kluwer Academic Publishers, Netherlands, 2003.

[16] M. Bellmore and G.L. G. L. Nemhauser. The traveling salesman problem: A survey. *Operations Research*, 59(3):538–558, May 2011.

[17] Y. Bestaoui. *Lighter Than Air Robots*. Number 58 in Intelligent Systems, Control and Automation: Science and Engineering. Springer, Netherlands, 2012.

[18] Y. Bestaoui and S. Hima. Trajectory tracking of a dirigible in a high constant altitude flight. In *the 5th IFAC symposium on Nonlinear control systems*, Saint Petersburg, Russia, July 2001.

[19] Y. Bestaoui and E. Kahale. Analysis of time optimal 3d paths for an autonomous aircraft with a piecewise constant acceleration. In *49th AIAA Aerospace Sciences Meeting*, Orlando Florida, USA, January 2011.

[20] Y. Bestaoui and E. Kahale. Generation of time optimal trajectories of an autonomous airship. In *IEEE Workshop on Robot Motion Control (ROMOCO)*, Bukowy Dworek, Poland, June 2011.

[21] Y. Bestaoui and E. Kahale. Time optimal trajectories of a lighter than air robot with second order constraints and a piecewise constant velocity wind. *AIAA Journal of Aerospace Computing Information and Communication*, 10(4):155–171, April 2013.

[22] S. J. Biljlsma. Optimal aircraft routing in general wind fields. *AIAA Journal of Guidance, Control and Dynamics*, 32(3):1025–1029, May–June 2009.

[23] A. M. Bloch. *Nonholonomic mechanics and Control*. Springer, 2003.

[24] B. Blocken, P. Moonen, T. Stathopoulos, F. Asce, and J. Carmeliet. Numerical study on the existence of the venturi effect in passages between perpendicular buildings. *Journal of Engineering Mechanics*, 134(12):1021—-1028, December 2008.

[25] R. Bohlin and L. Kavraki. Path planning using lazy prm. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 521–528, San Francisco, CA, USA, April 2000.

[26] A. Bonnet. *Identification des coefficients aerodynamiques du dirigeable AS500 du LAAS*. Technical report. Étude Hydro-Aérodynamique, SUPAERO, 2003.

[27] V. Boor, M.H. Overmars, and A.F. Van der Stappen. The gaussian sampling strategy for probabilistic roadmap planners. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1018–1023, Detroit, Michigan, USA, May 1999.

[28] D. Boukraa, Y. Bestaoui, and N. Azouz. Three dimensional trajectory generation for an autonomous plane. *International Review of Aerospace Engineering*, 1(4):355–365, August 2008.

[29] E. Bryson Jr. and Y.C. Ho. *Applied Optimal Control*. Taylor & Francis, 1975.

[30] C. Büskens and H. Maurer. Sqp-methods for solving optimal control problems with control and state constraints: Adjoint variables, sensitivity analysis and real-time control. *Journal of Computational and Applied Mathematics*, 120(1–2):85–108, August 2000.

[31] C. Canudas de Wit, B. Siciliano, and G. (Eds.) Bastin. *Theory of Robot Control*. Springer-Verlag, London, 1996.

[32] P. Castillo, R. Lozano, and A. Dzul. Stabilization of a mini rotorcraft with four rotors. *IEEE Control Systems Magazine*, 25(6):45–55, December 2005.

[33] P. Castillo, R. Lozano, and A. E. Dzul. *Modeling and Control of Mini-Flying Machines*. Springer-Verlag, London, 2005.

[34] H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki, and S. Thrum. *Principles of Robot Motion, Theory, Algorithms and Implementation*. MIT Press, Cambridge, 2005.

[35] N. Christofides. *Worst case analysis of a new heuristic for the traveling salesman problem*. Report 388, Graduate School of Industrial Administration, Carniege-Mellon University, Pittsburg, PA, USA, 1976.

[36] G. Clark and J.W. Wright. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4):568–581, July 1964.

[37] C.I. Connolly, K. Souccar, and R.A. Grupen. A hamiltonian framework for kino-dynamic planning and control. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2746–2751, Nagoya, Japan, May 1995.

[38] M.V. Cook. *Flight Dynamics Principles*. Elsevier, UK, 2007.

