Perpetual Flight in Flow Fields

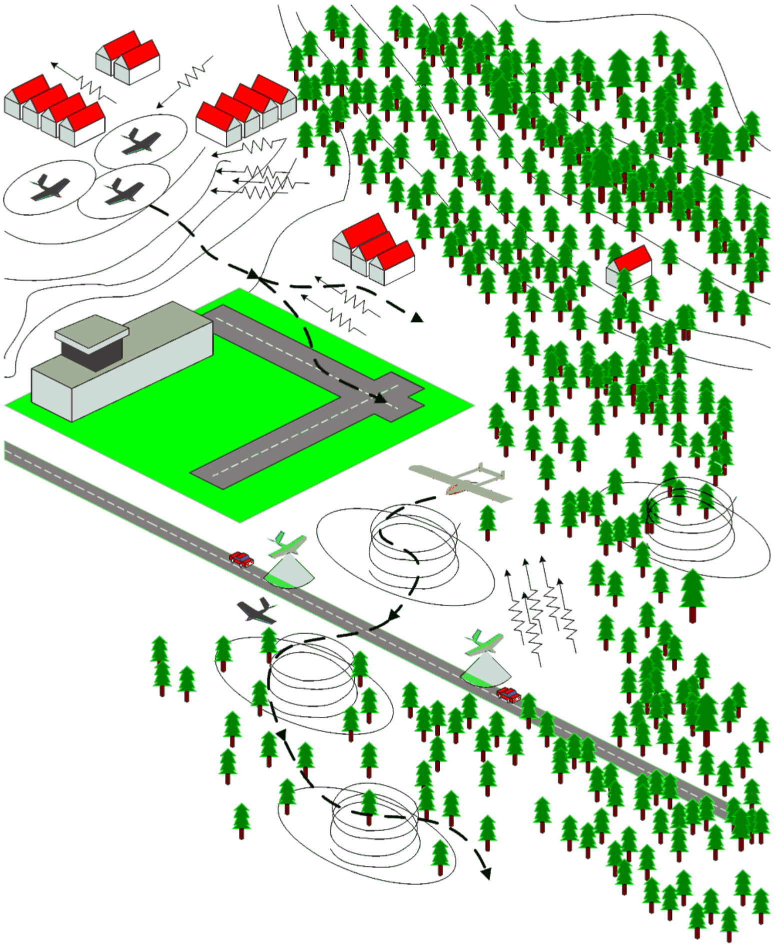
by

Ricardo Ayres Gomes Bencatel

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical and Computer Engineering)
in the University of Porto, School of Engineering
Faculdade de Engenharia da Universidade do Porto
2011

Doctoral Committee:

Professor Fernando Lobo Pereira, Chair
Professor Anouck Girard
Associate Luísa Bastos
Doctor José Morgado



PROGRAMA OPERACIONAL POTENCIAL HUMANO

QUADRO DE REFERÊNCIA ESTRATÉGICO NACIONAL
PORTUGAL 2007.2013

UNIÃO EUROPEIA
Fundo Social Europeu

To my family and my girlfriend, who supported my endeavors and endured the necessary absences.

ACKNOWLEDGEMENTS

I thank with all my heart my family and my girlfriend for the support in the easy and the hard times, and for the encouragement to always do my best and surpass myself. Furthermore, I would like to thank the friendship of Nuno Costa with whom I developed and tested several model aircraft, which contributed to the knowledge required for this work.

I would like to thank the support from AsasF group and the researchers from the Underwater Systems and Technology Laboratory, specially João Sousa and Gil Gonçálves, and Pedro Almeida for their support, and in particular João Correia, Joel Gomes, Eduardo Oliveira, Rui Caldeira, Filipe Ferreira, Bruno Terra, Filipe Costa Ferreira, Joel Cardoso, and Sérgio Ferreira for their friendship, support, and assistance.

I gratefully acknowledge the support of the Portuguese Air Force Academy, specially Cap. Elói Pereira, Lt. Tiago Oliveira, Cap. José Costa, Sgt. Joaquim Gomes, Sgt. Paulo Teixeira, Lt. Gonçalo Cruz, Lt. Col. Morgado, Maj. Madruga, Lt. João Caetano, and Sgt. Fernandes.

I'm most grateful for the time I spend at the Aerospace Department from the University of Michigan. I enjoyed everyone's friendliness and support, in particular Anouck Girard and all Aerospace & Robotic Controls Laboratory people, specially Baro Hyun, Justin Jackson, Mariam Faied, and Zahid Hasan.

The research leading to this work was partly funded by Financiamento pluri-anual of the R&D unit FEUP Instituto de Sistemas e Robótica - ISR - Porto. It was also funded by the Portuguese Foundation for Science and Technology (FCT - Fundação para a Ciência e a Tecnologia) under PhD grant SFRH/BD/40764/2007, co-financed by the Portuguese Human Potential Operational Program (POPH - Programa Operacional Potencial Humano) and the European Social Fund (FSE - Fundo Social Europeu).

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF ABBREVIATIONS

AOA Angle-of-Attack

AUV Autonomous Underwater Vehicle

ASV Autonomous Surface Vehicle

APF Adaptive Particle Filter

CAS Collision Avoidance System

CM Center of Mass

CPCAA Close Proximity Collision Avoidance Algorithm

DOF Degrees-of-Freedom

FCS Flight Control System

FOL First-Order Logic

GNSS Global Navigation Satellite System

GP Gaussian process

IC Internal Combustion

IMU Inertial Measurement Unit

NM nautical miles

PDA Path Deconflicting Algorithm

pdf probability density function

PL Propositional Logic

PF Particle Filter

RAPF Regularized Adaptive Particle Filter

RHC Receding Horizon Control

RPF Regularized Particle Filter

RPM Rotations Per Minute

TCAS Traffic Alert and Collision Avoidance System

TL Temporal Logic

UAV Unmanned Aerial Vehicle

VTOL Vertical Takeoff and Landing

ZOH Zero Order Hold

NOMENCLATURE

Vectors are represented by bold lowercase letters ($\mathbf{v}$).

Matrices are represented by bold uppercase letters ($\mathbf{M}$).

Scalars are represented by non-bold lowercase or uppercase letters (s or S).

Reference frame identification letters appear as a superscript on the right (s^R).

Other descriptions appear as a subscript on the right ($s_{d,i}$).

$\mathbf{x}_i$ - Position of UAV i

$\Delta\mathbf{x}_{ij}$ - Relative position between the two UAVs ($\mathbf{x}_j - \mathbf{x}_i$)

$\mathbf{v}_i$ - Velocity of UAV i

$\mathbf{v}_{cmd,i}$ - Commanded velocity of UAV i

$\Delta\mathbf{v}_{ij}$ - Relative velocity between the two UAVs ($\mathbf{v}_j - \mathbf{v}_i$)

g - Gravity acceleration

ρ - Air density

z_i - Mixing-layer thickness

w^* - Convective velocity scale

ABSTRACT

Perpetual Flight in Flow Fields
by
Ricardo Ayres Gomes Bencatel

Chair: Fernando Lobo Pereira

In this work we study the problem of flight in flow fields, i.e., with air flow currents and flow velocity variations. The main goal is to enable aircraft, and in particular UAVs, to perform very long flights and in the extreme perpetual flight using as energy source only the atmospheric air flow phenomena, also called flow field phenomena. That would allow the aircraft to fly without the need to land to refuel or recharge the propulsion batteries, depending on the propulsion system.

UAVs have become an important tool for military operations. There is also large potential for the use of UAVs in civilian applications. UAVs are specially useful for dull, dirty or dangerous missions. Large UAVs, as the Heron TP and the Global Hawk, have endurances greater than 36 hours. These endurances are due to the good airframe aerodynamic efficiency, but even more to the large fuel capacity. Small UAVs reduced form factor yields many advantages, as in terms of handling and required support facilities. Nevertheless, their reduced form factor also yields less aerodynamic efficiency and reduced fuel or battery capacity. This reduces small UAVs endurance.

Atmospheric flow field phenomena, such as thermal updrafts, wind shear, and gust, hold large amounts of energy. Aircraft may harvest some of this energy by executing static and dynamic soaring flight paths. These paths usually require good flight maneuverability. Most small UAVs present better maneuverability than larger UAVs and many other aircraft. Thereby, they are able to execute the required maneuvers to harvest energy from the flow field phenomena.

This work discusses the flow field phenomena characteristics, analyzes the necessary conditions to achieve perpetual flight with the different flow field phenomena and shows how these phenomena can be observed. We study and develop detailed flow field models and their interactions with the aircraft dynamics. For thermal updrafts we created a new *Bubble Thermal* model, including its movement and its interactions with the surrounding flow field. We also extended an existing *Chimney Thermal* model to include movement and interactions with the surrounding flow field. In terms of wind shear we created three models for the *Layer Wind Shear* phenomenon and a model for the *Ridge Wind Shear* phenomenon. These models serve as inputs to the inference methods used to localize and characterize the air flow phenomena.

Further, we analyze what is the required balance between each flow field phenomenon's characteristics and the aircraft aerodynamic characteristics to enable perpetual flight. The result is intuitive, showing that it is easier to perform perpetual flight with more efficient aircraft and when each phenomenon presents stronger effects, as an updraft or a flow gradient.

To enable the flow field energy exploitation, most control methods require an estimate of the exploited phenomenon parameters. We show that the flow field phenomena parameters are observable and derive methods to estimate these parameters. These methods are based on Particle Filters, coping well with the nonlinear nature of the phenomena models and the non-Gaussian probability distributions. The test results are very promising, showing good estimation performance and requiring low processing power.

We noticed a lack of experimental validation of existing phenomena models. As such, we developed control methods that would allow us to use UAVs to collect spatially distributed data to enable the validation of the phenomena models. We studied collision avoidance and formation flight methods and implemented them to allow safe environment sampling and also collaborative flight. The developed algorithms are based on hybrid systems and sliding mode control. The formation flight controller presented a good performance, with good path tracking and, even more importantly, maintaining a safe distance among the formation aircraft throughout the flight.

Voo Perpétuo com Correntes Atmosféricas
por
Ricardo Ayres Gomes Bencatel

Orientador: Fernando Lobo Pereira

Resumo

Este trabalho debruça-se sobre o problema do voo com correntes atmosféricas e com variações da velocidade das mesmas. O objectivo principal é criar a capacidade de fazer voos muito longos, e no limite perpétuos, com aviões, e em particular VANTs (Veículos Aéreos Não Tripulados - UAVs), usando como fonte de energia fenómenos atmosféricos. Tal capacidade permitiria que um avião voasse sem necessidade de aterrar, para reabastecer ou recarregar as baterias, dependendo do sistema de propulsão.

Os UAVs tornaram-se uma importante ferramenta para operações militares. Também existe um enorme potencial na aplicação de UAVs em aplicações civis. Os UAVs são especialmente úteis em missões entediantes, sujas ou perigosas. UAVs de grande porte, como o Heron TP e o Global Hawk, têm autonomias superiores 36 horas. Estas autonomias devem-se à boa eficiência aerodinâmica das aeronaves, mas ainda mais à enorme capacidade de combustível. Os UAVs pequenos oferecem muitas vantagens devido às suas dimensões reduzidas, tal como o seu manuseamento ou as instalações de suporte necessárias. No entanto, a dimensão reduzida também provoca a redução da eficiência aerodinâmica e reduz a capacidade para combustível ou baterias para a propulsão. Tal reduz a autonomia dos UAVs pequenos.

Os fenómenos de correntes atmosféricas, como as ascendentes térmicas, gradientes de vento e rajadas, contêm enormes quantidades de energia. Os aviões podem recolher alguma desta energia executando trajectos ao longo destes fenómenos, como voo planado estático e dinâmico. A execução destes trajectos normalmente requer que o avião tenha boa manobrabilidade. A maioria dos UAVs pequenos apresentam melhor manobrabilidade que os UAVs de grande porte e muitos outros aviões. Assim, a maioria dos UAVs pequenos são capazes de executar as manobras necessárias para recolher energia dos fenómenos de correntes atmosféricas.

Este trabalho discute as características dos fenómenos de correntes atmosféricas, analisa as condições necessárias para alcançar voo perpétuo com os diferentes fenómenos e mostra como estes fenómenos podem ser observados e estimados. Para tal, estudamos e desenvolvemos modelos detalhados para os fenómenos em foco e a sua interacção com a dinâmica dos aviões. Para as ascendentes térmicas criamos um novo modelo de *Térmica-Bolha*, incluído o seu movimento e as suas interacções com as correntes envolventes. Também extendemos o modelo de *Térmica-Chaminé*, para incluir movimento e as interacções com as correntes envolventes. Em termos de gradientes de vento, criamos três modelos para o fenómeno de *Gradiente entre Camadas* e um modelo para o fenómeno de *Gradiente de Crista*. Estes modelos servem de base para os métodos de estimação usados para localizar e caracterizar os respectivos fenómenos.

Também analisamos qual é o equilíbrio necessário entre as características dos fenómenos atmosféricos e as características aerodinâmicas das aeronaves, de forma a possibilitar a realização de voos perpétuos. O resultado é intuitivo, mostrando que é mais simples executar voos perpétuos se a aeronave for mais eficiente e se os fenómenos atmosféricos forem mais intensos, apresentando uma correntes ascendente mais forte ou um gradiente mais intenso.

De modo a permitir a exploração da energia existente nas correntes atmosféricas, a maior parte dos controladores de voo requerem uma estimativa dos parâmetros do fenómeno em questão. Nós mostramos que os parâmetros dos fenómenos estudados são observáveis e apresentamos métodos para os estimar. Estes métodos baseiam-se em Filtros de Partículas, que lidam bem com a natureza não linear dos modelos dos fenómenos e com distribuições de probabilidade não Gaussianas. Os resultados dos testes executados são promissores, mostrando uma boa performance de estimação e requerendo um poder computacional baixo.

Durante o estudo desta temática notamos a falta de dados experimentais para validação dos modelos dos fenómenos estudados. Porquanto, desenvolvemos métodos de controlo de voo que nos permitem usar UAVs para recolher dados espacialmente distribuídos que permitissem a validação dos modelos estudados. Para tal, estudamos e implementamos métodos que evitassem colisões e que controlassem voos em formação, para tornar seguros os voos de amostragem dos fenómenos e também a execução de outros voos colaborativos. Os algoritmos desenvolvidos baseiam-se em Sistemas Híbridos e controlo *Sliding Mode*. O controlador de voo em formação apresenta uma boa performance, com bom seguimento do trajecto solicitado e, ainda mais importante, mantendo uma distância de segurança entre as aeronaves em formação ao longo do trajecto de voo.

CHAPTER I

Introduction

1.1 Motivation & Background

This thesis focuses on the problem of flight over time-varying flow fields. As such, we investigate flow field models and methods to detect, explore and exploit them. The main goal is to make Unmanned Aerial Vehicle (UAV) operations more sustainable and increase flight endurance, while either avoiding the need for any fuel or reducing its consumption rate substantially. Longer endurance means enhanced flight productivity, by reducing the operational overhead time, i.e., the flight time spent on the transitions between the airfield and the target operation area. The reduction in fuel consumption will make UAVs more environmentally friendly and reduce both the operation costs and the airplane detection signature.

The use of aircraft, and particularly UAVs, has been steadily increasing, both in civilian and military contexts. Both the US Army and the US Air Force reached the milestone of a million flight hours with UAVs each [10], which means that a lot of fuel could be saved by using air flow energy. If the energy gain was only 50%, systems similar to the ScanEagle could fly almost 50 hours, with a range of about 2500 nautical miles (NM) and consuming only 0.0031 L/NM (1060 miles/gal).

We focus on airplanes (fixed wing aircraft) as these, unlike rotary wing aircraft, can idle or shutdown the propulsion system and still can take advantage of the flow field energy, saving more fuel. Further, we study the specific advantages that UAVs may take from flow fields, as a special kind of airplane that include avionic packages capable of autonomously controlling the flight dynamics and the flight path. We assume such an avionics package includes a position sensor, as a GPS receiver, an Inertial Measurement Unit (IMU), with accelerometers and gyroscopes, and airprobes, sensing the dynamic and static air pressures.

Edwards [11] achieved a 48.6 km flight with no engine and only 143 meters of initial height over Edwards Air Force Base in California. This means that the air energy harvesting strategy saved 96% of the total energy required for the whole flight. The useful flight time (path following) was about 79%. He also reported a 113.5 km (70.5 miles) unpowered flight in a cross-country soaring competition [12], which shows that long soaring flights over different terrain types is feasible with UAVs.

We now analyze the coastal patrolling case as just an example of an operational scenario where the use of UAVs that harvest airflow energy would improve operational time and cost efficiency. Data from Portuguese Air Force patrol sorties provided by sergeants Teixeira and Gomes [13] reveals that flights are limited to an average of 2.8 hours with 71% of useful time. Further, the fuel consumption is 1.4 L/NM (1.9 miles/gal), summing up to 140 thousand liters for 2009, just for one of the patrol squadrons, and just for Portugal. With UAVs using air harvested energy, the sorties flight time could easily reach 10 hours, increasing the useful flight time to 72,5%, and the range by at least 26%. Several UAVs may be needed to replace each of these manned aircraft. The fuel consumption would be reduced by 93.4% for a similar payload weight, saving about 130 thousand liters per year of fuel, only on these patrol missions, in Portugal. Most of these patrol missions were along the national coast. This allows us to extrapolate the fuel savings to other countries or regions. The Portuguese coastline length is 2830 kilometers (1528 NM) [14]. The United States coastline length is 133 thousand kilometers (72 thousand NM)[14], yielding possible savings above 6.1 million liters per year. This is a very conservative estimate if we consider the area-to-coastline factor for the US is a lot bigger than for Portugal. The European and world coastline length are about 326 thousand kilometers (176 thousand NM) and 1.6 million kilometers (883 thousand NM), respectively. This indicates that the possible savings are above 15 million liters for Europe and 75 million liters world wide. This may amount to more than 100 million euros of yearly savings.

One of the fields where UAVs are showing their invaluable capabilities is meteorology. They are more controllable and have higher ranges than blimps, executing precise surveys. Further, they pose a good alternative for manned aircraft, executing dull missions with superior trajectory tracking precision. Another advantage is the reduction in risk cost, allowing researchers to gain knowledge over more dangerous phenomena [15], like hurricanes, or with riskier operation concepts, like formation flight [16]. NASA has been involved in several programs aimed at improving meteorologic models and survey meteorologic phenomena. The Global Hawk UAV has been used in several studies and there are also initiatives to create smaller UAVs tailored for meteorologic surveys [17]. Reuder et al. used a small UAV to estimate meteorological profiles [18, 19]. Ramana et al. were only able to study the atmospheric solar absorption in a risk and cost controlled manner resorting to three UAVs flying in a vertical formation [16].

Furthermore, aquatic vehicles are also subject to flow fields. This means that some of the conclusions of the current work for UAVs can easily be extended to Autonomous Underwater Vehicles (AUVs) and Autonomous Surface Vehicles (ASVs). The Wave Glider is a good example of exploitation of unsteady flow fields [20]. The use of different fluid flow shear (Wind/Water) is also a potential alternative [21]. Water density is almost a thousand times larger than the air density, meaning that aquatic flow gradients may present a very powerful energy source. As such, aquatic flow field exploitation could be a strong alternative to some of the current methods based on buoyancy variation, used by AUV gliders.

1.2 Existing Work

The first theories on soaring mechanics appeared in the 19th century. Peal [1] reported his observations of birds soaring, including pelicans, adjutants, vultures, cyrus and other large birds (fig. 1.1). Rayleigh [22] proposed a theoretical explanation for the dynamic soaring mechanism, based on a vertical gradient of the horizontal wind. This was corroborated by Manley's observations [23] of gradient energy harvesting by kites. More recently, Sachs described how Albatross use the wind shear near the ocean surface to support their almost flap-less flight [8]. Further, Sachs et al. [24, 25] obtained measurements of Albatross flights, some extending more than 25,000 km in 43 days. Weimerskirch et al. [26] report on Frigatebird flights lasting up to 94 hours, meaning that night soaring is also feasible.