[39] J.-F. Cordeau, M. Gendreau, G. Laporte, J.-Y. Potvin, and F. Semet. A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, 53(5):512–522, May 2002.

[40] C.E. Corrigan, G.C. Roberts, M.V. Ramana, D. Kim, and V. Ramanathan. Capturing vertical profiles of aerosols and black carbon over the indian ocean using autonomous unmanned aerial vehicles. *Journal of Atmospheric Chemistry and Physics*, 8(3):737–747, 2008.

[41] J.A. Curry, J. Maslaink, G. Holland, and J. Pinto. Applications of aerosondes in the arctic. *Bull. Amer. Meteor. Soc.*, 85(12):1855–1861, 2004.

[42] L.K. Dale and Amato N.M. Probabilistic roadmaps - putting it all together. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 1940–1947, Seoul, Korea, May 2001.

[43] G.B. Dantzig and R. Ramser. The truck dispatching problem. *Management Science*, 6(1):80–91, October 1959.

[44] L. De Filippis and G. Guglieri. *Advanced Graph Search Algorithms for Path Planning of Flight Vehicles, Recent Advances in Aircraft Technology, R. K. Agarwal (Ed.)*. InTech, 2012.

[45] E. De Pavia, F. Benjovengo, and S. Bueno. Sliding mode control for the path following of an unmanned airship. In *the 6th IFAC Symposium on Intelligent Autonomous Vehicles*, Toulouse, France, September 2007.

[46] F. Derkx, J. Dumoulin, J.-L. Sorin, and V. Legeay. Concept de plate-forme mobile instrumentée (pmi) pour l'inspection des ouvrages d'art. In *journées Micro-Drones 2-ème édition*, Toulouse, France, September 2002.

[47] F. Derkx and F.L. Sorin. Inspection des ouvrages d'art par drone. bilan et prespectives des travaux du lcpc. *Bulletin des laboratoires des ponts et chausses*, (273):39–56, December 2008.

[48] C. Dever, B. Mettler, E. Feron, J. Popovic, and M. McConley. Trajectory interpolation for parametrized maneuvering and flexible motion planning of autonomous vehicles. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, RI, USA, August 2004.

[49] S. Dicheva and Y. Bestaoui. 3d waypoint generation in a dynamic environment for an airborne launch mission. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 10(226):1283–1297, October 2012.

[50] M. Dorigo and T. Stutzle. *Ant Colony Optimization*. Prentice Hall, 2004.

[51] L. E. Dubins. On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–517, 1957.

[52] C. Eberhart and J. Kennedy. A new optimizer using particle swarm optimization. In *International Symposium on Micro Machine and Human Science*, pages 39–43, Nagoya, Japan, October 1995.

[53] A.E. Eiben and J.E. Smith. *Introduction to Evolutionary Computing*. Springer-Verlag, 2003.

[54] A. Elfes, J. F. Montgomery, J. L. Hall, S. S. Joshi, J. Payne, and Bergh C. F. Autonomous flight control for a planetary exploration aerobot. In *AIAA Space Conference*, Long Beach, CA, August 2005.

[55] J. Etele. Overview of wind gust modelling with appli- cation to autonomous low-level uav control. In *Mechanical and Aerospace Engineering Departement*, Carelton University, Ottawa, Canada, 2006.

[56] B. Etkin. Turbulent wind and its effect on flight. *Journal of Aircraft*, 18(5):327–345, May 1981.

[57] M. Faied, A. Mostafa, and A. Gurard. Vehicle routing problem instances: Application to multi-uav mission planning. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Toronto, Ontario, Canada, August 2010.

[58] B.A. Fisher and R. Jaikumar. A generalized assignment heuristic for vehicle routing. *Networks*, 11(2):109–124, 1981.

[59] T. I. Fossen. *Guidance and Control of Ocean Vehicles*. John Wiley & Sons Ltd., 1994.

[60] E. Frazzoli, M.A. Dahleh, and E. Feron. A hybrid control architecture for aggressive maneuvering of autonomous helicopters. In *IEEE International Conference on Decision and Control*, volume 25, pages 2471–2476, Phoenix, AZ, USA, December 1999.

[61] E. Frazzoli, M.A. Dahleh, and E. Feron. Real-time motion planning for agile autonomous vehicles. *AIAA Journal of Guidance, Control, and Dynamics*, 25(1):116—-129, January 2002.

[62] R. A. Freemann and P. V. Kokotović. *Robust Nonlinear Control Design: State-Space and Lyapunov Techniques*. Birkhauser, 1996.

[63] E.W. Frew and T.X. Brown. Networking issues for small unmanned aircraft systems. *Journal of Intelligent and Robotic Systems*, 54(1–3):21–37, March 2009.

[64] T. Fukao, K. Fujitaniy, and T. Kanade. Image-based tracking control of a blimp. In *the 42nd IEEE Conference on Decision and Control*, pages 3462–3467, Hawaii, USA, December 2003.

[65] T. Fukao, T. Kanzawa, and K. Osuka. Inverse optimal tracking control of an aerial blimp robot. In *the 5th International Workshop on Robot Motion and Control (RoMoCo)*, Poland, June 2005.

[66] H. Fukushima, R. Saito, and F. Matsumo. Model predictive control of an autonomous blimp with input and output constraints. In *IEEE International Conference on Control Applications*, pages 2184–2189, Munich, Germany, October 2006.

[67] T.J. Gaskell. Bases for vehicle fleet scheduling. *Journal of the Operational Research Society*, 18(3):281–295, September 1967.

[68] B. Gillett and L.R. Miller. A heuristic algorithm for the vehicle-dispatch problem. *Operations Research*, 22(2):340–349, Mars 1974.

[69] C. Goerzen, Z. Kong, and B. Mettler. A survey of motion planning algorithms from the perspective of autonomous uav guidance. In KimonP. Valavanis, Randal Beard, Paul Oh, Aníbal Ollero, LeslieA. Piegl, and Hyunchui Shim, editors, *Selected papers from the 2nd International Symposium on UAVs, Reno, Nevada, U.S.A. June 8–10, 2009*, pages 65–100. Springer Netherlands, 2010.

[70] J.A. Guerrero, , and Y. Bestaoui. Uav path planning for structure inspection in windy environments. *Journal of Intelligent & Robotic Systems*, 69(1-4):297–311, January 2013.

[71] J.A. Guerrero and Y. Bestaoui. *Optimal Trajectory Planning for Structure Inspection.* Technical report. Laboratoire IBISC, Université d'Évry Val d'Essonne, Évry, France, December 2011.

[72] V. Guillemin and A. Pollack. *Differential Topology.* Prentice-Hall, New York, USA, 1974.

[73] J. Hacker and B. H. Kroplin. An experimental study of visual flight trajectory tracking and pose prediction for the automatic computer control of a miniature airship. In *SPIE International Society for Optical Engineering*, pages 25–36, 2003.

[74] P. Hart, N. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, July 1968.

[75] S. Haykin. *Neural Networks.* Prentice Hall, 1999.

[76] M.T. Heath. *Scientific Computing: An Introductory Survey, 2nd Ed.* McGraw-Hill, New York, 2002.

[77] S. Hima. *Planification des trajectoires d'un dirigeable autonome.* PhD. Thesis. Université d'Évry Val d'Essonne, Évry, France, 2005.

[78] S. Hota and D. Ghose. Optimal path planning for an aerial vehicle in 3d space. In *the 49th IEEE Conference on Decision and Control*, pages 4902–4907, Atlanta, Georgia, USA, December 2010.

[79] D. Hsu, T. Jiang, J.H. Reif, and Z. Sun. The bridge test for sampling narrow passages with probabilistic roadmap planners. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4420–4426, Taipei, Taiwan, September 2003.

[80] D. Hsu, L.E. Kavraki, J.-C. Latombe, R. Motwani, and S. Sorkin. On finding narrow passages with probabilistic roadmap planners. In *Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, Houston, Texas, USA, March 1998.

[81] D. G. Hull. *Fundamentals of Airplane Flight Mechanics.* Springer, Berlin, 2007.

[82] F. Imado, Y. Heike, and T. Kinoshita. Research on a new aircraft point-mass model. *Journal of Aircraft*, 48(4):1121–1130, July-August 2011.

[83] M. R. Jardin and A. E. Bryson. Neighboring optimal aircraft guidance in winds. *AIAA Journal of Guidance, Control and Dynamics*, 24(4):710–715, July–August 2001.

[84] M.R. Jardin and E. Bryson Jr. Methods for computing minimum-time paths in strong winds. *Journal of Guidance, Control, and Dyncamics*, 35(1):165–171, January–February 2012.