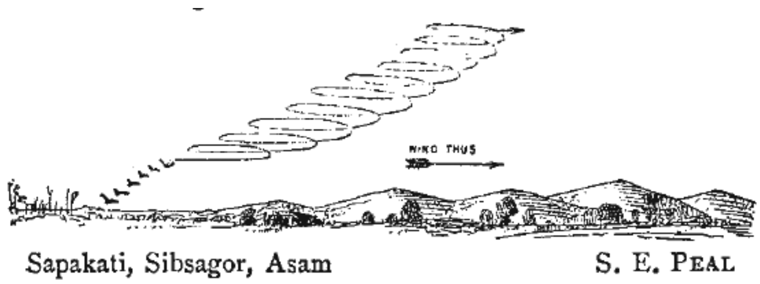


Figure 1.1: Observations of soaring birds by S. E. Peal (Illustration from [1])

In most of the flight control literature, the flow field is considered as a disturbance, which has to be compensated for. Some authors propose methods to harvest energy from air flow fields such as thermals [27, 28, 3, 29, 30, 11], wind shear [3], and gusts [9].

Thermals are updrafts (rising air masses) created by temperature variations (fig. 1.2). Wind shear is an air layer which presents a flow gradient (fig. 1.3). It appears between fluid masses moving with different speeds or in different directions.

There are several studies about the required conditions to achieve perpetual flight with wind shear [6, 8, 7] and gusts [9, 31]. For thermals, we were not able to find similar studies, although there are lots of reports of day long manned-glider flights and of long UAV-glider flights [12].

Wharington [27] was the first to propose thermal soaring control for UAVs. Andersson and Kaminer [29] proved the stability of a flight control algorithm which centered the flight on a thermal avoiding the need to map the thermal itself, by using the aircraft energy balance derivatives. Klesh et al. [30] showed the similarities between the thermals mapping problem and the general exploration problem. The authors used an intensity grid mapping of vertical disturbances for multi-vehicle measurements. Lawrance and Sukkarieh [3] presented an energy based method for soaring path planning. This method is suitable for static and dynamic soaring, i.e., in thermals, wind shear and other flow fields. As such, this method extends the favorable soaring conditions.

Hazen [32] describes a cooperative thermal soaring scheme using a thermal centering scheme similar to Andersson and Kaminer [29]. First, the operation area is divided in search blocks among the team UAVs. When an UAV detects a thermal all the other UAVs converge to that position, unless they find other thermals on the way. The information exchanged is the position of the UAV that detected a thermal. The thermal center is not explicitly localized as in [29]. One of the most interesting results

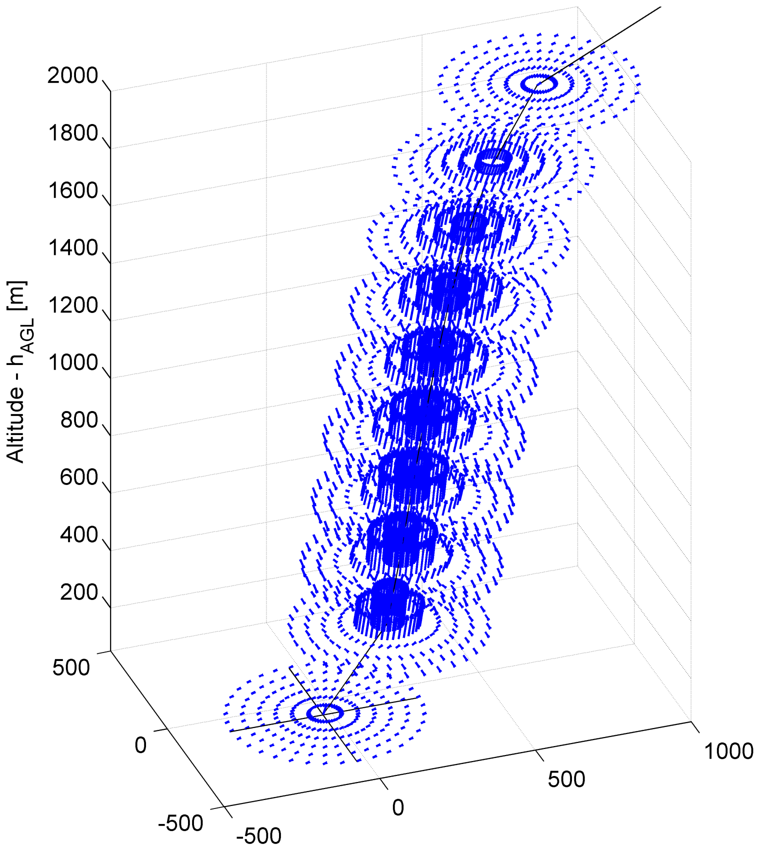


Figure 1.2: Column thermal updraft 3D representation.

was the comparison among the distribution of the maximum soaring flight times for different team sizes. As expected, this showed an improvement, i.e., increased flight time, with the number of UAVs, although the largest difference was from a single UAV to pair of UAVs. Andersson et al. [33] describe a similar approach where they use a thermal localization technique based on Allen's work [28]. They also present an analytical study about the relation between the thermal detection probability and the UAV team size. The derivation results confirm the simulation results obtained by Hazen [32]. Although an explicit thermal center localization method is used, this is only able to localize thermals that are encircled by the flight path. Further, the localization method runs independently on each vehicle, making no use of the extra information provided by the cooperation.

Most of the methods developed to exploit thermals and wind shear require the estimation of the thermal position or at least of the flow field over an array of spatially distributed points. There exist several estimators for thermals more or less capable of tracking the center thermal position (sec. 4.3). Allen [34] presented and flight-tested a centroid-based method to dynamically estimate the localization of a thermal center. The tests revealed problems of the algorithm, yielding poor quality estimates [11]. Edwards [11] combined Allen's method with a neural-network based locator [27]. Flight tests revealed that the estimation quality was good enough to allow highly effective thermal energy harvesting. Both methods assumed the estimation of a single

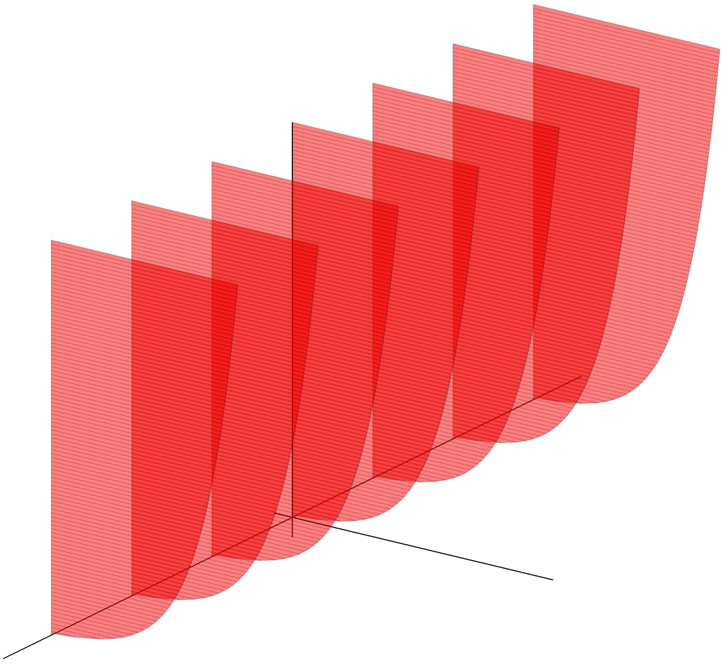


Figure 1.3: *Surface Wind Shear* 3D representation

thermal at each time and a simplified 2D thermal model. Both simplifications limit the advantages a team of UAVs could benefit from by cooperating in the exploitation of thermals and the estimation of thermals' parameters. Other estimation solutions as the Gaussian process (GP) regression presented by Lawrance and Sukkarieh [35] describe the flow field over spatially distributed points. This method is model independent, being able the estimate a general flow field. As such, it assumes nothing relative to the atmospheric phenomena number or three dimensional structure. This makes it capable of characterizing any flow field phenomena and, theoretically, allows a team of UAVs to exploit the information in a more advantageous manner. That said, the estimate uses a quantity of variables that would be very heavy on any communication system being used to synchronize the cooperating UAVs.

There are models for several of the phenomena discussed (sec. 2.5) which support operation simulations and many of the estimation algorithms. For thermals, there are simple models based on a Gaussian curve [27, 9] and more detailed models describing *Chimney Thermals* [4] and *Bubbles Thermals* [3]. The studied atmospheric phenomena models are derived mainly from bird flight observations, human pilot's observations and fluid dynamics theory. It is important that models are compared to real airflow data to validate their accuracy. As described in section 2.5.2.9, in the case of updrafts, there are studies presented by Lenschow and Stephens [36] and Allen [4] with real updraft measurements about some of the external parameters governing the thermal strength, size, and appearance. Other than those, we have not found any experimental studies validating the flow field details described by the phenomena models.

Most methods to harvest flow field energy gather the surplus energy as potential energy, i.e., gaining height. An alternative could be the use of electric motors installed on motor gliders as generators with variable pitch propellers. Grenestedt and Spletzer describe how an UAV equipped with a ram air turbine could generate electric power for the control electronics and payload while soaring through the Jet-stream [37].

1.3 Technical Approach

1.3.1 Perpetual Flow Field Flight

The main goal of the thesis is to define how UAVs may stay aloft perpetually or at least extend their endurance by harvesting energy from the atmospheric air flow phenomena. We analyze the necessary conditions for perpetual flight. These conditions depend on the flow field short and long-term parameters and the aircraft performance parameters. We study several atmospheric air flow phenomena as their spatial distribution is not even, and to execute a mission an UAV may have to use different phenomena available over its path. For example, updrafts tend to be abundant in some regions (mountains and plains) and rare in others (swamps). *Surface Wind Shear* appears over all water or terrain surfaces, but is usable only over the ocean, wide rivers or plains.

The current work develops better phenomena estimation methods for flight over partially/locally observed flow fields. The estimation methods rely on the aircraft sensor data. The inference methods implemented to fuse this data, take into account the nonlinear nature of the phenomena, can handle non-Gaussian probability distributions, and are more suitable for multi-vehicle collaborative inference. We also study the observability requirements and constraints for the different phenomena. This allows us to define a minimum set of required sensor data and informative flight paths.

Thermals have been studied by several authors, in terms of localization and exploitation. We extend their work with estimation methods that are capable of estimating the thermal 3D parameters, and which are more suitable for collaborative localization. Further, we adapt some of the existing thermal models and present new ones to better capture the thermal shape and dynamics.

Some authors presented methods to exploit wind shear. These methods usually require some knowledge of the flow field. We present a localization method for wind shear which, as the thermal localization methods, is more suitable for collaborative estimation.

1.3.2 Standard UAVs

The current approach is applicable to standard UAVs. The energy is harvested through flight path control, avoiding the need to integrate special components such as solar panels. Further, the inference methods which feed the control system require only measurement data available in standard autopilots, i.e., autopilots equipped with an IMU, a Global Navigation Satellite System (GNSS) receiver, and an airspeed probe.

1.3.3 Control Tools

1.3.3.1 Atmospheric Spatial Sampling - Formation Flight

As stated before, we have not found any experimental studies which validated the existing atmospheric phenomena models. UAVs may be used to measure several environmental characteristics, including the air flow velocity. The air flow may be computed by most UAVs' navigation systems. This make the UAVs themselves an useful tool for characterizing the space they move in.

To validate the air flow phenomena models in terms of shape, flow field, and dynamics, we need to obtain a set of air flow samples distributed in space and time. In particular, to verify the flow field, we need to collect spatially distributed samples simultaneously. Multiple UAVs flying simultaneously may execute this task. However, the atmospheric phenomena studied are sized on the order of tens to hundreds of meters. As such, to obtain good spatial samples we need the UAVs to fly close together. To maintain several aircraft flying closely, we envisioned that the best choice would be to control them with a formation flight control system. This has the advantages of facilitating collision avoidance and maintaining a well defined and nearly constant spatial distribution of the samples.

1.3.3.2 Collision Avoidance

Before implementing the formation flight control with collision avoidance we studied the collision avoidance problem by itself. This is important not only for the formation flight control but also to implement collaborative flight strategies safely. As an example, if there is a team of UAVs tracking several atmospheric phenomena in the same area, such as thermals, they need a Collision Avoidance System (CAS).

1.4 Original Contributions

The original contributions of this thesis are as follows.

1.4.1 Flight Perpetuity Requirements

Flight perpetuity with thermals Necessary condition for *sustainable flight*, relating the updraft average strength, the thermals' appearance rate and the aircraft aerodynamic parameters.

Flight perpetuity with wind shear Necessary condition for *perpetual flight*, relating the gradient function with altitude and the aircraft aerodynamic parameters.

1.4.2 Flow Field Phenomena Localization

Phenomena models This work improves on several of the proposed models for thermals. We included modeling features for the interaction of the phenomenon with wind and the resulting 3D structure. Further, we present new models for *Bubble Thermals*, *Layer Wind Shear*, and *Ridge Wind Shear.*

Thermal phenomenon observability Observability study of the nonlinear thermal models.

Column and Bubble Thermal locators Development, implementation and validation of two Particle Filter estimators which handle the nonlinear Column and Bubble Thermal models.

Wind vector observability Observability study of the horizontal wind vector components.

Surface Wind Shear estimators Development, implementation and validation of a Particle Filter estimator which handle the nonlinear *Surface Wind Shear* models.

1.4.3 Control Tools

Close proximity collision avoidance system Hybrid Systems controller for helicopter collision avoidance in close proximity operation.

Formation flight with collision avoidance Sliding mode controller for safe formation flight of fixed wing aircraft.

1.5 Scenario - Team Cooperation with Flow Field Supported Flight

We describe a potential UAV operational scenario, where a team of UAVs would use methods developed in the thesis to take advantage of the flow field energy to improve the operation efficiency and effectiveness.

In this scenario we define three different UAV types:

Mission UAV is the UAV carrying the necessary payload to accomplish the mission goal. Preferably, it would have somewhat better soaring performance than average aircraft. It would harvest air energy when necessary or during the mission fulfillment, if both activities were compatible.

Spotters are the UAVs designed for soaring flight, which continuously survey a region of interest for updrafts and wind shear, and maintain long flight endurance by harvesting air energy.

Communication relays are the UAVs that soar on the spotted updrafts or wind shear, establishing a communication network from the ground station(s) to the *Mission UAV*.

The scenario is a search and surveillance mission. In this scenario the UAV team is constituted by two *Mission UAVs*, three *Spotters* and one *Communication relay UAV*. During the deployment and recovery legs, i.e., the paths to and from operation area, the *Mission UAVs* and the *Communication relay UAV* fly in formation increasing their individual efficiency. The *Spotters* fly ahead and around to localize *on-way* hopping (high flow field energy) spots.

When one of these hopping spots is localized the UAV team converges to it to gather energy. The UAVs remain at the hopping spot until the operator commands them to proceed with the mission or an energy threshold is reached. As the *Spotters* soar more efficiently, some of them will reach the threshold sooner and will leave to search for other hopping spots.

The *Communication relay UAV* leaves the formation when the team crosses a mountain range, and starts relaying the communications. One of the *Spotters* remains with it to help with the localization of *Ridge Lift* or thermals. Once they localize one of these phenomena they gather energy in the form of altitude. When the *Spotter* has enough energy it leaves to search for more phenomena so that the *Communication relay UAV* may use them later. It only returns to one of the detected phenomena when its altitude reaches a lower limit.

When the rest of the UAVs reach the mission target area, the *Spotters* maintain the search for useful phenomena in and around the operational area. They head to one of the localized phenomena only when they reach the lower energy limit. The *Mission UAVs* start the search for the goal target. When their energy reaches a lower threshold they head to the nearest localized phenomenon. They stay there until they harvest enough energy to return to the mission. Each of the *Mission UAVs* may switch between the goal search mode and the energy harvesting mode independently, as they may have different energy levels and use different phenomena.

When the goal target is found, the team decides if there is a phenomenon near enough to the target so that a *Mission UAV* may be harvesting energy while maintaining the target under surveillance. If this is not possible one of the *Mission UAVs* maintains the target under surveillance while the other harvests energy. They switch whenever one of them reaches the lower or upper energy threshold, respectively.

1.6 Thesis Organization

The remainder of the thesis is organized in five chapters: *Models* (II), *Perpetual Flight Conditions* (III), *Flow Field Estimation* (IV), *Control* (V), and finally the *Conclusions* (VI).

The *Models* chapter introduces all models used throughout the thesis. We start be defining the reference frames (2.2), the aircraft describing parameters (2.3), and several aircraft dynamics models (2.4). The *Environment Models* section (2.5) is one of the most important parts of this thesis. It describes several flow field phenomena models, some already presented by other authors, others adapted from existing ones, and some developed during the work in this thesis. These models describe wind shear layers (2.4.9), updraft flows (2.5.2), including thermals, and gusts (2.5.3).

In the *Perpetual Flight Conditions* chapter we study how several aircraft and environmental parameters relate, and how these relations affect the capability to perform perpetual flight. In the beginning we introduce several definitions about the aircraft energy (3.2) and the *perpetual flight* terminology (3.3). Then we describe the necessary conditions to perform perpetual flight with thermals (3.4), wind shear (3.5) and gusts (3.6).

The *Flow Field Estimation* chapter describes estimation methods to determine the parameters of thermals and wind shear. The chapter is initiated with the description of a method to study the observability of nonlinear systems (4.2). We show thermals are observable and present a Particle Filter (PF) thermal estimator (4.3). Then we study the wind components observability and present a PF estimator for the *Surface Wind Shear* parameters (4.4).

The *Control* chapter presents two control systems: a Collision Avoidance System (CAS) for helicopters (5.2) and a formation flight controller with collision avoidance for fixed wing aircraft (5.3). The CAS controller is defined in the framework of hybrid systems and allows us to understand the problem of checking the collision avoidance capabilities of a controller. The formation flight controller was created to enable aircraft to fly safely in close proximity. This allows us to measure the flow field in several points simultaneously, enabling an accurate characterization. In order to fly safely in close proximity the airplane formation controller includes a collision avoidance that took into account the lessons learned with the CAS controller for helicopters.

To finish we present the conclusion of the thesis studies. As part of those conclusions we also describe the problems that remain unsolved and propose some steps to improve the knowledge of flow fields and the methods to exploit their energy.

CHAPTER II

Models

2.1 Introduction

This chapter describes the models used throughout the thesis. There are two main sets of models: Unmanned Aerial Vehicle (UAV) models and flow field models. The UAV models describe aircraft dynamics with varying detail (2.4). The environment models define the flow fields in which the vehicles move and by which they are affected. The described models are: the wind shear (2.4.9), which is a phenomenon that appear in between the atmosphere layers; the updrafts (2.5.2), with more focus on thermals which appear in the lower layer of the atmosphere and are characterized by an updraft limited to a radius from 100 to 1000 meters; and the gusts (2.5.3), which are mainly the result of turbulence. The nomenclature and frames lay down the basis for the described models.

2.2 Frames

In this work we use the standard reference frames for aircraft. The aircraft body frame has the x_B axis pointing to the nose, the y_B axis pointing to the right wing tip, the z_B axis pointing down, and the origin at the mass center of the aircraft. The subscript B stands for Body frame. The ground reference frame, also assumed to be an inertial frame, is centered at the ground origin, with the x_G axis pointing to North, the y_G axis pointing to East, and the z_G axis pointing to the center of the Earth . The ground origin is a point on the Earth surface, e.g., an operation ground station. The subscript G stands for ground frame.

The body frame origin is translated $[x, y, z]^G$ in the ground frame relative to the latter's origin. From now on we will drop the subscript G on when referring to a position relative to the ground frame origin, which will be also referred to as the UAV position.