[85] A. L. Jennings, R. Ordonez, and N. Ceccarelli. Dynamic programming applied to uav way point path planning in wind. In *IEEE International Symposium on Computer-Aided Control System Design*, pages 215–220, San Antonio, Texas, USA, Septembre 2008.

[86] R. Jia, M. T. Frye, and C. Qian. Control of an airship using partical swarm optimization and neural network. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 1809–1814, San Antonio, TX, USA, October 2009.

[87] Z. Jiang and R. Ordonez. Robust approach and landing trajectory generation for reusable launch vehicles in winds. In *IEEE International Conference on Control Applications*, pages 930–935, San Antonio, Texas, USA, Septembre 2008.

[88] V. Jurdjevic. *Geometric control theory.* Cambridge University Press, Cambridge, U.K., 2008.

[89] E. Kahale, Y. Bestaoui, and P. Castillo. Path tracking of a small autonomous airplane in wind gusts. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 402–407, Vilamoura, Algarve, Portugal, October 2012.

[90] E. Kahale, P. Castillo, and Y. Bestaoui. Autonomous path tracking of a kinematic airship in presence of unknown gust. In *International Conference on Unmanned Air- craft Systems (ICUAS)*, Philadelphia, PA, USA, June 2012.

[91] E. Kahale, P. Castillo Garcia, and Y. Bestaoui. Autonomous path tracking of a kinematic airship in presence of unknown gust. *Journal of Intelligent and Robotic Systems*, 69(1-4):431–446, January 2013.

[92] F Kamrani and R. Ayani. *UAV Path Planning in Search Operations, Aerial Vehicles, T. M. Lam (Ed.).* InTech, 2009.

[93] S. Karaman and E. Frazzoli. Vehicle routing problem with metric temporal logic specifications. In *the 47th IEEE Conference on Decision and Control*, pages 3953–3958, Cancun, Mexico, December 2008.

[94] L. Kavraki, P. Svestka, J.-C. Latombe, and M.H. Overmars. Probabilistic roadmaps for path planning in high dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, August 1996.

[95] Y. Kawai, S. Kitagawa, S. Izoe, and M. Fujita. An unmanned planar blimp on visualfeedback control: Experimental results. In *the 42nd SICE Annual Conference*, pages 680–685, 2003.

[96] H.J. Kelley. Guidance theory and extremal fields. *IRE Transactions on Automatic Control*, 7(5):75–82, October 1962.

[97] O. Khatib and L.M. Mampey. *Fonction Decision-Commande d'un Robot Manipulateur*. Technical report 2/7156. DERA/CERT, Toulouse, France, 1978.

[98] G. A. Khoury and J. D. Gillet. *Airship Technology*. Cambridge University Press, Cambridge, 1999.

[99] J. Kim, J. Keller, and V. Kumar. Design and verification of controllers for airships. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 54–60, Las Vegas, Nevada, USA, October 2003.

[100] S. Kirkpatrik. Optimization by simulated annealing: Quantitative study. *Journal of Statistical Physics*, 34(5-6):975–986, March 1984.

[101] P. Klein. *Lecture notes on Hilbert spaces and the projection theorem*. Stockholms Universitet, 2009.

[102] E. Kreyszig. *Differential Geometry*. Dover Publications, New York, 1991.

[103] S.N. Kumar and R. Panneerselvam. A survey on the vehicle routing problem and its variants. *Intelligent Information Management*, 4(3):66–74, May 2012.

[104] H. Kurniawati and D. Hsu. Workspace importance sampling for probabilistic roadmap planning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1618–1623, Sendai, Japan, September 28–October 2 2004.

[105] T. Kämpke and A. Elfes. Optimal aerobot trajectory planning for wind based opportunity flight control. In *IEEE/RSI International Conference on Intelligent Robots and Systems*, pages 67–74, Las Vegas, Nevada, USA, October 2003.

[106] G. Lafferriere and H. J. Sussmann. A differential geometric approach to motion planning. In *NONHOLONOMIC MOTION PLANNING*, pages 235–270. Kluwer, 1993.

[107] G. Laporte. The traveling salesman problem: An overview of exact and approximate algoritgms. *European journal of Operational Research*, 59(2):231–247, June 1992.

[108] G. Laporte and Y. Norbert. *Exact Algorithms for the Vehicle Routing Problem*. Surveys in Combinatorial Optimization. ElSEVIER, Amsterdam, 1987.

[109] R.C. Larson and A.R. Odoni. *Urban Operations Research*. Prentice-Hall, Englewood Cliffs,NJ, USA, 1981.

[110] J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Press, Massachusetts, USA, 1991.

[111] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006.

[112] S.M. LaValle. *Rapidly-exploring random trees: a new tool for path planning*. Technical report (TR 98–11). Computer Science Department, Iowa State University, USA, 1998.

[113] S.M. LaValle and J.J. Kuffner. Rapidly-exploring random trees: Progress and prospects. In *Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, Hanover, NH, USA, March 2000.

[114] S.M. LaValle and J.J. Kuffner. Randomized kinodynamic planning. *International Journal of Robotics Research*, 20(5):378–400, May 2001.

[115] W.S. Levine. *The Control Handbook, Second Edition : Control System, Advenced Methods*. CRC Press, Taylor & Francis Group, 2011.

[116] Y. Li and M. Nahon. Modeling and simulation of airship dynamics. *Journal of Guidance, Control, and Dynamics*, 30(6):1691–1700, November–December 2007.

[117] Z. Li and J. F. (Ed.) Canny. *Non holonomic Motion Planning*. Kluwer Academic Press, Berlin, 1992.

[118] Y. Liu, Z. Pan, D. Stirling, and F. Naghdy. Control of autonomous airship. In *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 2457–2462, Guilin, China, December 2009.

[119] A. M. Lyapunov. Problème général de la stabilité du mouvement. *Annales de la faculté des sciences de Toulouse*, 9(2):203–474, 1907.

[120] T. McGee and J. K. Hedrick. Optimal path planning with a kinematic airplane model. *AIAA Journal of Guidance, Control and Dynamics*, 30(2):629–633, March–April 2007.

[121] D. Mclean. *Automatic Flight Control Systems*. Prentice Hall, UK, 1990.

[122] M. Morales, S. Rodríguez, and N.M. Amato. Improving the connectivity of prm roadmaps. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4427–4432, Taipei, Taiwan, September 2003.

[123] P. Morin and C. Samson. *Trajectory Tracking for Non-holonomic Vehicles, Robot Motion and Control, K. R. Kozlowski (Ed.)*. Springer-Verlag, London, 2006.

[124] A. Moutinho and J. R. Azinheira. Stability and robustness analysis of the aurora airship control system using dynamic inversion. In *IEEE International Conference on Robotics and Automation*, pages 2265–2270, Washington D. C., USA, April 2005.

[125] J. Mueller, Y. Zhoa, and M. Paluszek. Development of an aerodynamic model and control law design for a high altitude airship. In *AIAA Unmanned Unlimited Conference*, Chicago, USA, September 2004.

[126] J. B. Mueller, Y. J. Zhao, and W. L. Garrard. Optimal ascent trajectories for stratospheric airships using wind energy. *Journal of Guidance, Control, and Dynamics*, 32(4):1232–1245, July 2009.

[127] L. E. Munoz. *Stabilisation d'un véhicule aérien autonome en présence de vent*. PhD. Thesis. Université de Technologie de Compiègne, Compiègne, France, 2012.

[128] L. E. Munoz, O. Santos, and P. Castillo. Robust nonlinear real-time control strategy to stabilize a pvtol aircraft in crosswind. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1606–1611, Taipei, Taiwan, October 2010.

[129] G. Murguia-Rendon, H. Rodriguez-Cortes, and M. Velasco-Villa. Trajectory tracking control for the planar dynamics of a thrust vectored airship. In *the 52nd IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 329–332, Cancun, Mexico, August 2009.

[130] D.S. Naidu. *Optimal Control Systems*. CRC Press LLC, 2003.

[131] R. Nelson, B. Barber, T. McLain, and R. Beard. Vector field path following for miniature air vehicle. *IEEE Transactions on Robotics*, 23(3):519–529, June 2007.

[132] R. C. Nelson. *Flight Stability and Automatic Control*. McGraw-Hill, 1989.

[133] H. Nijmeijer and A. Van Der Schaft. *Nonlinear Dynamical Control Systems*. Springer-Verlag, 1990.