The rotation between the frames is defined by the 3-2-1 Euler angles $[\phi, \theta, \psi]$ (fig. 2.1). ψ is the yaw angle, which sets the rotation about the z_G axis. θ is the pitch angle, which is the angle between the x_B axis and the horizontal plane. ϕ is the roll angle, also referred to as bank, which defines the rotation about the x_B axis. The rotation matrix which transforms a vector expressed in the inertial frame to a vector

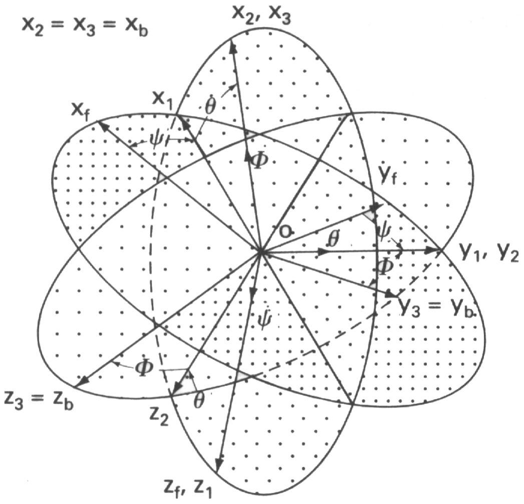


Figure 2.1: Euler rotation angles (illustration from [2])

in the body frame is

$$\boldsymbol{R}_{GB} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix} \tag{2.1a}$$

$$\mathbf{x}^B = \boldsymbol{R}_{GB}\mathbf{x}^G. \tag{2.1b}$$

2.3 Aircraft

Here we describe some of the most common parameters for aircraft.
All forces are assumed to act on the center of gravity CG. This is assumed to be the same as the Center of Mass (CM). As such, the aircraft is represented as a mass point with mass m and inertia characterized by the matrix:

$$\boldsymbol{I} = \begin{bmatrix} I_x & I_{xy} & I_{xz} \\ I_{yx} & I_y & I_{yz} \\ I_{zx} & I_{zy} & I_z \end{bmatrix}. \tag{2.2}$$

Most aircraft are symmetric about the vertical-longitudinal plane XZ, yielding $I_{xy} = I_{yx} = I_{yz} = I_{zy} = 0$.

In addition to the Euler angles, there are two important angles: the angle-of-attack α and the side-slip angle β (fig. 2.2). Both are important to set the aerodynamic coefficients. The Angle-of-Attack (AOA) is the angle between the aircraft body XY plane and its airspeed vector $\mathbf{v}_a$, as defined in (2.3a) . The side-slip angle is the angle between $\mathbf{v}_a$ and the aircraft's XZ plane.

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \tag{2.3a}$$

$$\beta = \sin^{-1}\left(\frac{v}{V_a}\right), \tag{2.3b}$$

where the surge (u), sway (v), and heave (w) are the body frame components of the airspeed vector ($\mathbf{v}_a$), and V_a is the airspeed norm.

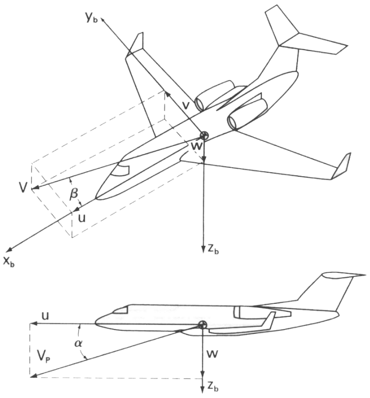


Figure 2.2: Velocity vector diagram (illustration from [2])

The wing main geometric parameters are the span b, area S, mean chord $\bar{c}$, aspect ratio AR, taper ratio λ, and incidence angle α_w. These also apply to the horizontal and vertical stabilizers: b_H, S_H, $\bar{c}_H$, AR_H, λ_H, α_H, b_V, S_V, $\bar{c}_V$, AR_V, and λ_V. All these lifting surfaces are characterized by an aerodynamic center position, about which the aerodynamic pitching moment is approximately constant relative to α. x_W is the distance from the wing aerodynamic center horizontal to the CM. The stabilizers are also characterized by their stabilization arms l_H and l_V. These are the longitudinal distances between the stabilizers' aerodynamic center and the aircraft's CM [2].

The main forces actuating on the aircraft are the aerodynamic lift, the aerodynamic drag, the propulsion thrust, and the gravity force, represented in figure 2.3 and defined as:

$$\mathbf{L} = L\left(\mathbf{x}_B \sin\alpha - \mathbf{z}_B \cos\alpha\right) \tag{2.4a}$$

$$\mathbf{D} = -D\left(\mathbf{x}_B \cos\alpha + \mathbf{z}_B \sin\alpha\right) \tag{2.4b}$$

$$\mathbf{T} = T\mathbf{x}_B \tag{2.4c}$$

$$m\mathbf{g} = mg\mathbf{z}_G \tag{2.4d}$$

$$\mathbf{v}_a = V_a\left(\mathbf{x}_B \cos\alpha - \mathbf{z}_B \sin\alpha\right). \tag{2.4e}$$

The equations for the aerodynamic forces are:

$$L = \rho\frac{V_a^2}{2}S \cdot C_L \tag{2.5a}$$

$$D = \rho\frac{V_a^2}{2}S \cdot C_D, \tag{2.5b}$$

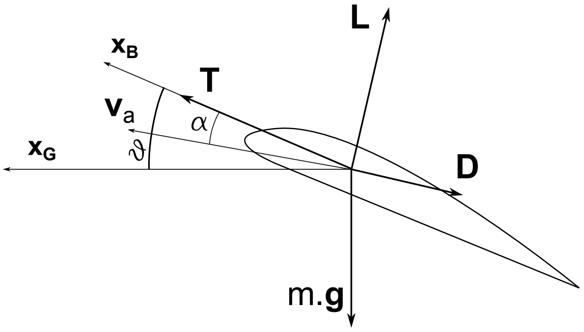


Figure 2.3: Aircraft forces diagram. *In this illustration the pitch angle θ is represented by ϑ.*

where $\rho \frac{V_a^2}{2} = Q$ is the dynamic pressure, ρ is the air density, S is the wing area, and C_L and C_D are the lift and drag coefficients. C_L may be considered linear for some ranges of α, and approximated by:

$$C_L \approx C_{L0} + C_{L\alpha}\alpha, \tag{2.6}$$

where C_{L0} and $C_{L\alpha} = \frac{dC_L}{d\alpha}$ are characteristic parameters of the aircraft. C_D is sometimes approximated by a quadratic function of α:

$$C_D \approx C_{D0} + k_{D/L} \cdot C_L^2. \tag{2.7}$$

where C_{D0} is the zero-lift drag coefficient and $k_{D/L}$ is a characteristic constant of the aircraft. $k_{D/L}$ is usually defined for a wing as

$$k_{D/L} = \frac{1}{\pi AR\epsilon}. \tag{2.8}$$

where AR is the wing aspect ratio and ϵ is the Oswald efficiency factor, which characterizes the deviation of the real wing shape relative to the elliptical case. Another definition for $k_{D/L}$ is that from [38],

$$k_{D/L} = \frac{1}{4E_{max}^2 C_{D0}}, \tag{2.9}$$

where E_{max} is an aerodynamic efficiency factor.

When the AOA is beyond the stall limits the aerodynamic coefficients converge to those of the ”flat plate” model, where:

$$C_L = 1.2\sin(2\alpha) \tag{2.10a}$$
$$C_D = 2\sin|\alpha|, \tag{2.10b}$$

As such, the aerodynamic coefficients for the whole AOA range (fig. 2.4) may be

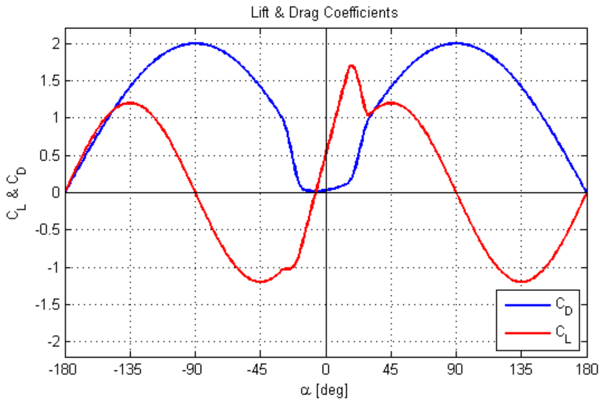


Figure 2.4: Aerodynamic coefficients through the full range of the AOA.

approximated by:

$$C_L = \begin{cases} C_{L0} + \frac{dC_L}{d\alpha}\alpha & \Leftarrow \alpha \leq \alpha_{stall} \\ \left(C_{L0} + \frac{dC_L}{d\alpha}\alpha\right)\lambda + 1.2\sin\left(2\alpha\right)\left(1-\lambda\right) & \Leftarrow \alpha_{stall} < \alpha \leq \alpha_{plate} \\ 1.2\sin\left(2\alpha\right) & otherwise \end{cases} \tag{2.11a}$$

$$C_D = \begin{cases} C_{D0} + k_{D/L}C_L^2 & \Leftarrow \alpha \leq \alpha_{stall} \\ \left(C_{D0} + k_{D/L}C_L^2\right)\lambda + 2\sin|\alpha|\left(1-\lambda\right) & \Leftarrow \alpha_{stall} < \alpha \leq \alpha_{plate} \\ 2\sin|\alpha| & otherwise \end{cases} \tag{2.11b}$$

$$with\ \lambda = 1 + 2\left(\frac{|\alpha| - \alpha_{stall}}{\alpha_{plate} - \alpha_{stall}}\right)^3 - 3\left(\frac{|\alpha| - \alpha_{stall}}{\alpha_{plate} - \alpha_{stall}}\right)^2,$$

where α_{stall} is the stall AOA and α_{plate} is the angle at which the airfoil enters in the "flat plate" flow regime.

Aircraft are controlled through deflections of the control surfaces or adjustments on the propulsion thrust δ_p. The usual controls are the elevator deflection δ_e, the ailerons deflection δ_a, the thrust change δ_p, and the rudder deflection δ_r. Other possible surfaces are the flaps δ_f and canards δ_c.

2.4 UAV Dynamic Models

This section describes several aircraft dynamic models. The different models present varying Degrees-of-Freedom (DOF) and detail levels, which means different complexity, computation costs, and modeling accuracy. Simpler models often provide clearer insight about the system behavior. Further, the computation cost reduction of simpler models is an important advantage, when response delays and other dynamics are irrelevant. More complex models provide a more accurate representation of the real aircraft dynamics, which may be important to check system performance and stability in the presence of more complex responses, in particular control delays.

In this work we use holonomic models to simulate helicopter UAVs. A helicopter can force a movement in any direction. This control is not constrained either by the direction of movement or by the aircraft yaw angle (ψ).

Airplanes are non-holonomic vehicles. This means that the vehicle state depends on the path executed until the current moment. Further, the feasible path depends on the aircraft's current state, as forces cannot be imposed in every direction. The two main control forces of the aircraft are the lift and the thrust. Thrust magnitude is controlled directly through the motor power. Lift magnitude is controlled indirectly through the aircraft AOA (α) and the aircraft airspeed. We use several non-holonomic models to simulate airplane dynamics. The *2D* motion models assume the vehicle is constrained to move on the horizontal plane (XY). All the other models simulate the vehicle behavior in a *3D* space.

2.4.1 Holonomic Models

The *holonomic model* for helicopters is constrained to the horizontal plane (XY). The first model is a perfect holonomic vehicle, modeled by the simplest possible dynamics, a single integrator:

$$\begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \mathbf{v}_i = \mathbf{v}_{cmd_i} \end{cases}, \tag{2.12}$$

where $\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$ and $\mathbf{v}_{cmd_i} = \begin{bmatrix} u_i \\ v_i \end{bmatrix}$. In this model the available control is in the form of velocity which is not realistic, but enables some kinematic analysis.

A more realistic holonomic model is the double integrator:

$$\begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = \mathbf{a}_{cmd_i} \end{cases}, \tag{2.13}$$

where $\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$, $\mathbf{v}_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix}$, and $\mathbf{a}_{cmd_i} = k_a \left(\mathbf{v}_{cmd_i} - \mathbf{v}_i \right)$. In this model the control input affects the accelerations. This model is more realistic, because vehicles can generate forces, and by Newton's second law, accelerations.

2.4.2 2D 3DOF Kinematic Model

The *2D 3DOF* motion model simulates an airplane, in which the autopilot forces coordinated turns (zero side-slip) at constant altitude. The planar motion of the airplane is simulated by a simple 3 DOF kinematic model. The model described next is an extension of a unicycle model [39].

$$\dot{x} = V_{a,cmd} \cdot \cos\left(\psi\right) + W_x = V_g \cdot \cos\left(\chi\right) \tag{2.14a}$$

$$\dot{y} = V_{a,cmd} \cdot \sin\left(\psi\right) + W_y = V_g \cdot \sin\left(\chi\right) \tag{2.14b}$$

$$\dot{\psi} = \omega = \frac{g \cdot \tan\left(\phi_{cmd}\right)}{V_g}, \tag{2.14c}$$

where ϕ and V_a are the controlled variables. V_a and V_g are the air-relative and ground-relative velocities, also designated as airspeed and ground speed, respectively. χ is the course angle, i.e., the direction the aircraft is actually following. W_x and W_y are the wind velocity components on x and y axis.

2.4.3 2D 4DOF Kinematic Model

The *2D 4DOF* motion model is yet a further extension of the unicycle model. In this model the autopilot also forces coordinated turns at constant altitude, but the vehicle responds to the controls with some delay. The airspeed (V_a) and roll (ϕ) are regarded as the autopilot pseudo-controls, with first-order responses. Further, we assume the dynamics and effects of the AOA are negligible. The planar motion of the airplane is simulated by a simple 4 DOF kinematic model:

$$\dot{x} = V_a \cdot \cos(\psi) + W_x = V_g \cdot \cos(\chi) \tag{2.15a}$$

$$\dot{y} = V_a \cdot \sin(\psi) + W_y = V_g \cdot \sin(\chi) \tag{2.15b}$$

$$\dot{\psi} = \omega = \frac{g \cdot \tan(\phi)}{V_g} \tag{2.15c}$$

$$\dot{V_a} = \frac{V_a - V_{a,cmd}}{T_V} \tag{2.15d}$$

$$\dot{\phi} = \frac{\phi - \phi_{cmd}}{T_\phi}, \tag{2.15e}$$

where V_a and V_g are the air-relative and ground-relative velocities, also designated as airspeed and ground speed, respectively. χ is the course angle, i.e., the direction the aircraft is actually following. W_x and W_y are the wind velocity components on x and y axis. ϕ and ψ are the aircraft roll and yaw angles. $V_{a,cmd}$ and ϕ_{cmd} are the controller outputs. T_V and T_ϕ are the controls' response times.

This model represents the fixed wing aircraft horizontal motion more accurately than the *2D 3DOF* model (sec. 2.4.2). The difference is noticeable only if the short term dynamics of roll and speed are important, as is the case of close proximity operation, e.g., formation flight and collision avoidance. Yet the mathematical and computational complexity are lower than some of the following models, providing a good basis for controller synthesis and state propagation in real time.

2.4.4 3D 4DOF Kinematic Model

The *3D 4DOF* motion model is another extension of the unicycle model by the addition of altitude dynamics. In this model the autopilot still forces coordinated turns, but the altitude may change. The airspeed (V_a), roll (ϕ), and vertical rate ($\dot{h}$) are regarded as the autopilot pseudo-controls, effecting instantly:

$$\dot{x} = V_{a,cmd} \cdot \cos(\psi) + W_x = V_g \cdot \cos(\chi) \tag{2.16a}$$

$$\dot{y} = V_{a,cmd} \cdot \sin(\psi) + W_y = V_g \cdot \sin(\chi) \tag{2.16b}$$

$$\dot{\psi} = \omega = \frac{g \cdot \tan(\phi_{cmd})}{V_g} \tag{2.16c}$$

$$\dot{h} = \dot{h}_{cmd}, \tag{2.16d}$$

where $V_{a,cmd}$, ϕ_{cmd}, and $\dot{h}_{cmd}$ are the controlled variables, and, as before, V_a and V_g are the airspeed and ground speed, χ is the course angle, W_x and W_y are the wind velocity components, and ϕ and ψ are the aircraft roll and yaw angles. Notice that this is same model as the *2D 3DOF* model (sec. 2.4.2), except for the altitude variation.

2.4.5 3D 5DOF Kinematic Model

The *3D 5DOF* motion model is a merge of the *2D 4DOF* and the *3D 4DOF* models, with a delay on the vertical rate dynamics. This model still assumes that the autopilot forces coordinated turns. The airspeed ($V_{a,cmd}$), the vertical rate ($\dot{h}_{cmd}$), and roll (ϕ_{cmd}) are regarded as the autopilot pseudo-controls, with first-order responses:

$$\dot{x} = V_a \cdot \cos(\psi) + W_x = V_g \cdot \cos(\chi) \tag{2.17a}$$

$$\dot{y} = V_a \cdot \sin(\psi) + W_y = V_g \cdot \sin(\chi) \tag{2.17b}$$

$$\dot{\psi} = \omega = \frac{g \cdot \tan(\phi)}{V_g} \tag{2.17c}$$

$$\dot{V_a} = \frac{V_a - V_{a,cmd}}{T_V} \tag{2.17d}$$

$$\ddot{h} = \frac{\dot{h} - \dot{h}_{cmd}}{T_{\dot{h}}} \tag{2.17e}$$

$$\dot{\phi} = \frac{\phi - \phi_{cmd}}{T_\phi}, \tag{2.17f}$$

where T_V, $T_{\dot{h}}$, and T_ϕ are the control response times.

2.4.6 3D Stabilized Motion Model

The *3D stabilized motion* model simulates an airplane complete 6 DOF model. It is an important model for the aircraft energy dynamics analysis, developed in section 3.2. In this model, as in the previous ones, it's assumed that there is an autopilot controlling the low level dynamics. The velocity equation is:

$$\vec{V} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \dot{x}_a \\ \dot{y}_a \\ \dot{z}_a \end{bmatrix} + \mathbf{w} = V_a \begin{bmatrix} \cos\psi\cos\gamma_a \\ \sin\psi\cos\gamma_a \\ \sin\gamma_a \end{bmatrix} + \begin{bmatrix} W_x \\ W_y \\ W_z \end{bmatrix} \tag{2.18}$$

where $\vec{V} = [\dot{x}, \dot{y}, \dot{z}]^\mathsf{T}$ is the velocity vector, $[\dot{x}_a, \dot{y}_a, \dot{z}_a]^\mathsf{T}$ is the velocity vector relative to the air (flow field), and $[W_x, W_y, W_z]^\mathsf{T}$ is the wind velocity vector. All these vectors are projected on the ground reference frame. V_a is a scalar representing the total air relative speed. The air-climb angle (γ_a - fig. 2.5) is defined as,

$$\gamma_a = \arctan \frac{\dot{z}_V - W_z}{\sqrt{(\dot{x}_V - W_x)^2 + (\dot{y}_V - W_y)^2}}. \tag{2.19}$$

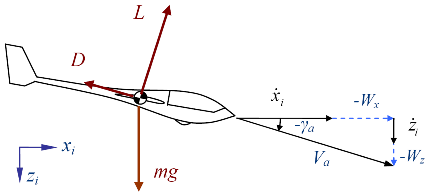


Figure 2.5: Wind reference frame (illustration from [3])

If we differentiate equation 2.18, we obtain,

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \dot{V}_a \begin{bmatrix} \cos\psi\cos\gamma_a \\ \sin\psi\cos\gamma_a \\ -\sin\gamma_a \end{bmatrix} + V_a\dot{\psi} \begin{bmatrix} -\sin\psi\cos\gamma_a \\ \cos\psi\cos\gamma_a \\ 0 \end{bmatrix} - \\ - V_a\dot{\gamma}_a \begin{bmatrix} \cos\psi\sin\gamma_a \\ \sin\psi\sin\gamma_a \\ \cos\gamma_a \end{bmatrix} + \begin{bmatrix} \dot{W}_x \\ \dot{W}_y \\ \dot{W}_z \end{bmatrix}. \tag{2.20}$$

If we assume a static flow field the flow velocity derivatives may be simplified to:

$$\begin{bmatrix} \dot{W}_x \\ \dot{W}_y \\ \dot{W}_z \end{bmatrix} = \mathbf{J}_W \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}, \tag{2.21}$$

where $\mathbf{J}_W$ is the Jacobian of the wind speeds:

$$\mathbf{J}_W = \begin{bmatrix} \frac{\partial W_x}{\partial x} & \frac{\partial W_x}{\partial y} & \frac{\partial W_x}{\partial z} \\ \frac{\partial W_y}{\partial x} & \frac{\partial W_y}{\partial y} & \frac{\partial W_y}{\partial z} \\ \frac{\partial W_z}{\partial x} & \frac{\partial W_z}{\partial y} & \frac{\partial W_z}{\partial z} \end{bmatrix}. \tag{2.22}$$

The angle-of-attack is important for this model and is defined by:

$$\alpha = \theta - \gamma_a. \tag{2.23}$$

As the autopilot is assumed to control the low level dynamics, including the regulation of the side-slip, we assume that $\beta \approx 0^{\circ}$.

Let us now take the simpler case where the UAV is flying with the wings leveled and constrained to the x-z plane, with $\psi = 0$, $\phi = 0$ and $\dot{\psi} = 0$. Let us also consider that α is small enough to approximate $T \cdot [\cos\theta, \sin\theta]^{\intercal}$ by $T \cdot [\cos\gamma_a, \sin\gamma_a]^{\intercal}$. The equations of motion governing the UAV are (fig. 2.5):

$$m\ddot{x} = -L\sin\gamma_a + (T - D)\cos\gamma_a \tag{2.24a}$$

$$m\ddot{z} = -L\cos\gamma_a - (T - D)\sin\gamma_a + mg, \tag{2.24b}$$

where L and D are the aerodynamic lift and drag, T is the propulsion thrust, m is the aircraft mass, and g is the acceleration of gravity. If we differentiate $\dot{x}_V$ and $\dot{z}_V$ from (2.18), we obtain,

$$\begin{bmatrix} \ddot{x} \\ \ddot{z} \end{bmatrix} = \dot{V}_a \begin{bmatrix} \cos\gamma_a \\ -\sin\gamma_a \end{bmatrix} - V_a \begin{bmatrix} \sin\gamma_a \\ \cos\gamma_a \end{bmatrix} \dot{\gamma}_a + \begin{bmatrix} \dot{W}_x \\ \dot{W}_z \end{bmatrix}. \tag{2.25}$$

Combining (2.25) with the equations of motion (2.24):

$$\left(\dot{V}_a - \frac{T-D}{m}\right)\begin{bmatrix}\cos\gamma_a \\ -\sin\gamma_a\end{bmatrix} + \begin{bmatrix}\dot{W}_x \\ \dot{W}_z\end{bmatrix} = \left(V_a - \frac{L}{m}\right)\begin{bmatrix}\sin\gamma_a \\ \cos\gamma_a\end{bmatrix}\dot{\gamma}_a + \begin{bmatrix}0 \\ g\end{bmatrix} \tag{2.26}$$

Solving for $\frac{dV_a}{dt}$ yields,

$$\dot{V}_a = \frac{T-D}{m} - g\sin\gamma_a - \begin{bmatrix}\cos\gamma_a \\ -\sin\gamma_a\end{bmatrix}^\intercal \begin{bmatrix}\dot{W}_x \\ \dot{W}_z\end{bmatrix}. \tag{2.27}$$

This expression will enable us to evaluate the kinetic energy gain.
Extending this derivation for turning maneuvers, we may still consider the UAV $\hat{\mathbf{x}}_B$ axis instantaneously coincident with the x-z plane, i.e., without loss of generality, $\psi = 0$. The equations of motion are:

$$m\ddot{x} = -L\sin\gamma_a + (T-D)\cos\gamma_a \tag{2.28a}$$

$$m\ddot{y} = L\cos\gamma_a\sin\phi + (T-D)\sin\gamma_a\sin\phi \tag{2.28b}$$

$$m\ddot{z} = -L\cos\gamma_a\cos\phi - (T-D)\sin\gamma_a\cos\phi + mg, \tag{2.28c}$$

Combining (2.20) with (2.28) yields a similar result to (2.27) with the addition of the $\dot{\psi}$ equation:

$$\dot{V}_a = \frac{T-D}{m}\cos\phi - g\sin\gamma_a - \begin{bmatrix}\cos\gamma_a \\ -\sin\gamma_a\end{bmatrix}^\intercal \begin{bmatrix}\dot{W}_x \\ \dot{W}_z\end{bmatrix} + \frac{L\dot{\gamma}_a}{m}\tan\gamma_a\left(\cos\phi - 1\right) \tag{2.29a}$$

$$\dot{\psi} = \frac{L}{mV_a}\sin\phi + \frac{T-D}{mV_a}\tan\gamma_a\sin\phi - \frac{1}{V_a\cos\gamma_a}\dot{W}_y. \tag{2.29b}$$

2.4.7 Full 3D Dynamics Model

The *full 3D dynamics* model is a 6 DOF model. It represents the complete dynamics of a rigid body airplane. In the context of this work it is used to run higher fidelity simulations and to support the development of some of the estimation algorithms, such as the aircraft parameter identification and the flow field observer. It includes full rotation dynamics and accounts for wind gradient induced rotations.

In this section, we extend the models of [40] and [41] in order to incorporate the effects of wind. Most models used for aircraft stability and control disregard the flow field, assuming it is null for the equations derivation. As this work is about the interaction between the aircraft and the flow field, we take into account the aircraft velocity relative to the airflow $\mathbf{v}_a$ and the airflow velocity $\mathbf{w}^G$ to define the aircraft dynamics. As such, the linear velocity and acceleration equations are:

$$\dot{\mathbf{x}} = \mathbf{v}_a + \mathbf{w} \tag{2.30a}$$

$$\ddot{\mathbf{x}} = \frac{d}{dt}\mathbf{v}_a + \frac{d}{dt}\mathbf{w} \tag{2.30b}$$

where

$$\frac{d}{dt}\mathbf{v}_a^B = \dot{\mathbf{v}}_a^B + \omega^B \times \mathbf{v}_a^B \tag{2.31}$$

where $\mathbf{v}_a^B = [u, v, w]^\intercal$.

We will now describe the equations governing the forces and moments, followed by their effect on linear and angular accelerations. Next we present the equations describing the position and attitude evolution, using the ground frame velocities $\left[\dot{x}^G, \dot{y}^G, \dot{z}^G\right]^\intercal$ or the wind-relative speed and angles $[V_a, \alpha, \beta]^\intercal$. To finish we present the aerodynamic coefficients relations and decomposition, which are commonly referred to as stability derivatives.

If we group the forces and accelerations (2.30), we obtain the aircraft linear dynamics equation:

$$\begin{bmatrix} X+T \\ Y \\ Z \end{bmatrix} + m\mathbf{R}_{GB}\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = m\left(\begin{bmatrix} \dot{u}+qw-rv \\ \dot{v}+ru-pw \\ \dot{w}+pv-qu \end{bmatrix} + \dot{\mathbf{w}}^B\right), \tag{2.32}$$

where X, Y, and Z are the aerodynamic force vectors in the body axis $\hat{\mathbf{x}}_B$, $\hat{\mathbf{y}}_B$, and $\hat{\mathbf{z}}_B$, defined by their coefficients through:

$$X = C_X \cdot Q \cdot S \tag{2.33a}$$
$$Y = C_Y \cdot Q \cdot S \tag{2.33b}$$
$$Z = C_Z \cdot Q \cdot S, \tag{2.33c}$$

mg is the gravity force, T is the propulsion thrust force, and $\dot{\mathbf{w}}^B$ is the vector of flow field velocity derivatives transformed to the body frame:

$$\dot{\mathbf{w}}^B = \begin{bmatrix} \dot{W}_x^B \\ \dot{W}_y^B \\ \dot{W}_z^B \end{bmatrix} = \boldsymbol{R}_{GB}\dot{\mathbf{w}}^G, \tag{2.34}$$

and $\mathbf{w}^G$ is the vector of flow field velocity derivatives on the ground reference frame.

If we rearrange (2.32) we obtain the force equations:

$$\begin{aligned} \dot{u} &= rv - qw + \frac{QS}{m}C_X - g\sin\theta + \frac{T}{m} - \dot{W}_x^B \\ \dot{v} &= pw - ru + \frac{QS}{m}C_Y + g\cos\theta\sin\phi - \dot{W}_y^B \\ \dot{w} &= qu - pv + \frac{QS}{m}C_Z + g\cos\theta\cos\phi - \dot{W}_z^B. \end{aligned} \tag{2.35a}$$

where C_X and C_Z are:

$$C_X = C_L\sin\alpha - C_D\cos\alpha \tag{2.36a}$$
$$C_Z = -C_L\cos\alpha - C_D\sin\alpha, \tag{2.36b}$$

with C_L and C_D defined in detail further below, in (2.43). An alternative form of the force equations (2.35) are the wind-axes force equations:

$$\begin{aligned}\dot{V}_a &= -\frac{QS}{m}C_{D_W} + \frac{T}{m}\cos\alpha\cos\beta - \begin{bmatrix}\cos\alpha\cos\beta\\ \sin\beta\\ \sin\alpha\cos\beta\end{bmatrix}^{\intercal}\begin{bmatrix}\dot{W}_x^B\\ \dot{W}_y^B\\ \dot{W}_z^B\end{bmatrix} + ...\\ &\quad ... + g\left(\cos\phi\cos\theta\sin\alpha\cos\beta + \sin\phi\cos\theta\sin\beta - \sin\theta\cos\alpha\cos\beta\right)\end{aligned} \tag{2.37a}$$

$$\begin{aligned}\dot{\alpha} &= -\frac{QS}{mV_a\cos\beta}C_L + q - \tan\beta\left(p\cos\alpha + r\sin\alpha\right) - \frac{T\sin\alpha}{mV_a\cos\beta} + ...\\ &\quad ... + \frac{g}{V_a\cos\beta}\left(\cos\phi\cos\theta\cos\alpha + \sin\theta\sin\alpha\right) + \frac{\dot{W}_x^B\sin\alpha - \dot{W}_z^B\cos\alpha}{V_a\cos\beta}\end{aligned} \tag{2.37b}$$

$$\begin{aligned}\dot{\beta} &= -\frac{QS}{mV_a}C_{Y_W} + p\sin\alpha - r\cos\alpha + \frac{g}{V_a}\sin\beta\sin\phi\cos\theta + ...\\ &\quad ... + \frac{\sin\beta}{V_a}\left(g\cos\alpha\sin\theta - g\sin\alpha\cos\phi\cos\theta + \frac{T\cos\alpha}{m}\right) + \frac{1}{V_a}\begin{bmatrix}\cos\alpha\sin\beta\\ \cos\beta\\ \sin\alpha\sin\beta\end{bmatrix}^{\intercal}\begin{bmatrix}\dot{W}_x^B\\ \dot{W}_y^B\\ \dot{W}_z^B\end{bmatrix}\end{aligned} \tag{2.37c}$$

where C_{D_W} and C_{Y_W} are defined as:

$$C_{D_W} = C_D\cos\beta - C_Y\sin\beta \tag{2.38a}$$

$$C_{Y_W} = C_Y\cos\beta + C_D\sin\beta, \tag{2.38b}$$

with C_Y defined in detail further below, in (2.43).

For the rotation dynamics we have similar equations, the moment equations:

$$\dot{p} - \frac{I_{xz}}{I_x}\dot{r} = \frac{QSb}{I_x}C_l - \frac{I_z - I_y}{I_x}qr + \frac{I_{xz}}{I_x}qp \tag{2.39a}$$

$$\dot{q} = \frac{QS\bar{c}}{I_y}C_m - \frac{I_x - I_z}{I_y}pr - \frac{I_{xz}}{I_y}\left(p^2 - r^2\right) + \frac{I_p}{I_y}\Omega_p r \tag{2.39b}$$

$$\dot{r} - \frac{I_{xz}}{I_z}\dot{p} = \frac{QSb}{I_z}C_n - \frac{I_y - I_x}{I_z}pq - \frac{I_{xz}}{I_z}qr - \frac{I_p}{I_z}\Omega_p q, \tag{2.39c}$$

with C_l, C_m, and C_n defined in detail further below, in (2.43).

To obtain the position evolution we just need to integrate the linear velocities:

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix}^G = \mathbf{R}_{GB}^{\intercal}\begin{bmatrix}u\\ v\\ w\end{bmatrix} + \begin{bmatrix}W_x\\ W_y\\ W_z\end{bmatrix}^G \tag{2.40}$$

which can also be derived in terms of the wind-relative speed and angles, denominated the navigation equations:

$$\dot{x}^G = V_a \left[\cos\alpha\cos\beta\cos\psi\cos\theta + \sin\beta\left(\cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi\right) + ...\right.$$
$$\left. ... + \sin\alpha\cos\beta\left(\cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi\right)\right] + \dot{W}_x^G \tag{2.41a}$$
$$\dot{y}^G = V_a \left[\cos\alpha\cos\beta\sin\psi\cos\theta + \sin\beta\left(\sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi\right) + ...\right.$$
$$\left. ... + \sin\alpha\cos\beta\left(\sin\psi\sin\theta\sin\phi - \cos\psi\sin\phi\right)\right] + \dot{W}_y^G \tag{2.41b}$$
$$\dot{h} = V_a \left[\cos\alpha\cos\beta\sin\theta - \sin\beta\cos\theta\sin\phi - \sin\alpha\cos\beta\cos\theta\cos\phi\right] - \dot{W}_z^G \tag{2.41c}$$

Finally, the attitude can be obtained by integrating the kinematic equations:

$$\dot{\phi} = p + \tan\theta\left(q\sin\phi + r\cos\phi\right) \tag{2.42a}$$
$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{2.42b}$$
$$\dot{\psi} = \frac{q\sin\phi + r\cos\phi}{\cos\theta} \tag{2.42c}$$

The aerodynamic force and moment coefficients (C_L, C_D, C_Y, C_l, C_m, and C_n) are represented by a series of derivatives, called stability derivatives:

$$C_L = C_{L_0} + C_{L_V}\frac{\Delta V}{V_0} + C_{L_\alpha}\Delta\alpha + C_{L_{\dot{\alpha}}}\frac{\dot{\alpha}\bar{c}}{2V_0} + C_{L_q}\frac{q\bar{c}}{2V_0} + C_{L_{\delta_e}}\Delta\delta_e \tag{2.43a}$$
$$C_D = C_{D_0} + C_{D_V}\frac{\Delta V}{V_0} + C_{D_\alpha}\Delta\alpha + k_{D/L}\cdot C_L^2 + C_{D_q}\frac{q\bar{c}}{2V_0} + ...$$
$$... + C_{D_{\delta_e}}\Delta\delta_e + C_{D_{\delta_a}}\Delta\delta_a + C_{D_{\delta_r}}\Delta\delta_r \tag{2.43b}$$
$$C_Y = C_{Y_0} + C_{Y_\beta}\Delta\beta + C_{Y_p}\frac{pb}{2V_0} + C_{Y_r}\frac{rb}{2V_0} + C_{Y_{\delta_a}}\Delta\delta_a + C_{Y_{\delta_r}}\Delta\delta_r \tag{2.43c}$$
$$C_l = C_{l_0} + C_{l_\beta}\Delta\beta + C_{l_p}\frac{pb}{2V_0} + C_{l_r}\frac{rb}{2V_0} + C_{l_{\delta_a}}\Delta\delta_a + C_{l_{\delta_r}}\Delta\delta_r + C_{l_{W_y}}\dot{W}_y^B \tag{2.43d}$$
$$C_m = C_{m_0} + C_{m_V}\frac{\Delta V}{V_0} + C_{m_\alpha}\Delta\alpha + C_{m_{\dot{\alpha}}}\frac{\dot{\alpha}\bar{c}}{2V_0} + C_{m_q}\frac{q\bar{c}}{2V_0} + C_{m_{\delta_e}}\Delta\delta_e + ...$$
$$... + C_{m_{W_x}}\dot{W}_x^B + C_{m_{W_z}}\dot{W}_z^B \tag{2.43e}$$
$$C_n = C_{n_0} + C_{n_\beta}\Delta\beta + C_{n_p}\frac{pb}{2V_0} + C_{n_r}\frac{rb}{2V_0} + C_{n_{\delta_a}}\Delta\delta_a + C_{n_{\delta_r}}\Delta\delta_r + C_{n_{W_y}}\dot{W}_y^B \tag{2.43f}$$

where $k_{D/L}$ is defined by (2.8) or (2.9), $C_{l_{W(.)}}$, $C_{l_{W(.)}}$, $C_{n_{W(.)}}$ are the coefficients of rotational response to the airflow variation, and:

$$C_{a_V} = V_0 \left. \frac{\partial C_a}{\partial V} \right|_0 \tag{2.44a}$$

$$C_{a_\alpha} = \left. \frac{\partial C_a}{\partial \alpha} \right|_0 \tag{2.44b}$$

$$C_{a_{\dot\alpha}} = \frac{2V_0}{\bar{c}} \left. \frac{\partial C_a}{\partial \dot\alpha} \right|_0 \tag{2.44c}$$

$$C_{a_q} = \frac{2V_0}{\bar{c}} \left. \frac{\partial C_a}{\partial q} \right|_0 \tag{2.44d}$$

$$C_{a_{\delta_e}} = \left. \frac{\partial C_a}{\partial \delta_e} \right|_0 \tag{2.44e}$$

$$with\ a = L, D, m,$$

$$C_{a_\beta} = \left. \frac{\partial C_a}{\partial \beta} \right|_0 \tag{2.45a}$$

$$C_{a_p} = \frac{2V_0}{b} \left. \frac{\partial C_a}{\partial p} \right|_0 \tag{2.45b}$$

$$C_{a_r} = \frac{2V_0}{b} \left. \frac{\partial C_a}{\partial r} \right|_0 \tag{2.45c}$$

$$C_{a_{\delta_a}} = \left. \frac{\partial C_a}{\partial \delta_a} \right|_0 \tag{2.45d}$$

$$C_{a_{\delta_r}} = \left. \frac{\partial C_a}{\partial \delta_r} \right|_0 \tag{2.45e}$$

$$with\ a = Y, l, n.$$

2.4.7.1 Simulation Equations

The *full 3D dynamics* simulations for this thesis are based on two sets of parameters. The simplest is based directly on the stability derivatives (2.44) and (2.45). If the stability derivatives for an aircraft are not known, we use geometry based equations and parameters. These are referenced by Nelson [2].

2.4.8 Propeller/Motor Propulsion Model

This model describes the dynamics of a propulsion system combining a motor with a fixed pitch propeller. This is the most common propulsion system in small UAVs, as it is well suited for low speed and low fuel consumption.

In general aviation it is more common to use variable pitch propellers. These provide better efficiency for a wide range of flight speeds. Fixed pitch propellers provide the maximum efficiency only at a specific flight speed, usually selected to be the aircraft cruise speed. Their advantage is in their integration and maintenance simplicity.

The propeller blades work like small wings (fig. 2.6). The propeller thrust results from the aerodynamic lift and the aerodynamic drag results in the resistant torque.

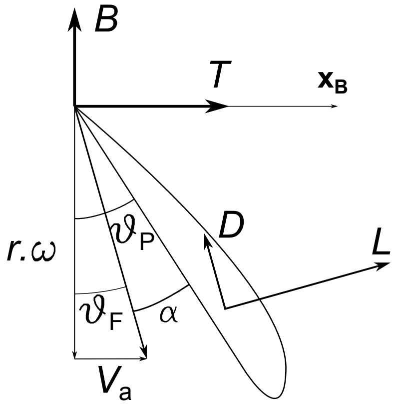


Figure 2.6: Propeller forces and torque diagram. T is the propeller thrust, B is the resistant torque, L and D are the aerodynamic lift and drag forces, ϑ_P is the propeller advance angle, $\vartheta_F = \tan^{-1}(V_a/r\omega)$ is the flow advance angle, $\alpha_P = \vartheta_P - \vartheta_F$ is the propeller angle-of-attack, r is the a radius over the propeller blade, ω is the propeller angular rate, V_a is the aircraft air speed, and $\mathbf{x}_B$ is the body frame longitudinal axis.