[134] K. Nonami, F. Kendoul, S; Suzuki, W. Wang, and D. Nakazawa. *Autonomous Flying Robots: Unmanned Aerial Vehicles and Micro Aerial Vehicles*. Springer, 2010.

[135] H. Paessens. The savings algorithm for the vehicle routing problem. *European journal of Operational Research*, 34(3):336–344, March 1988.

[136] C. S. Park, H. Lee, M. J. Tahk, and H. Bang. Airship control using neural network augmented model inversion. In *IEEE Conference on Control Applications*, pages 558–563, Istanbul, Turkey, June 2003.

[137] J. Rao, Z. Gong, J. Luo, and Xie S. A flight control and navigation system of a small size unmanned airship. In *IEEE International Conference on Mechatronics and Automation*, pages 1491–1496, Niagara Falls, Canada, July 2005.

[138] J. Rao, Z. Gong, J. Luo, and Xie S. Unmanned airships for emergency management. In *IEEE International Workshop on Safety, Security and Rescue Robotics*, pages 125–130, Kobe, Japan, June 2005.

[139] F. Repoulias and E. Papadopoulos. Robotic airship trajectory tracking control using a backstepping methodology. In *IEEE International Conference on Robotics and Automation*, pages 188–193, Pasadena, CA, May 2008.

[140] A. Richards, Y. Kuawta, and J.P. How. Experimental demonstrations of real-time milp control. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Austin, Texas, USA, August 2003.

[141] E. Rimon and D.E. Koditschek. Exact robot navigation using cost functions: the case of distinct spherical boundaries in e^n. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 1791–1796, Philadelphia, PA, USA, April 1988.

[142] J. Roskam. *Airplane Flight Dynamics and Automatic Flight Controls Part I*. DAR-corporation, USA, 2001.

[143] R. Rysdyk. Course and heading changes in significant wind. *AIAA Journal of Guidance, Control and Dynamics*, 30(4):1168–1171, July–August 2007.

[144] K. Savla, E. Frazzoli, and F. Bullo. Traveling salsperson problems for dubins vehicle. *IEEE Transactions on Automatic Control*, 53(6):1378–1391, July 2008.

[145] T. Schouwenaars, B. Mettler, E. Feron, and J. How. Hybrid model for trajectory planning of agile autonomous vehicles. *Journal of Aerospace Computing, Information, and Communication*, 1(12):629–651, December 2004.

[146] C. Schumacher, P. Chandler, M. Pachter, and L. Pachter. Uav task assignment with timing constraints via mixed-integer linear programming. In *AIAA 3rd Unmanned Unlimited Technical Conference, Workshop and Exhibit*, Chicago, Illinois, USA, September 2004.

[147] C. Schumacher, P. Chandler, M. Pachter, and L. Pachter. Optimization of air vehicles operations using mixed-integer linear programming. *Journal of Operational Research Society*, 58:516–527, April 2007.

[148] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: Modelling, Planning and Control*. Springer Verlag, London, 2009.

[149] G. F. Silveira, J. R. Azinheira, P. Rives, and Bueno S. S. Line following visual servoing for aerial robots combined with complementary sensors. In *the 11th International Conference on Advanced Robotics*, Coimbra, Portugal, June 2003.

[150] G. F. Silveira, J. R. H. Carvalho, P. Rives, J. R. Azinheira, S. S. Bueno, and M. K. Madrid. Optimal visual servoed guidance of outdoor autonomous robotic airships. In *American Control Conference*, pages 779–784, Anchorage, AK, USA, May 2002.

[151] T. Siméon, J.P. Laumond, and C. Nissoux. Visibility based probabilistic roadmaps for motion planning. *Advanced Robotics Journal*, 14(6):477–493, December 2000.

[152] S.N. Singh, M.L. Steinberg, and A.B. Page. Nonlinear adaptive and sliding mode flight path control of f/a-18 model. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1250–1262, 2003.

[153] L. Solaque, Z. Pinzon, and M. Duque. Nonlinear control of the airship cruise flight phase with dynamical decoupling. In *Electronics, Robotics and Automotive Mechanics Conference (CERMA)*, pages 472–477, Morelos, Mexico, October 2008.

[154] E. D. Sontag. A lyapunov-like characterization of asymptotic controllability. *SIAM Journal on Control and Optimization*, 21(3):462–471, May 1983.