Each section of the propeller blades contributes to the overall thrust and resistant torque. These may be computed by integrating the lift and drag contributions along the blades:

$$T = N_B \int_0^{R_P} C_T \rho \frac{V_a^2 + \omega^2 r^2}{2} c(r)\, dr \tag{2.46a}$$

$$B_P = N_B \int_0^{R_P} r C_B \rho \frac{V_a^2 + \omega^2 r^2}{2} c(r)\, dr, \tag{2.46b}$$

where N_B is the number of propeller blades, R_P is the propeller blade radius, $c(r)$ is the blade section chord, and with

$$C_T = C_L \cos\vartheta_F - C_D \sin\vartheta_F \tag{2.47a}$$

$$C_B = C_D \cos\vartheta_F + C_L \sin\vartheta_F \tag{2.47b}$$

$$\alpha = \vartheta_P - \tan^{-1}\frac{V_a}{\omega r} \tag{2.47c}$$

$$\cos\vartheta_F = \frac{\omega r}{V_a^2 + \omega^2 r^2} \tag{2.47d}$$

$$\sin\vartheta_F = \frac{V_a}{V_a^2 + \omega^2 r^2}, \tag{2.47e}$$

yielding:

$$T = \frac{\rho N_B}{2} \int_0^{R_P} \left(C_L \omega r - C_D V_a\right) \sqrt{V_a^2 + \omega^2 r^2} c\left(r\right) dr \tag{2.48a}$$

$$B_P = \frac{\rho N_B}{2} \int_0^{R_P} r \left(C_D \omega r + C_L V_a\right) \sqrt{V_a^2 + \omega^2 r^2} c\left(r\right) dr. \tag{2.48b}$$

Propellers usually present an advance angle (ϑ_P) variable with the blade section radius. It increases for smaller radius to compensate the decreasing section speed (ωr) and maintain an near constant angle-of-attack. Fixed pitch propellers are usually classified by *Pitch* and diameter: $D \times Pitch$. *Pitch* is defined as the distance the propeller would advance in a complete rotation if it was slicing through a "solid" material. As such, the propeller advance angle can approximated by:

$$\vartheta_P = \tan^{-1} \frac{Pitch}{2\pi r} \tag{2.49}$$

We first approximated (2.48) by a summation of the forces from several section partitions. We assumed a propeller with elliptical section chords and no flow interference between propeller blades.

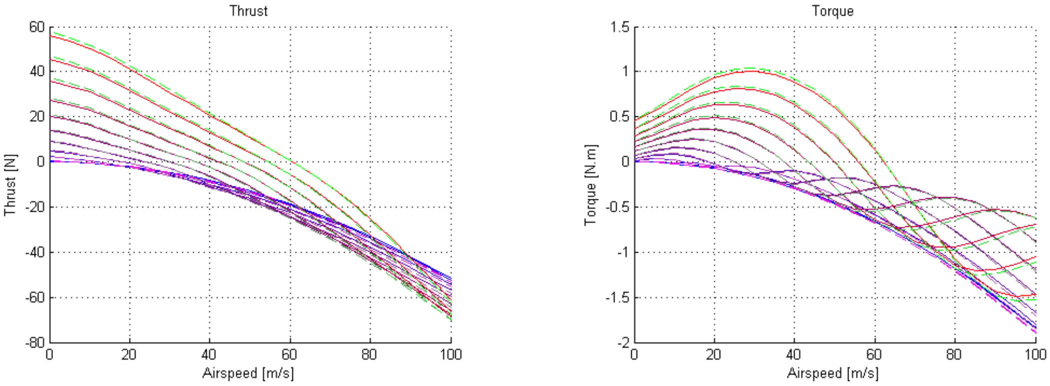
$$T = \frac{\rho N_B}{2} \sum_{i=1}^{N} \frac{R_P}{N} \left(C_L \omega \frac{i-0.5}{N} R_P - C_D V_a\right) \sqrt{V_a^2 + \left(\omega \frac{i-0.5}{N} R_P\right)^2} c\left(r\right) \tag{2.50a}$$

$$B_P = \frac{\rho N_B}{2} \sum_{i=1}^{N} \frac{i-0.5}{N^2} R_P^2 \left(C_D \omega \frac{i-0.5}{N} R_P + C_L V_a\right) \sqrt{V_a^2 + \left(\omega \frac{i-0.5}{N} R_P\right)^2} c\left(r\right) \tag{2.50b}$$

$$c\left(r\right) = c_{max} \sqrt{1 - \left(\frac{i-0.5}{N}\right)^2}, \tag{2.50c}$$

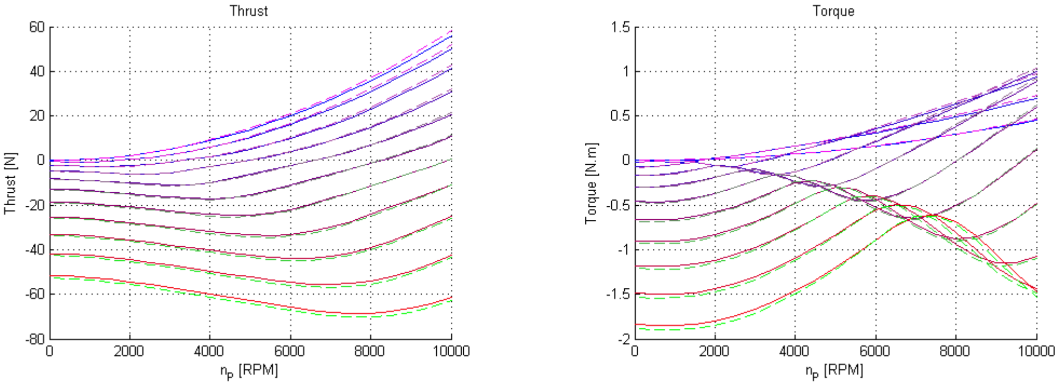
where c_{max} is the maximum chord, near the propeller hub. The aerodynamic coefficients throughout the blade sections were defined as in (2.11). As figures 2.7 and 2.8 illustrate, this approximation method is consistent even for a small number of partitions. The results remain almost the same for any number of partitions above 8.

Another approximation to (2.48) is to assume that there is a representative section of the blades at an effective radius R_{Ef}, that may be different for the thrust approximation and the torque approximation. The section aerodynamic forces may



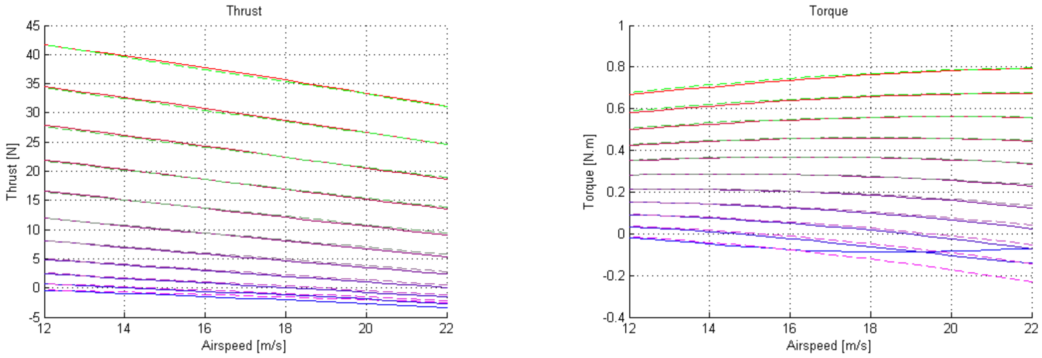
(a) Propeller thrust (b) Propeller torque

Figure 2.7: Propeller thrust and torque curves with the airspeed, for a set of angular rates. The illustrated angular rates range from 0 to 10000 RPM. The solid lines represent the thrust and torque predictions of (2.50) with 100 section partitions, which should represent (2.48) quite accurately. The dashed lines use the same approximation but with only 4 partitions.



(a) Propeller thrust (b) Propeller torque

Figure 2.8: Propeller thrust and torque curves with the angular rate (in RPM), for a set of airspeeds. The illustrated airspeeds range from 0 to 100 m/s. The solid lines represent the thrust and torque predictions of (2.50) with 100 section partitions, which should represent (2.48) quite accurately. The dashed lines use the same approximation but with only 4 partitions.



(a) Propeller thrust

(b) Propeller torque

Figure 2.9: Propeller thrust and torque curves with the airspeed, for a set of angular rates. The illustrated angular rates range from 0 to 10000 RPM. The approximated model (2.51) is illustrated in dashed lines.

be approximated by the lift and drag equations (2.6) and (2.7). This results in:

$$
\begin{aligned}
T &= \rho k_P \left(C_L \omega R_{Ef} - C_D V_a \right) && \text{(2.51a)}\\
B_P &= \rho k_P \left(C_D \omega R_{Ef} + C_L V_a \right) R_{Ef} && \text{(2.51b)}\\
C_L &= C_{L0} + \frac{dC_L}{d\alpha}\alpha && \text{(2.51c)}\\
C_D &= C_{D0} + k_{D/L} C_L^2 && \text{(2.51d)}\\
\alpha &= \tan^{-1}\frac{Pitch}{2\pi R_{Ef}} - \tan^{-1}\frac{V_a}{\omega R_{Ef}}, && \text{(2.51e)}
\end{aligned}
$$

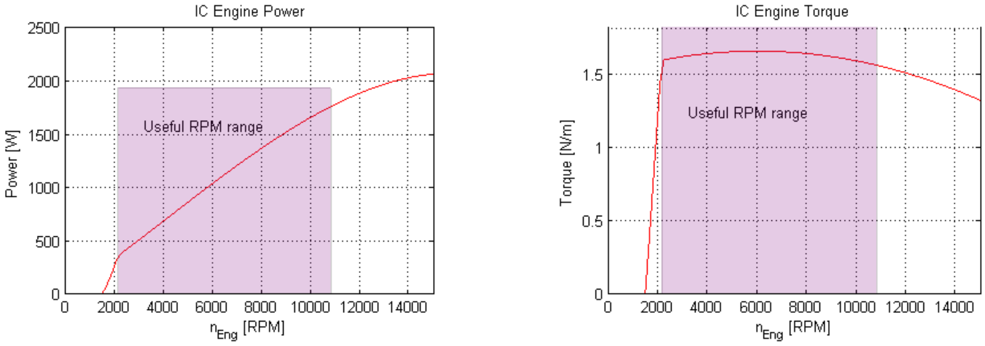
In our work the effective radius to approximate the thrust was very similar to the one used for the torque approximation. For the thrust approximation the error was mostly constrained to less than *1%* for an effective radius of *65%* of the propeller radius. The torque approximation error was mostly smaller than *1%* for an effective radius of *64%* of the propeller radius.

We will now define the motorization model. To generate thrust the propeller needs a driving motor to overcome the resistant torque. Small UAVs usually have Internal Combustion (IC) engines, also called reciprocating engines, or electric motors. The power and torque curves for both motorizations are quite different. IC engines show a near-constant torque for most of the useful RPM regimes (fig. 2.10). Electric motors provide a linearly decreasing torque with the RPM, generating the maximum torque when still (fig. 2.11).

An IC engine's torque and power may be approximated by:

$$
\begin{aligned}
B_M &\approx B_{max} - k_\omega \left(\frac{\omega}{\omega_{BMax}} - 1 \right)^2 && \text{(2.52a)}\\
P &= B\omega. && \text{(2.52b)}
\end{aligned}
$$

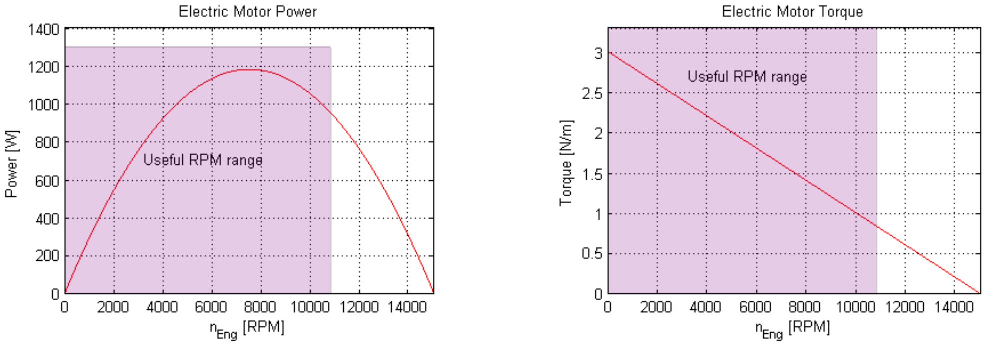
where B_{max} is the maximum torque, k_ω defines the deviation from the constant torque curve and ω_{BMax} is the RPM value at which the torque is maximum.



(a) IC engine power

(b) IC engine torque

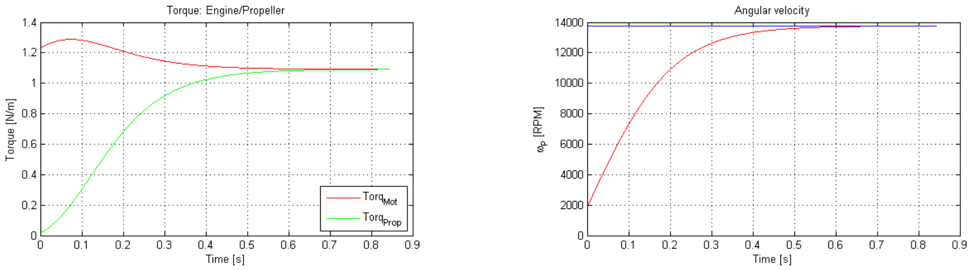
Figure 2.10: IC engine power and torque curves, with useful RPM range, i.e., from engine stall to the maximum achievable RPM in level flight.



(a) Electric motor power

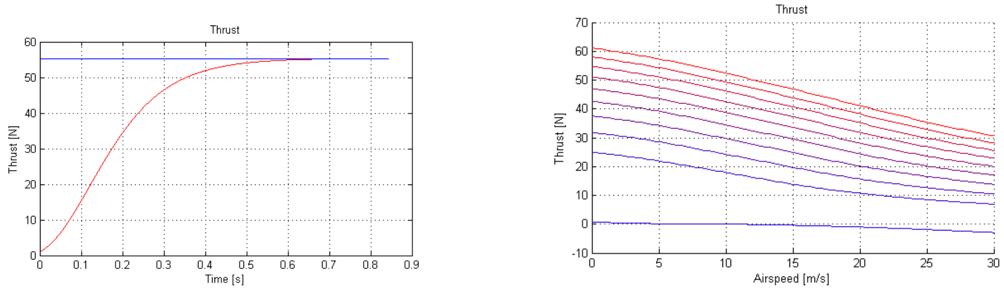
(b) Electric motor torque

Figure 2.11: Electric motor power and torque curves, with useful RPM range, i.e., from null RPM to the maximum achievable RPM in level flight.



(a) Torque time evolution with a propeller/motor pair

(b) Angular rate time evolution with a propeller/motor pair



(c) Thrust time evolution with a propeller/motor pair

(d) Thrust generated by a propeller/motor pair over varying airspeed at different throttle settings.

Figure 2.12: Dynamics of a propeller/motor pair.

An electric motor's torque and power curves are better approximated by:

$$B_M \approx B_{max} - k_\omega \omega \tag{2.53a}$$

$$P \approx B_{max}\omega - k_\omega \omega^2. \tag{2.53b}$$

where k_ω is the proportion with which the torque decreases with increasing RPM.

The element connecting both the propeller and the motor is the angular rate dynamics, which are governed by the excess torque, i.e., the difference between the generated and the resistant torque, damped by the propeller inertia I_P:

$$\dot{\omega} = \frac{B_M - B_P}{I_P}. \tag{2.54}$$

Figure 2.12 illustrates the dynamics of a motor/propeller pair. As expected, the angular rate grows (fig. 2.12b) until the torque excess is null (fig. 2.12a). During the transitory state the angular rate presents a first-order response (fig. 2.12b), while the thrust presents a second-order response (fig. 2.12c).

2.4.9 Simplified Propulsion Model

This propulsion model is a simplification of the propeller/motor propulsion system. The necessity for this simpler model comes from the propulsion parameter identification algorithm. From figure 2.12 it is clear that for some velocity ranges the thrust can be approximated by a linear function without introducing large errors:

$$T = K_T \delta_T - K_V \frac{\Delta V_a}{V_{a,ref}} \delta_T, \tag{2.55}$$

where δ_T is the throttle setting, $V_{a,ref}$ is the airspeed reference value, usually the cruise speed, $\Delta V_a = V_a - V_{a,ref}$ is a velocity variation, and K_T and K_V are the propulsion model parameters.

2.5 Environment Models

An important issue for aircraft operation is the energy consumption. Aircraft can use updrafts, gusts and wind gradients to diminish fuel consumption. Small Unmanned Air Vehicles (UAVs) in particular, can take great benefits from these phenomena, due to their low flight airspeeds and good maneuverability.

In this work the flow field is the spatial characterization of the air velocity vector.

Definition II.1. A flow field is defined at a given position $\mathbf{x} = [x, y, z]^\intercal$ by the air flow velocity vector $\mathbf{w}$ and gradient matrix $\mathbf{J}_W$:

$$\mathbf{w} := \begin{bmatrix} W_x \\ W_y \\ W_z \end{bmatrix} \tag{2.56a}$$

$$\mathbf{J}_W := \begin{bmatrix} \frac{\partial W_x}{\partial x} & \frac{\partial W_x}{\partial y} & \frac{\partial W_x}{\partial z} \\ \frac{\partial W_y}{\partial x} & \frac{\partial W_y}{\partial y} & \frac{\partial W_y}{\partial z} \\ \frac{\partial W_z}{\partial x} & \frac{\partial W_z}{\partial y} & \frac{\partial W_z}{\partial z} \end{bmatrix} . \tag{2.56b}$$

Unless stated otherwise, the flow field variables will be written in the ground reference frame, meaning that W_x is the flow velocity component towards the North, W_y is the flow velocity component towards the East, W_z is the downward flow velocity component.

We will refer to the flow field horizontal velocity vector as the wind vector.

Definition II.2. The wind vector is the flow field horizontal velocity vector at a given position:

$$\mathbf{w}_H := \begin{bmatrix} W_x \\ W_y \end{bmatrix}_{\mathbf{x}} \tag{2.57}$$

Further, we will refer to the flow field vertical velocity as updraft or downdraft, for a negative or positive vertical air flow, respectively.

Definition II.3. An updraft is the flow field vertical velocity at a given position:

$$\mathbf{w}_{Upd} := -\mathbf{w}_z = -W_z|_{\mathbf{x}} \tag{2.58}$$

2.5.1 Wind Shear

Wind shear is the atmospheric phenomenon which occurs on thin layers separating two regions where the predominant air flow is different. This difference can be either in speed, in direction, or in both speed and direction. The air layer between these regions is usually under 100 meters thick, resulting in a persistent gradient in the flow field. This gradient may be exploited by UAVs [3, 9, 42] as it is by birds [8, 24, 25].

Generally, the wind shear phenomena can be classified by directionality as horizontal or vertical shear. Horizontal wind shear is a variation of the air flow with altitude. It exists near a surface (ground or water) [8, 24, 25], over inversion layers, on the limits of the jet stream [37] and over geographic obstacles [42]. Vertical wind shear is a variation of the air flow with the horizontal position $[x, y]$. It appears across weather fronts, near the coast, and in the vicinity of mountain ranges.

We will focus on horizontal wind shear, as surface, inversion, and jet stream shear are quite steady phenomena [43, 8, 7]. In horizontal wind shear we distinguish *Surface*, *Layer* and *Ridge Wind Shear*, as the flow gradient is different for each phenomenon. The first two phenomena take place over large areas, which makes them difficult to characterize as a whole. For that reason our approach is to simplify the phenomena to uniaxial (z) wind vector variations. The *Ridge Wind Shear* depends greatly on the distance to the ridge crest, and so its model is defined over the plane perpendicular to the ridge.

For the sake of simplicity, and because in this section we regard only the horizontal flow, we will refer to the wind vector as $\mathbf{w}$ and to its total speed as W.

2.5.1.1 Surface Wind Shear

Surface Wind Shear is a special case of horizontal shear where instead of two air mass regions we have one air mass region and a surface. The surface is usually still or moving at very low speeds relatively to the general air mass, as is the case of water surfaces. *Surface Wind Shear* enables albatross to fly thousands of kilometers over the ocean almost without flapping their wings [24, 25]. *Surface Wind Shear* is also known in the aviation community mainly by its effects on aircraft landing and take-off operations. The reduction of flow speed towards the ground causes the aircraft airspeed to decrease in the same amount, if no compensation is applied. This effect can induce stall, leading to possibly catastrophic results. The wind shear layer starts at the ground level and may be modeled by [8, 44]:

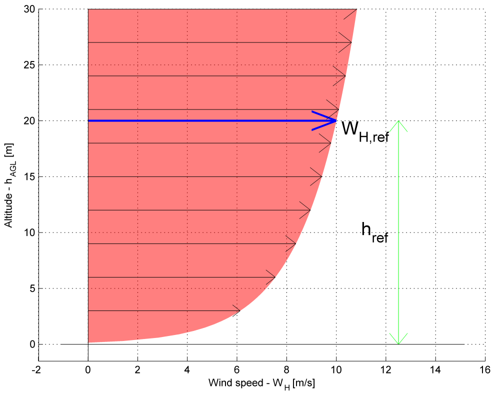


Figure 2.13: *Surface Wind Shear* profile

$$W = W_{h_{ref}} \frac{\ln\left(h/h_0\right)}{\ln\left(h_{ref}/h_0\right)}, \tag{2.59}$$

where W is the total wind speed and $W_{h_{ref}}$, h_{ref}, and h_0 are reference values. $W_{h_{ref}}$ is the reference wind speed at a reference altitude h_{ref} away from the surface. h_0 defines the shape of the flow gradient, reflecting the surface properties, like irregularity, roughness and drag. In the Military Specification MIL-F-8785C [44]:

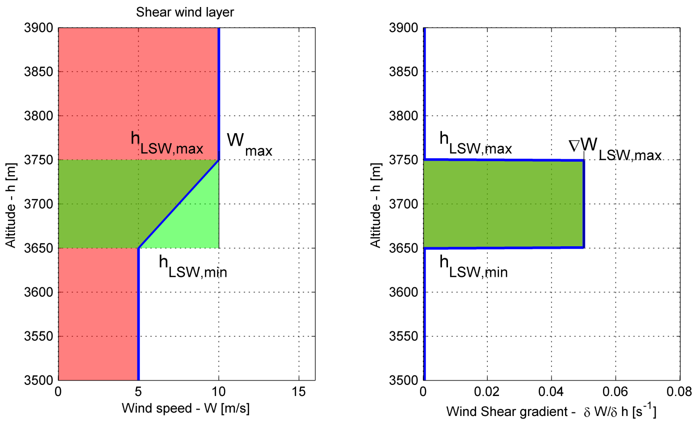
$$\begin{aligned} h_{ref} &= 6m \\ h_0 &= \begin{cases} 0.15 & for\,Category\,C\,flight\,phases \\ 2.0 & otherwise \end{cases}, \end{aligned} \tag{2.60}$$

if $1m \leq h \leq 300m$. Category C flight phases are the terminal flight phases, which include takeoff, approach, and landing, as defined in reference [44].

2.5.1.2 Layer Wind Shear

Layer Wind Shear is the most general wind shear type. It can represent both a horizontal or a vertical wind shear, although in this work we will focus on the horizontal shear. Two of the most common atmospheric phenomena associated with *Layer Wind Shear* are the Inversion Layer and the Jet Stream. An Inversion Layer is characterized by, as the name indicates, an inversion of the temperature gradient with altitude. Often the Convective or Mixed-Layer, the lowest in the atmosphere, is separated from the upper Troposphere layers by an Inversion Layer. If the Inversion Layer is thin enough and the flow of the separated air masses is different enough, the generated gradient may be strong enough to provide aircraft the energy necessary to maintain flight. The Jet Stream phenomenon is characterized by a region of high-speed winds. The regions between the Jet Stream core and slower wind currents exhibit a wind gradient [7]. Glider pilots observe *Layer Wind Shear* sometimes above 3 knots per 1000 feet [43].

Sachs and da Costa [7] defined a model for the *Layer Wind Shear* observed below Jet Stream regions. The model presents a constant wind gradient, with the wind speed converging to the Jet Stream speed:



(a) Wind speed profile (b) Wind speed gradient

Figure 2.14: Sach-daCosta *Layer Wind Shear* model.

$$W\left(h\right) = W_{JetStream} + \frac{\Delta W_{LWS}}{\Delta h}\left(h - h_{JetStream}\right), \tag{2.61}$$

where $W_{JetStream}$ is the Jet Stream speed, $h_{JetStream}$ is the lower boundary of the Jet Stream core, and $\frac{\Delta W_{LWS}}{\Delta h}$ is the wind shear vertical gradient. This model does not represent the wind gradient over the transition regions, i.e., the regions where the wind speed stabilizes (fig. 2.17).

We now present a simple model for the *Layer Wind Shear* with gradient transition regions. We approximate the wind gradient with a Gaussian. This model shows the expected convergence to the boundary wind speeds both over the lower and top layer limits. The model wind speed profile is illustrated in figure 2.15a and defined by:

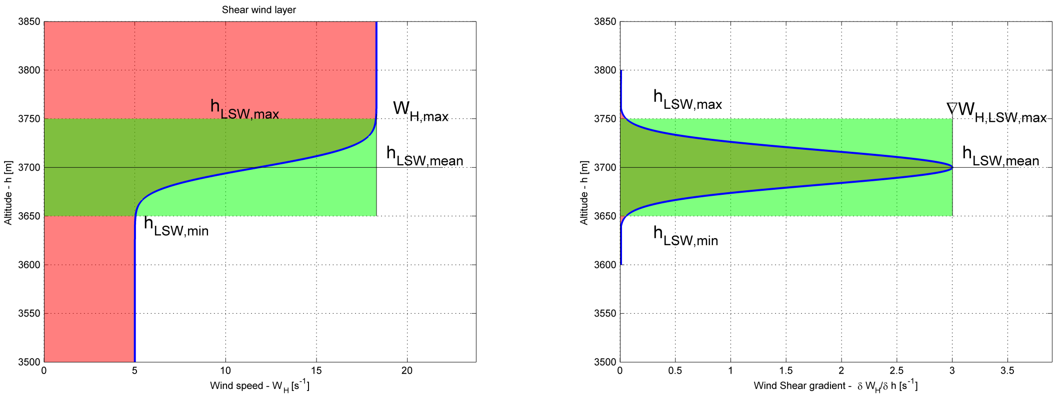
$$\mathbf{w}\left(h\right) = \mathbf{w}\left(h_{min}\right) + \frac{\Delta \mathbf{w}_{LWS}}{2}\left[1 + \operatorname{erf}\left(4\frac{h - \overline{h}_{LWS}}{\Delta h_{LWS}}\right)\right], h \in \left[h_{min}, h_{max}\right], \tag{2.62}$$

where h_{min} and h_{max} are the wind shear layer limit altitudes, $\mathbf{w}\left(h\right)$ is the wind vector due to the wind shear phenomenon at an altitude h, and:

$$\Delta \mathbf{w}_{LWS} = \mathbf{w}\left(h_{max}\right) - \mathbf{w}\left(h_{min}\right) \tag{2.63a}$$

$$\overline{h}_{LWS} = \frac{h_{max} + h_{min}}{2} \tag{2.63b}$$

$$\Delta h_{LWS} = h_{max} - h_{min}. \tag{2.63c}$$



(a) Wind speed profile

(b) Wind speed gradient

Figure 2.15: Gaussian *Layer Wind Shear* model.

The wind gradient is illustrated in figure 2.15b and modeled by:

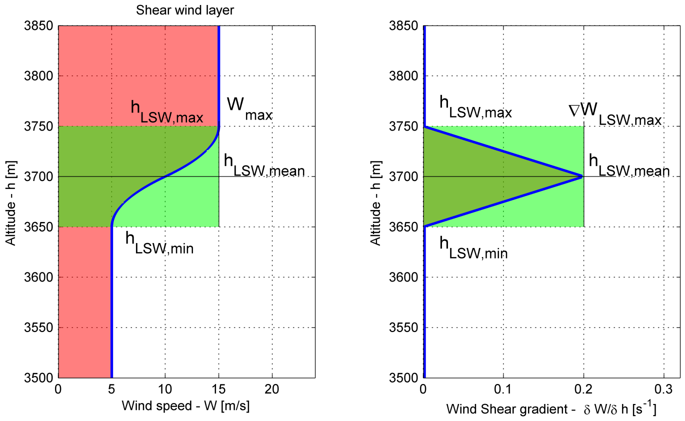
$$\left.\frac{\delta \mathbf{w}_{LWS}}{\delta h}\right|_{h} = \frac{4\left\|\Delta \mathbf{w}_{LWS}\right\|}{\Delta h_{LWS}\sqrt{\pi}} e^{-\left(4\frac{h - \overline{h}_{LWS}}{\Delta h_{LWS}}\right)^2}. \tag{2.64}$$

An alternative model for the *Layer Wind Shear* is an approximation by a quadratic function:

$$\mathbf{w}\left(h\right)=\begin{cases}\mathbf{w}\left(h_{min}\right) & h\leq h_{min}\\ \mathbf{w}\left(h_{min}\right)+2\Delta\mathbf{w}_{LWS}\left(\frac{h-h_{min}}{\Delta h_{LWS}}\right)^{2} & h\in\left(h_{min},\overline{h}_{LWS}\right]\\ \mathbf{w}\left(h_{max}\right)-2\Delta\mathbf{w}_{LWS}\left(\frac{h_{max}-h}{\Delta h_{LWS}}\right)^{2} & h\in\left(\overline{h}_{LWS},h_{min}\right)\\ \mathbf{w}\left(h_{max}\right) & h\geq h_{max}.\end{cases} \tag{2.65a}$$

$$\left.\frac{\delta\mathbf{w}_{LWS}}{\delta h}\right|_{h}=\begin{cases}0 & h\leq h_{min}\\ 4\Delta\mathbf{w}_{LWS}\frac{h-h_{min}}{\Delta h_{LWS}^{2}} & h\in\left(h_{min},\overline{h}_{LWS}\right]\\ 4\Delta\mathbf{w}_{LWS}\frac{h_{max}-h}{\Delta h_{LWS}^{2}} & h\in\left(\overline{h}_{LWS},h_{min}\right)\\ 0 & h\geq h_{max}.\end{cases} \tag{2.65b}$$

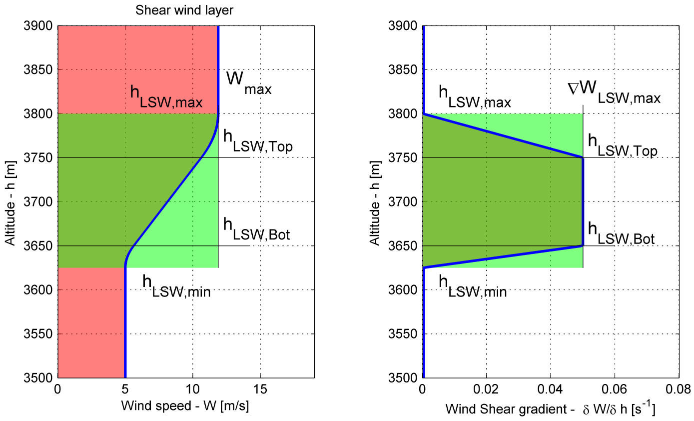
As the Gaussian model, the quadratic model converges to the boundary wind speeds at the limits (fig. 2.16a). Its advantages over the Gaussian model are the easier computation and smooth transitions at the layer limits.



(a) Wind speed profile (b) Wind speed gradient

Figure 2.16: Quadratic *Layer Wind Shear* model.

Both the Gaussian and the quadratic models for the *Layer Wind Shear* include regions representing a gradient transition in the bottom and the top of the wind shear, and an almost constant gradient region at the center of the wind shear. Sach and da Costa [7] report the existence of large regions with a constant vertical gradient. We now extend the quadratic *Layer Wind Shear* model to include a linear part. This linear section of the wind shear represents a variable sized region with constant vertical gradient. Further, this model allows a definition of gradient transition regions with different sizes (fig. 2.17).



(a) Wind speed profile (b) Wind speed gradient

Figure 2.17: Linear and quadratic *Layer Wind Shear* model.

$$\mathbf{w}\left(h\right)=\begin{cases}\mathbf{w}\left(h_{min}\right) & h\leq h_{min}\\ \mathbf{w}\left(h_{min}\right)+\left(\frac{\delta\mathbf{w}}{\delta h}\right)_{max}\frac{\left(h-h_{min}\right)^{2}}{2\Delta h_{LWS,Bot}} & h\in\left(h_{min},h_{LWS,Bot}\right)\\ \mathbf{w}\left(h_{LWS,Bot}\right)+\left(\frac{\delta\mathbf{w}}{\delta h}\right)_{max}\left(h-h_{LWS,Bot}\right) & h\in\left[h_{LWS,Bot},h_{LWS,Top}\right]\\ \mathbf{w}\left(h_{max}\right)-\left(\frac{\delta\mathbf{w}}{\delta h}\right)_{max}\frac{\left(h_{max}-h\right)^{2}}{2\Delta h_{LWS,Top}} & h\in\left(h_{LWS,Top},h_{min}\right)\\ \mathbf{w}\left(h_{max}\right) & h\geq h_{max}.\end{cases}\tag{2.66a}$$

$$\left.\frac{\delta\mathbf{w}_{LWS}}{\delta h}\right|_{h}=\begin{cases}0 & h\leq h_{min}\\ \left(\frac{\delta\mathbf{w}}{\delta h}\right)_{max}\frac{h-h_{min}}{\Delta h_{LWS,Bot}} & h\in\left(h_{min},h_{LWS,Bot}\right)\\ \left(\frac{\delta\mathbf{w}}{\delta h}\right)_{max} & h\in\left[h_{LWS,Bot},h_{LWS,Top}\right]\\ \left(\frac{\delta\mathbf{w}}{\delta h}\right)_{max}\frac{h_{max}-h}{\Delta h_{LWS,Top}} & h\in\left(h_{LWS,Top},h_{min}\right)\\ 0 & h\geq h_{max},\end{cases}\tag{2.66b}$$

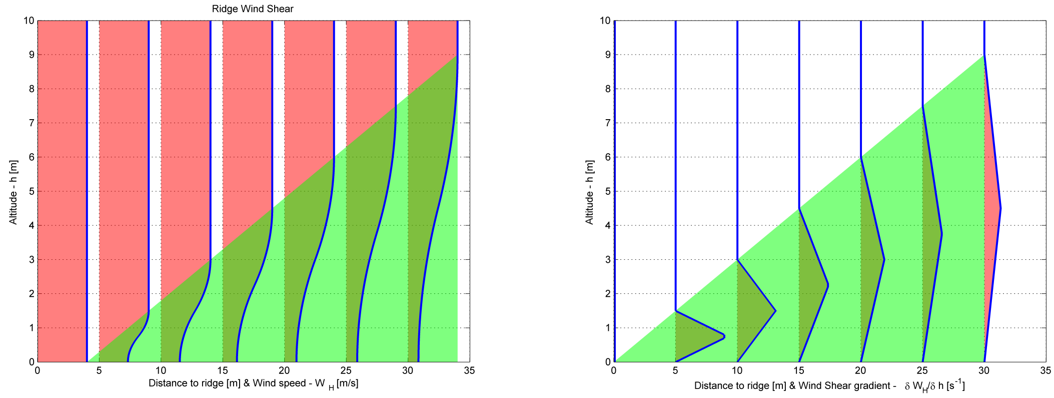
where $\left(\frac{\delta\mathbf{w}}{\delta h}\right)_{max}$ is the maximum vertical gradient of the wind shear, $h_{LWS,Bot}$ is the maximum altitude of the bottom gradient transition region, $h_{LWS,Top}$ is the minimum altitude of the top gradient transition region, $\Delta h_{LWS,Bot}$ is the thickness of the bottom gradient transition region, $\Delta h_{LWS,Top}$ is the thickness of the top gradient transition region, and $\mathbf{w}\left(h_{LWS,Bot}\right)=\mathbf{w}\left(h_{min}\right)+\left(\frac{\delta\mathbf{w}}{\delta h}\right)_{max}\Delta h_{LWS,Bot}/2$.

2.5.1.3 Ridge Wind Shear

The *Ridge Wind Shear* is a special case of the *Layer Wind Shear*, which appears on the leeward side of a mountain. It is generated when the free moving air finds a large obstacle (the ridge) and, while flowing over it, is not capable of accelerating instantaneously the air mass behind the ridge. This wind shear type is the most used by radio controlled gliders, as it is strong enough, maintains a constant position and the trajectory required to use it is safe enough to avoid ground collisions [42]. From Parle's wind velocity measurements [42] it is clear that the vertical gradient appears over the leeward side of the ridge. An hypothetical model which approximately fits Parle's data is defined as:

$$\mathbf{w}_{RWS} = \mathbf{w}_\infty e^{-\frac{\lambda x}{W_\infty}} + \left(1 - e^{-\frac{\lambda x}{W_\infty}}\right) \cdot \mathbf{w}_{LWS}\left(h\right)\Big|_{\substack{h_{min} = h_{Ridge} \\ h_{max} = h_{Ridge} + k_1 x \\ \mathbf{w}\left(h_{min}\right) = k_2 \mathbf{w}_\infty \\ \mathbf{w}\left(h_{max}\right) = \mathbf{w}_\infty}} , \tag{2.67}$$

where $\mathbf{w}_{LWS}\left(h\right)$ is the *Layer Wind Shear* gradient model. In this model the gradient strength varies as a first order response with the spatial coefficient λ. The gradient thickness is modeled as proportional (k_1) to the distance from the ridge crest. The gradient bottom wind speed is proportional (k_2) to the undisturbed wind speed $(\mathbf{w}_\infty)$. Note that this model has 3 degrees-of-freedom: λ, k_1, and k_2. k_1 should be similar to the terrain slope on the windward side of the ridge. λ should depend on the ridge abruptness. A sharp ridge should yield a larger λ. k_2 probably depends mainly on the terrain slope on the ridge leeward side.



(a) Wind speed profile

(b) Wind speed gradient

Figure 2.18: Variation of wind speed over a *Ridge Wind Shear* with altitude and distance to the ridge crest.

Figure 2.18 illustrates the wind speed variation within the *Ridge Wind Shear*. Notice the decrement of the minimum wind speed with the distance to the ridge crest, modeled by the first order response. It is also clear that the wind gradient is stronger closer to the ridge, indicating that this is the best area to harvest energy, as experienced by the radio controlled glider pilots.

2.5.1.4 Generic Wind Shear

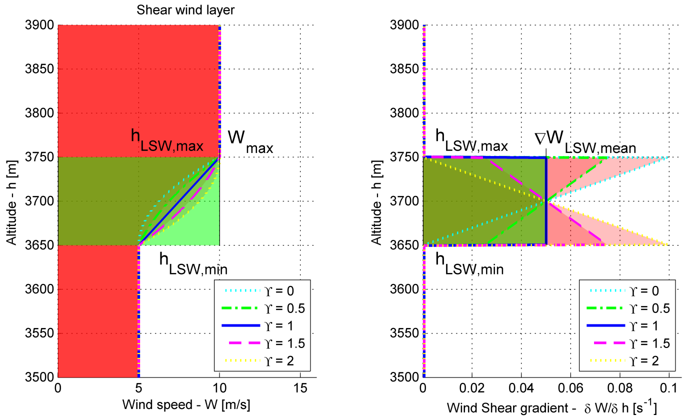
Zhao [6] proposed a wind shear model that is able to represent a linear gradient, as well as an exponential-like gradient or a logarithmic-like gradient. This model defines a quadratic wind speed profile with an average slope $(\frac{\Delta W}{\Delta h})$ over an altitude range $([0, h_{max}])$:

$$W = \frac{\Delta W}{\Delta h}\left[\Upsilon \cdot h + \frac{1-\Upsilon}{\Delta h} \cdot h^2\right], \tag{2.68}$$

where $\Delta h = h_{max} - h_{min}$ and Υ defines the profile shape. Υ is required to remain within $[0, 2]$, to keep the wind profile variation within $[0, \Delta W]$. $\Upsilon = 1$ results in a constant vertical gradient. $0 \leq \Upsilon < 1$ yields an exponential-like wind profile. And $1 < \Upsilon \leq 2$ yields a logarithmic-like wind profile.