[155] B. L. Stevens and F. L. Lewis. *Aircraft Control and Simulation*. Wiley, Canada, 2003.

[156] S. Subchan and R. Zbikowski. *Computational Optimal Control Tools and Practice*. John Wiley & Sons Inc., 2009.

[157] K. Sundar and S. Rathinam. Algorithms for routing an unmanned aerial vehicle in the presence of refueling depots. *IEEE Transactions on Automation Science and Engineering*, 11(1):287–294, January 2014.

[158] H. J. Sussmann. A general theorem on local controllability. *SIAM Journal on Control and Optimization*, 1(25):158–194, 1987.

[159] L. Techy. Optimal navigation in planar time-varying flow: Zermelo's problem revisited. *Intelligent Service Robotics*, 4(4):271–283, October 2011.

[160] P. G. Thomasson. Equations of motion of a vehicle in a moving fluid. *Journal of Aircraft*, 37(4):630–639, July–August 2000.

[161] P. Toth and D. Vigo. *The Vehicle Routing Problem*. SIAM, Philadelphia, USA, 2002.

[162] K.P. Valavanis. *Advances in Unmanned Aerial Vehicles: State of the Art and the Road to Autonomy*. Number 33 in Intelligent Systems, Control and Automation: Science and Engineering. Springer, Netherlands, 2007.

[163] S. Van Der Zwaan, M. Perrone, A. Bernardino, and J. Santos-Victor. Control of an aerial blimp based on visual input. In *the 8th International Symposium on Intelligent Robotic Systems*, Reading, UK, July 2000.

[164] G.B. Venturi. *Recherches expérimentales sur le principe de la communication latérale du mouvement dans les fluides, appliqué à l'explication de différens phénomènes hydrauliques*. Houel et Ducros: T. Barrois, Paris, France, 1797.

[165] N. X. Vinh. *Flight Mechanics of High-Performance Aircraft*. Number 4 in Cambridge aerospace series. Cambridge University Press, Cambridge, 1993.

[166] A. Weinstein and C. Schumacher. *UAV Scheduling vie the Vehicle Routing Problem with Time Windows*. Technical report, AFRL-VA-WP-TP-2007-306. Air Force Research Laboratory, USA, 2007.

[167] D. A. Wimmer, M. Bildstein, K. H. Well, M. Schlenker, P. Kungl, and B. H. Kröplin. Development and operation controllers for autonomous flight phases. In *Workshop on Aerial Robotics, IEEE International Conference on Intelligent Robots and Systems*, pages 55–68, Lausanne, Switzerland, October 2002.

[168] H. Wong, V. Kapila, and R. Vaidyanathan. Uav optimal path planning using c-c-c class paths for target touring. In *the 43rd IEEE Conference on Decision and Control*, pages 1105–1110, Nassau, Bahamas, December 2004.

[169] A. Wren and A. Holliday. Computer scheduling of vehicles from one or more depots to a number of delivery points. *Journal of the Operational Research Society*, 23(3):333–344, September 1972.

[170] X. Wu, C. H. Moog, L. A. M. Martinez, and Y. Hu. Nonlinear control of a buoyancy-driven airship. In *the 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, pages 2849–2854, Shanghai, China, December 2009.

[171] G. Xia and D. R. Corbett. Cooperative control systems of searching targets using unmanned blimps. In *the 5th Worth Congress on Intelligent Control and Automation*, pages 1179–1183, Hangzhou, P.R. China, June 2004.

[172] G. Yang and V. Kapila. Optimal path planning for unmanned air vehicles with kinematic and tactical constraints. In *the 41st IEEE Conference on Decision and Control*, volume 2, pages 1301–1306, Las Vegas, Nevada, USA, December 2002.

[173] M. I. Zelikin and V. F. Borison. *Theory of Chattering Control*. Birkhäuser, Basel, 1994.

[174] E. Zermelo. Über das navigationsproblem bei ruhender oder veränderlicher windverteilung. *Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 11(2):114–124, 1931.

[175] Y. Zhang, W. D. Qu, Y. G. Xi, and Z. L. Cai. Adaptive stabilization and trajectory tracking of airship with neutral buoyancy. *Acta Automatica Sinica*, 34(11):1437–1440, November 2008.

[176] A. S. I. Zinober. *Variable Structure and Lyapunov Control*. Springer-Verlag, London, 1994.