This model may be extended to a 2D wind vector profile with an average variation of $\frac{\Delta \mathbf{w}}{\Delta h}$ over an altitude range ($[h_{min}, h_{max}]$):

$$\mathbf{w} = \mathbf{w}\left(h_{min}\right) + \frac{\Delta \mathbf{w}}{\Delta h}\left[\Upsilon\left(h - h_{min}\right) + \frac{1-\Upsilon}{\Delta h}\left(h - h_{min}\right)^2\right], \tag{2.69}$$



(a) Wind speed profile (b) Wind speed gradient

Figure 2.19: Zhao *Generic Wind Shear* quadratic model.

This model may approximate a *Surface Wind Shear* with $1 < \Upsilon \leq 2$. It may also approximate the linear region of a *Layer Wind Shear* with $\Upsilon = 1$ and the transition regions with $0 \leq \Upsilon < 1$.

2.5.1.5 Wind Shear Models Comparison

We now summarize the main similarities and differences among the presented wind shear models. The models diverge in the vertical gradient variation with altitude, the bounding altitudes and the definition of a horizontal gradient.

Table 2.1: Wind Shear Models Comparison

Model	*WS type*^{1*}	*Vert. grad.*^{2*}	*Low. alt. bound*^{3*}	*Spatial def.*^{4*}
Surface WS [44]	SWS^{5*}	Logarithmic	Surface	1D
Sachs-daCosta LWS^{9*}	LWS^{6*}	Linear	Air flow layer	1D
Bencatel G. LWS^{10*}	LWS^{6*}	Gaussian	Air flow layer	1D
Bencatel LQ. LWS^{11*}	LWS^{6*}	Lin. & Quad.	Air flow layer	1D
Bencatel RWS^{12*}	RWS^{7*}	Quadratic	Ridge	2D
Zhao WS^{13*}	S/LWS^{8*}	Quadratic	Any^{14*}	1D

$1\star$ *Type of wind shear represented.*

2⋆ *Function type of the vertical gradient.*

3⋆ *Lower altitude bound of the wind shear phenomenon.*

4⋆ *Wind shear spatial definition: Only over the vertical axis (1D) or with a defined vertical and horizontal gradient (2D).*

5⋆ *Surface Wind Shear.*

6⋆ *Layer Wind Shear.*

7⋆ *Ridge Wind Shear.*

8⋆ *Surface Wind Shear or parts of a Layer Wind Shear.*

9⋆ *Sachs-daCosta Linear Layer Wind Shear.*

10⋆ *Bencatel Gaussian Layer Wind Shear.*

11⋆ *Bencatel Linear and Quadratic Layer Wind Shear*

12⋆ *Bencatel Ridge Wind Shear*

13⋆ *Zhao Wind Shear*

14⋆ *Surface or air flow layer*

2.5.2 Updraft

Updrafts are upward moving air masses. These air masses can be small upward gusts, created by turbulence and ranging from centimeters to a few meters. Updrafts can also be the result of huge rising air mass bodies, ranging from 50 meters to kilometers. These originate from terrain topography or from thermal flows. The first type, called orographic updrafts, are generated when the wind hits a terrain slope, creating strong updrafts above the terrain. The thermal flows are generated by hot spots on the ground. These create a thermal gradient, heating the surrounding air. The density of the heating air decreases, forcing an upward movement. Thermal updrafts don't depend as much on topography or on wind as orographic updrafts.

Thermals are part of the convection flows that develop in the mixing-layer of the atmosphere, the lowest atmospheric layer, also called convective layer. A thermal model may be characterized by three types of parameters: the atmosphere parameters, the internal parameters, and the terrain distribution. The atmosphere parameters characterize the regional environment characteristics. These concern the variables that are slow varying and very similar in adjacent flight areas. The internal parameters are individual to each thermal. These parameters define the spatial effects of each specific thermal, and may be distinct for different adjacent thermals. The terrain distribution concerns the sparsity among thermals, which affects the rate of appearance of thermals in the aircraft flight path.

There are several types of thermals. *Chimney Thermals*, also designated as *Column Thermals*, are continuous columns of rising air that extend from the ground surface right up to the mixing-layer maximum altitude. *Bubble Thermals* are closed shells [45] of rising air. They are formed near the ground when the temperature differences are large enough to create buoyancy, like in a balloon. These shells then rise and depart from the ground. Cone [45] describes them as rising vortex rings, where there is a circulatory flow generated by strong core updraft. This core updraft is fed by the buoyant air. When this air leaves the bubble core, it starts to cool down, becoming less buoyant. The cooled air then moves downward on the outside of the vortex ring, completing the cycle.

The main model features of a thermal are its position, the updraft field, representing how the vertical airspeed varies with the position relative to the thermal center, and its radius, defining where the updraft speed is null or almost null. The model we will present next differ on three main modeling features: the effect of altitude variation, the representation of the thermal skirt downdraft, and the interdependence among the updraft field and thermal radius.

2.5.2.1 Thermals

The most used thermal model is based on a scaled 2D Gaussian, for its simplicity [29, 11]. This is a simple model where the center of the thermal is the center of the Gaussian [27] (fig. 2.20):

$$w(r) = W_{z.max} \cdot e^{-(r/R_T)^2} \tag{2.70}$$

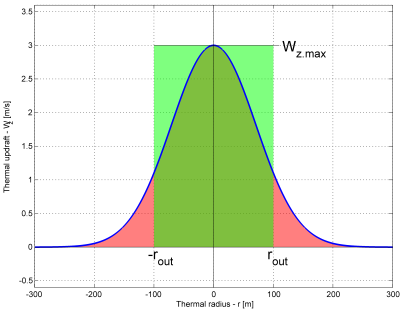


Figure 2.20: Thermal updraft Gaussian model

The Gaussian is scaled so that its maximum matches the maximum updraft (core updraft). Furthermore, the Gaussian variance is adjusted so that an almost null velocity is found at the thermal outer radius. Therefore, this model does not include the exterior downdraft and the thermal radius altitude dependence. However, the model is good enough to represent the thermal core at close altitudes.

Gedeon's Thermal model [9] is an extension of the Gaussian model. He adjusted the updraft function so that it presented a negative speed outside the thermal radius and a null speed at the thermal radius (fig. 2.21):

$$w(r) = W_{z.max} \cdot e^{-(r/R_T)^2} \cdot \left[1 - (r/R_T)^2\right] \tag{2.71}$$

This model only represents the thermal in the horizontal dimensions, because it does not present any dependency of the thermal diameter with the altitude. As such, it is a more realistic model of a thermal at close altitudes than the Gaussian model.

There are some references which argue that the *bell-shape* changes according to the size and strength of the thermal [46, 47, 48]. Observations lead to the hypothesis that there are two more prevalent thermal shape types [46, 47, 48], one with a pronounced maximum (*type "b"*) and another with several maxima (*type "a"*), almost as a plateau (fig. 2.22). According to the same source, *type "b"* thermals appear

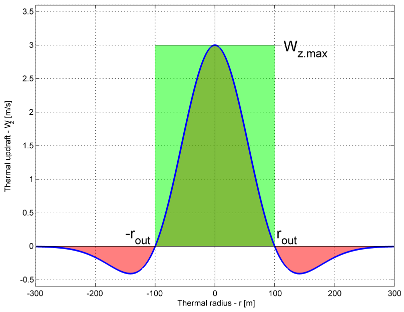


Figure 2.21: Thermal updraft Gedeon model

more frequently in lower energy environments. *Type "a"* is observed more frequently when the temperature gradient is larger, and seems to be the result of a merge between several *type "b"* thermals. Furthermore, Lenschow and Stephens [36] state that the magnitude of vertical velocity variation within a thermal may be larger than the magnitude of the overall mean updraft velocity.

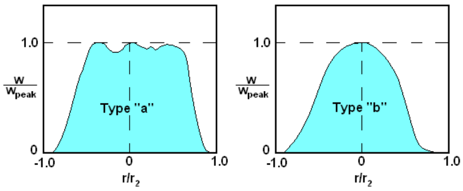


Figure 2.22: Updraft velocity magnitude

Lawrance and Sukkarieh [3] present a toroidal model as an hypothesis for a *Bubble Thermal* flow structure. It represents the vortex ring proposed by Cone [45] in all three dimensions. Further, the model represents the flow field in every direction instead of only the vertical flow.

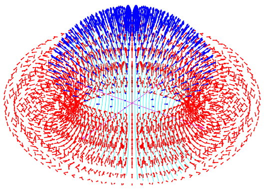


Figure 2.23: *Bubble Thermal*

Allen [4] presents a detailed updraft model developed at NASA Dryden Flight Research Center with real flow measurements. This model represents a continuous *Chimney Thermal* in all three dimensions. It relates the atmospheric parameters with their effect on the vertical flow at different altitudes. There is no prediction on the horizontal flow velocity, but the predicted vertical flow respects the mass conservation along any horizontal plane.

Thermals will lean or drift with the prevalent direction of the wind [43]. The organization in *chimney* or *bubbles* is affected by the existing wind shear. The fraction between the thermal strength and the wind shear plays an important role on the organization definition [43].

There are different thesii about which is the most prevalent type of thermals, *Chimney* (fig. 2.24) or *Bubbles* (fig. 2.23). [43] states that the *Chimney Thermal* type is the most prevalent kind of thermals, indicating that *Bubbles* appear when the heating is slow or intermittent. This may happen due to the radiated surface properties or due to moving cloud shadows. On the other side, Cone [45] defends that *Bubble Thermals* are the most frequent. This thesis is sustained by predictions of the vortex theory and experiments that attested to the formation of rising vortex rings. Furthermore, continuous *Chimney Thermals* would require a continuous air supply near the ground, which would be sensed as a continuous ground wind. However, the appearance of thermals is not usually associated with sustained wind. A more common phenomenon is sudden wind gusts, that can be associated with the formation of a *Bubble Thermal.* Groups of soaring birds take off in what seems like a reaction to these sudden gusts. Another observation that sustains the prevalence of *Bubble Thermals* is the fact that birds usually cluster in a short altitude range when soaring. Birds below the main cluster frequently have to flap their wings to reach the soaring group.

2.5.2.2 Chimney Thermal - Allen Model

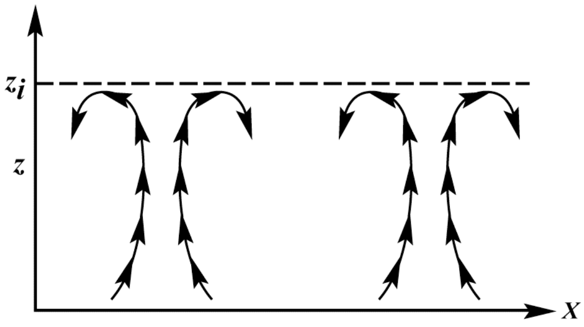


Figure 2.24: Chimney Thermal representation with mixing-layer thickness (z_i) (illustration from [4])

Allen [4] developed a thermal model which represents the updraft strength in all three dimensions. His model defines the radius and the updraft field at each altitude as a function of two atmospheric parameters: the mixing-layer thickness and the convective velocity scale. The mixing-layer thickness (z_i) is the mixing-layer maximum altitude (fig. 2.24), where the convection flows appear. As such, z_i is also the maximum altitude for the thermal activity. This depends on the ground temperature T_S and on the predawn air temperature profile [4] (fig. 2.25):

$$z_i = f\left(T_S, Predawn\ air\ temperature\ profile\right). \tag{2.72}$$

The mixing-layer thickness (z_i) varies slowly with the location, presenting a low spatial variation frequency. The thickness is usually 0 at dawn, rises to the day

Table 2.2: Yearly convective lift statistical properties (from reference [4])

Description	$w^\star$, m/s	$\mu_{z_i} - \sigma_{z_i}$, m	μ_{z_i}, m	$\mu_{z_i} + \sigma_{z_i}$, m
$\mu_{w^\star} - 2\sigma_{w^\star}$	0.46	25.6	53.6	97.4
$\mu_{w^\star} - \sigma_{w^\star}$	1.27	150	210	1007
$\mu_{w^\star}$	2.56	767	1401	2319
$\mu_{w^\star} + \sigma_{w^\star}$	4.08	2134	2819	3638
$\mu_{w^\star} + 2\sigma_{w^\star}$	5.02	2913	3647	4495

maximum at late afternoon [46] and returns to 0 during the night. Therefore, the regional variation time constant should be high, on the order of hours.

The convective velocity scale ($w^\star$) is a reference which indicates the predicted velocity magnitudes in and around a thermal. Again reference [4] shows that this velocity is a function of the mixing-layer thickness (z_i), the surface virtual potential temperature flux (Q_{OV}), the ground temperature T_S, and the static pressure at ground level (p) [4]. Further, the virtual potential temperature flux is a function of the net radiation at the surface (Q_S), the air relative humidity (rh), the saturated vapor pressure (es), and the ground temperature T_S.

$w^\star$ seems to have a spatial and temporal dependency similar to z_i. The only extra factor is the wind speed, which seems to disrupt any thermal if blowing above 12.87 m/s (in the Mojave Desert [4]), but also favors organized thermal convection if above 5 m/s [43].

Allen [4] describes the yearly and monthly statistics for $w^\star$ and z_i in Desert Rock, Nevada. Table 2.2 show the yearly statistics. The inference process described in chapter IV will take these table statistics as a prior belief.

The internal parameters are: the outer radius, and the vertical flow field, with:

$$w_T \sim w* \tag{2.73a}$$

$$r_2 \sim z_i \tag{2.73b}$$

The outer radius varies from thermal to thermal, and inside the thermal with the height above the ground. Lenschow and Stephens [36] state that the thermal radius

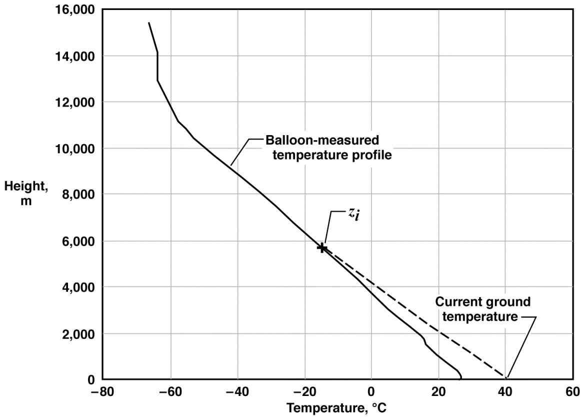


Figure 2.25: Example of z_i calculation (illustration from [4])

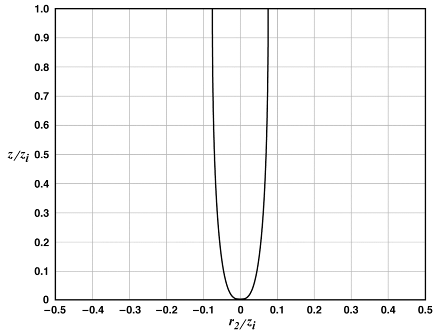


Figure 2.26: Updraft outer radius (r_2) function of z_i (illustration from [4])

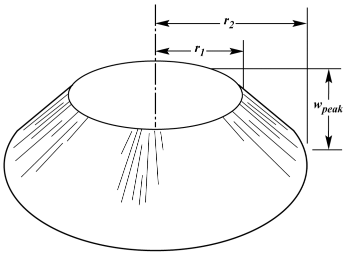


Figure 2.27: Updraft velocity trapezoidal model (illustration from [4])

average at a certain height is a direct function of the mixing-layer thickness (z_i), by (fig. 2.26):

$$r_2 = \max\left[10, 0.2513\left(\frac{z}{z_i}\right)^{\frac{1}{3}}\left(1 - 0.25\frac{z}{z_i}\right)\cdot z_i\right] \tag{2.74}$$

Note that multiplier 0.2513 is different from the one used to by Allen [4], as this author corrected the calculations later on.

The equation governing the vertical velocity is [4]:

$$w = w_{peak}\left(\frac{1}{1 + \left|k_1\frac{r}{r_2} + k_3\right|^{k_2}} + k_4\frac{r}{r_2} + w_D\right)\left(1 - \frac{w_e}{w_{peak}}\right) + w_e \tag{2.75}$$

The constants k_1, k_2, k_3, and k_4 are defined in table 2.3.

The variables w_{peak}, w_D, w_e, and r_1 are defined next. We first define the average updraft speed function (fig. 2.29):

$$\overline{w} = w^{\star}\left(\frac{z}{z_i}\right)^{\frac{1}{3}}\left(1 - 1.1\frac{z}{z_i}\right) \tag{2.76}$$

Now, the radius for which the updraft speed is almost constant (r_1) comes from

$$\frac{r_1}{r_2} = \begin{cases} 0.0011r_2 + 0.14 & r_2 < 600 \\ 0.8 & otherwise \end{cases} \tag{2.77}$$

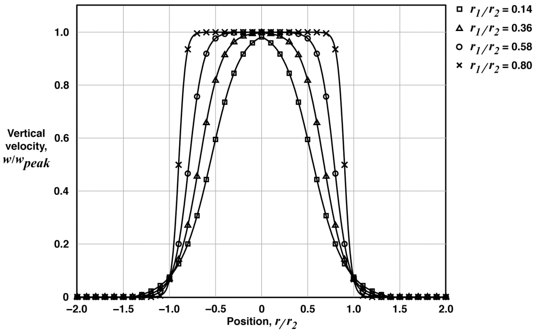


Figure 2.28: Updraft velocity bell-shape model (illustration from [4])

Table 2.3: Shape constants for bell-shaped vertical velocity distribution [4]

$\frac{r_1}{r_2}$	k_1	k_2	k_3	k_4
0.14	1.5352	2.5826	-0.0113	0.0008
0.25	1.5265	3.6054	-0.0176	0.0005
0.36	1.4866	4.8354	-0.0320	0.0001
0.47	1.2042	7.7904	0.0848	0.0001
0.58	0.8816	13.972	0.3404	0.0001
0.69	0.7067	23.994	0.5689	0.0002
0.80	0.6189	42.797	0.7157	0.0001

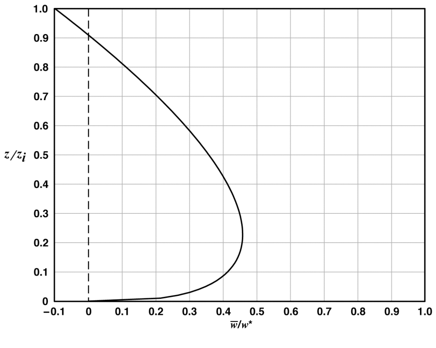


Figure 2.29: Average updraft velocity at different altitudes (illustration from [4])

The maximum updraft speed (w_{peak}) is defined as

$$w_{peak} = \overline{w}\frac{3r_2^2\left(r_2 - r_1\right)}{r_2^3 - r_1^3} \tag{2.78}$$

The skirt downdraft speed (w_D) is computed by

$$w_D = \begin{cases} \frac{\overline{w}5\pi}{12}\left(\frac{z}{z_i} - 0.5\right)\sin\left(\frac{\pi r}{r_2}\right) & \begin{array}{c} r \in (r_1, 2r_2) \\ \vee \\ \frac{z}{z_i} \in (0.5, 0.9) \end{array} \\ 0 & otherwise \end{cases} \tag{2.79}$$

To maintain a null regional net vertical velocity, we have to define the natural sink speed (w_e) as:

$$w_e = -\overline{w}\frac{N\pi r_2^2}{A_{reg} - N\pi r_2^2} \cdot \begin{cases} \left[1 - 2.5\left(\frac{z}{z_i} - 0.5\right)\right] & \frac{z}{z_i} \in (0.5, 0.9) \\ 1 & otherwise \end{cases}, \tag{2.80}$$

where A_{reg} is the affected region.

2.5.2.3 Chimney Thermal - Interaction with the Wind

There are three main effects the wind may have in *Chimney Thermals.* Wind seams to be fully disruptive for the thermals if blowing faster than 13 m/s. If it is slower, it may lean the thermal, move it, or both. The thermal drift depends both on the wind speed and on the terrain radiating properties. The drift velocity generally follows the prevailing wind, but not always [4]. If the terrain radiating properties are very uneven there is a tendency for the thermal to stay anchored to the most radiating points, usually called hot-spots, such as rocks, building roofs, asphalt, etc. When the thermal source, the lowest section of the thermal, is anchored to a hot spot or moves slower than the wind speed, the thermal has the tendency to lean to the leeward direction (fig. 2.30). During a thermal soaring flight thermal leaning is sometimes wrongly perceived as a drift by the whole thermal.

As such, the movement dynamics of a *Chimney Thermal* may be captured by:

$$\dot{x}_T = u_T = V_T \cos\psi_T \tag{2.81a}$$

$$\dot{y}_T = v_T = V_T \sin\psi_T \tag{2.81b}$$

$$V_T \sim \mathcal{N}\left(\mu_{V_T}, \sigma_{V_T}\right), \mu_{V_T} \in [0, \|\mathbf{w}\|] \tag{2.81c}$$

$$\psi_T \sim \mathcal{N}\left(\mu_{\psi_T}, \sigma_{\psi_T}\right), \tag{2.81d}$$

where x_T and y_T are the *Chimney Thermal* source position coordinates, u_T and v_T are the *Chimney Thermal* source drift velocities, μ_{V_T} and σ_{V_T} are the thermal drift speed mean and standard deviation and μ_{ψ_T} and σ_{ψ_T} are the thermal drift direction probability parameters.

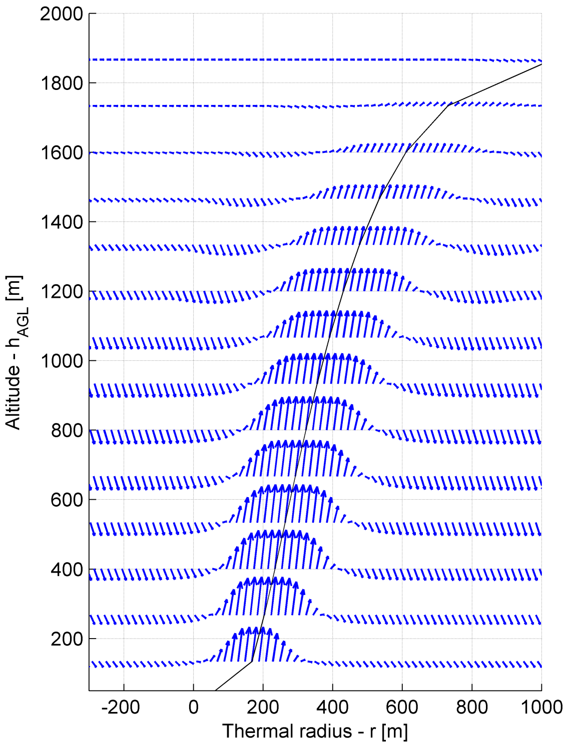


Figure 2.30: Leaning *Chimney Thermals*. Effect of wind on the shape of the thermal flow field.

The *Chimney Thermal* leaning may be characterized by the change in the updraft core center for different altitudes:

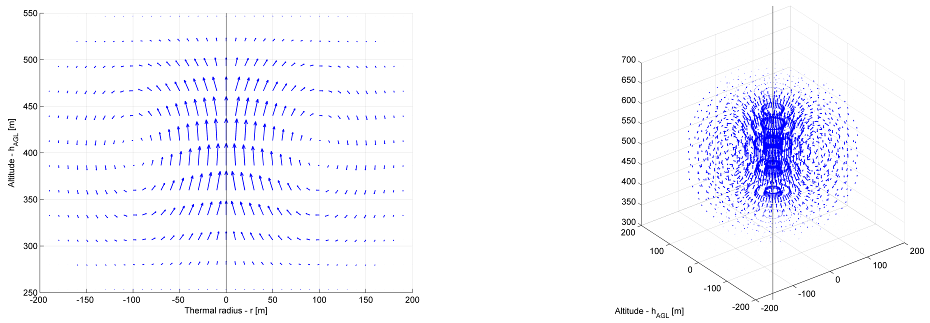
$$x_t\left(H\right) \approx x_T + \int\limits_0^H \frac{\overline{W}_x\left(h\right) - u_T}{\overline{W}_{T,z}\left(h\right)} dh \tag{2.82a}$$

$$y_t\left(h\right) \approx y_T + \int\limits_0^H \frac{\overline{W}_y\left(h\right) - v_T}{\overline{W}_{T,z}\left(h\right)} dh, \tag{2.82b}$$

where $x_t\left(H\right)$ represents the updraft core center coordinates at an altitude above the ground H, $\overline{W}_x\left(h\right)$ and $\overline{W}_y\left(h\right)$ are the mean wind velocities away from the thermal at each altitude h, and $\overline{W}_{T,z}\left(h\right) = \overline{w}$ is the updraft mean speed at each altitude h. From this equation it is clear that the thermal will not lean if the source is moving at the same velocity as the wind. As stated before, and because *Chimney Thermal* movements depend on the terrain, the thermal will present some leaning associated with some drift.

2.5.2.4 Bubble Thermal - Lawrance Model

Lawrance [3] developed a *Bubble Thermal* model. Unlike the *Chimney Thermal* model developed by Allen, this model describes an unsteady upward moving phenomenon. In this model the hotter air mass rises in a bubble like structure, disconnected from the ground or the inversion layer. The model described in [3] defines a toroidal 3D flow field at a given instant (Fig. 2.31). The flow field model created



(a) 2D flow field

(b) 3D flow field

Figure 2.31: Flow field in a Lawrance Bubble Thermal. The vortex is noticeable around the thermal radius (100m) at the mean altitude (400m).

by Lawrance is defined by:

$$w_x = -w_z \frac{z}{(d_H - R) \cdot k^2} \frac{x}{d_H} \tag{2.83a}$$

$$w_y = -w_z \frac{z}{(d_H - R) \cdot k^2} \frac{y}{d_H} \tag{2.83b}$$

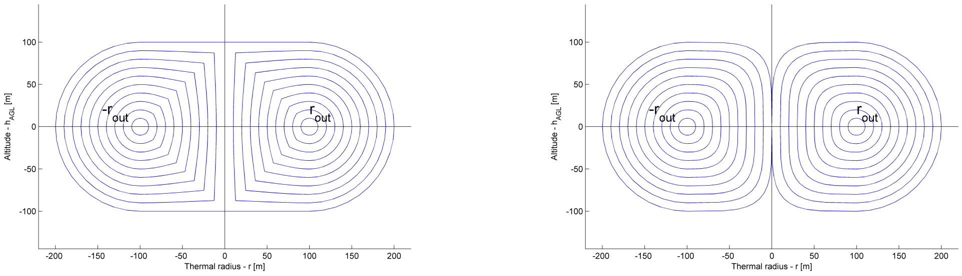
$$w_z = \begin{cases} w_{core} & x = 0 \\ \frac{\cos\left(1+\frac{\pi z}{k \cdot R}\right)}{2} \frac{R \cdot w_{core}}{\pi d_H} \sin\left(\frac{\pi d_H}{R}\right) & d_H \in (0, 2R] \\ 0 & otherwise \end{cases}, \tag{2.83c}$$

(2.83d)

where $d_H = \sqrt{x^2 + y^2}$, w_{core} is the bubble core updraft speed, R is the distance which limits the updraft area, i.e., the area around the bubble center where the flow moves upwards, x, y, and z are positions relative to the bubble center, and k is the bubble eccentricity factor, i.e., $k = \frac{\Delta z_{flow}}{2R}$, where Δz_{flow} is the bubble height.

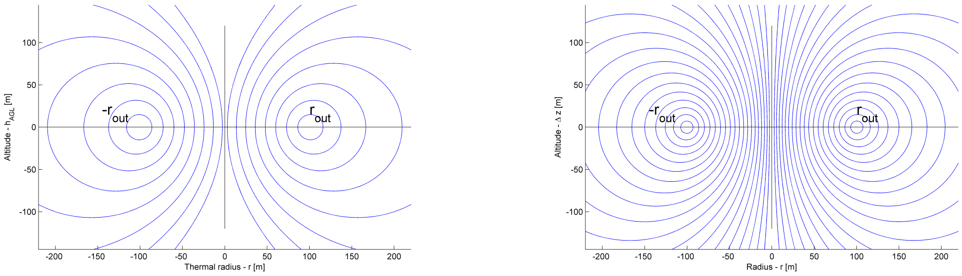
2.5.2.5 Bubble Thermal - Conservative Flow Model

The Lawrance *Bubble Thermal* model is not mass conservative, does not include any effects of the interaction with the prevailing wind, and does not present the possibility to represent an updraft core plateau described by some authors. Figure 2.32 illustrates some of the models we developed to try and overcome some of these issues. The bubble flow is only conservative if the whole bubble region can be described by streamlines, i.e., the lines ”followed” by air particles in the absence of disturbances. The first two models are adaptations of the toroid streamline model on which the Lawrance model is also based. The first model (fig. 2.32a) presents sharp flow direction changes which are not realistic at all, as they would require infinite flow acceleration at those points. The second model (fig. 2.32b) presents more realistic



(a) 1^{st} streamline hypothesis. Presents sharp flow direction changes.

(b) 2^{nd} streamline hypothesis. Lacks any flow near the top and the bottom of the bubble.



(c) 3^{rd} streamline hypothesis. Presents no flow near the bubble axis, when away from the central plane.

(d) Final streamline hypothesis.

Figure 2.32: Sequence of streamline models for a conservative flow *Bubble Thermal* model.

streamlines. The two main handicaps are the complex computation of some of the parameters, when taking a relative position as an input, and the lack of flow near the top and the bottom of the bubble. In the next streamline models we drop the constraint for all the streamlines to be centered at the toroid center, i.e., at the updraft outer radius on the mean bubble altitude plane. Instead, we define the streamlines' center to tend to the toroid center as they approach it. The third model (fig. 2.32c) presents nice properties in terms of parameter computation, when taking a relative position as an input, but presents no flow near the bubble axis, when away from the central plane. The final model (fig. 2.32d) presents a flow tending to vertical near the bubble axis, a vortex around the updraft outer radius, and is conservative in terms of mass exchange. The computation of the streamline parameters requires an iterative process when the input to define those parameters is a relative position to the bubble center. This will be shown below in equation (2.84).

The final model presents a flow tending to vertical near the bubble axis, as we define that the streamlines center should tend to infinite as those approach the bubble axis. We also define that the streamlines direction should be perpendicular to a circle at a distance R_T from the bubble center (fig. 2.33), where R_T is the *Bubble Thermal*

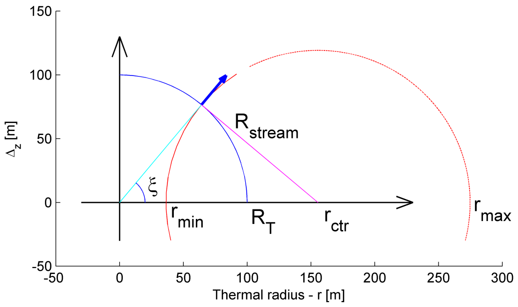


Figure 2.33: Streamline parameters definition.

updraft outer radius. The streamline parameters are then defined as:

$$\begin{aligned}
r_{max} &= \frac{R_T^2}{r_{min}} && \text{(2.84a)}\\
r_{ctr} &= \frac{r_{max} + r_{min}}{2} = \frac{R_T}{\cos\xi} && \text{(2.84b)}\\
R_{stream} &= \frac{r_{max} - r_{min}}{2} = R_T \tan\xi && \text{(2.84c)}\\
d_H &= r_{max} - \Delta z \tan\zeta && \text{(2.84d)}\\
\Delta z &= (d_H - r_{min}) \tan\zeta && \text{(2.84e)}\\
\frac{R_T^2}{r_{min}} + r_{min} &= \frac{\Delta z^2 + d_H^2 + R_T^2}{d_H}, && \text{(2.84f)}
\end{aligned}$$

where r_{min} and r_{max} are the minimum and maximum distances of the streamline to the *Bubble Thermal* axis, r_{ctr} is the streamline center distance to the *Bubble Thermal* axis, R_{stream} is the streamline radius, $d_H = \sqrt{\Delta x^2 + \Delta y^2}$ is the horizontal distance of a point to the *Bubble Thermal* axis, Δx, Δy, and Δz are the coordinates of the relative position to the *Bubble Thermal* center, ξ is the "exit" angle of the flow at a distance R_T from the bubble center, and ζ is the angle of a point relative to the streamline center (fig. 2.34). The parameter iteration need arises from r_{min} being defined by the implicit function (2.84f).

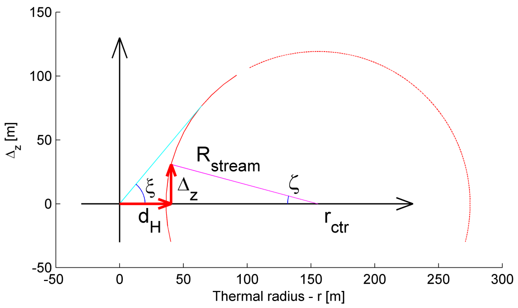


Figure 2.34: Random point definition on a streamline.

We may also define the streamlines such that the flow is perpendicular to an ellipse, which crosses the bubble horizontal plane at the updraft outer radius R_T. The streamline parameter may then be computed by solving:

$$r_{max} = \frac{R_T^2 \left(1 - e^2 \sin^2 \xi\right)}{r_{min}}. \tag{2.85}$$

To define the flow speed over each streamline, such that it presents a mass conservative flow, the volume rate has to be constant through each streamline:

$$\dot{V} = c^{st} = vr\pi dA \Leftrightarrow v_1 r_1 = v_2 r_2, \tag{2.86}$$

where $\dot{V}$ is the volume rate, v is the flow total speed, and dA is an infinitesimal area on the $r - z$ plane centered at radius $r_{(.)}$. This yields a flow speed over a streamline defined by:

$$W_2 = W_1 \frac{d_{H,2}}{d_{H,1}}, \tag{2.87}$$

where $W_{(.)}$ are total flow speeds at points 1 and 2 with horizontal distances $d_{H,(.)}$ from the bubble central axis. Combining (2.87) with the streamline orientation at each point from (2.84), the flow velocity vector at each point may be defined as:

$$\mathbf{w} = W_{z,r_{min},\Delta z=0} \frac{r_{min}}{d_H} \begin{bmatrix} \sin\zeta \frac{\Delta x}{d_h} \\ \sin\zeta \frac{\Delta y}{d_h} \\ \cos\zeta \end{bmatrix}. \tag{2.88}$$

To solve (2.88) and define completely the *Bubble Thermal* flow field one just needs to define an updraft field at the bubble mean altitude ($W_{z,r_{min},\Delta z=0}$). We considered three different hypotheses. The first is based on the flow field defined by Lawrance [3], constrained to $d_H \in [0, R_T]$ and $\Delta z = 0$:

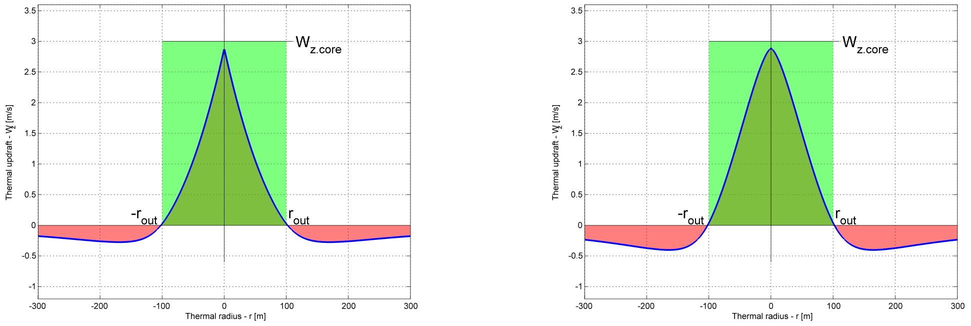
$$W_{z,d_H,\Delta z=0} = \begin{cases} w_{core} & d_H = 0 \\ \frac{w_{core}}{\pi} \left(\frac{R_T}{d_H}\right)^{\delta_{Flat}} \sin\left(\pi \left(\frac{R_T}{d_H}\right)^{\delta_{Flat}}\right) & d_H \in (0, R_T] \end{cases}, \tag{2.89}$$

where w_{core} is a positive value defining the maximum updraft speed and δ_{Flat} regulates the flatness of the updraft core and the abruptness of the updraft reduction towards to outer radius, as shown on figure 2.35.

The second model is based on the Gedeon updraft field [9], also constrained to $d_H \in [0, R_T]$ and $\Delta z = 0$:

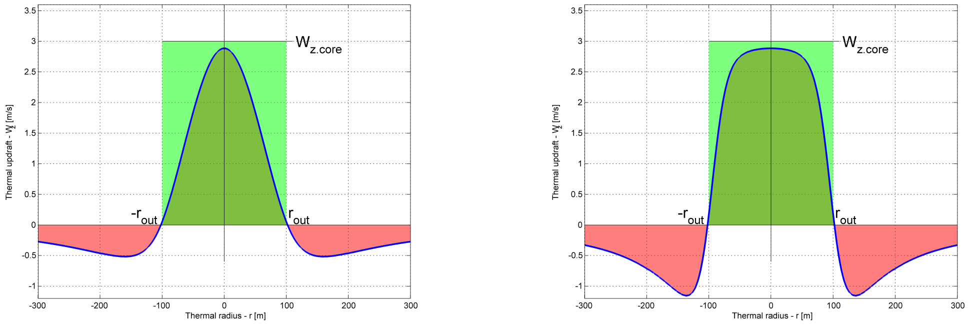
$$W_{z,d_H,\Delta z=0} = w_{core} e^{-(d_H/R_T)^{2\delta_{Flat}}} \cdot \left[1 - (d_H/R_T)^{2\delta_{Flat}}\right], \tag{2.90}$$

where w_{core} and δ_{Flat} are the same as in the first model.



(a) Vertical flow field with $\delta_{Flat} = 0.5$

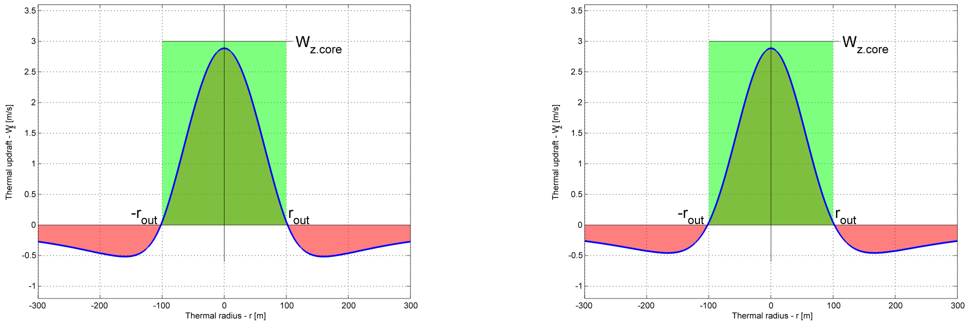
(b) Vertical flow field with $\delta_{Flat} = 0.75$



(c) Vertical flow field with $\delta_{Flat} = 1$

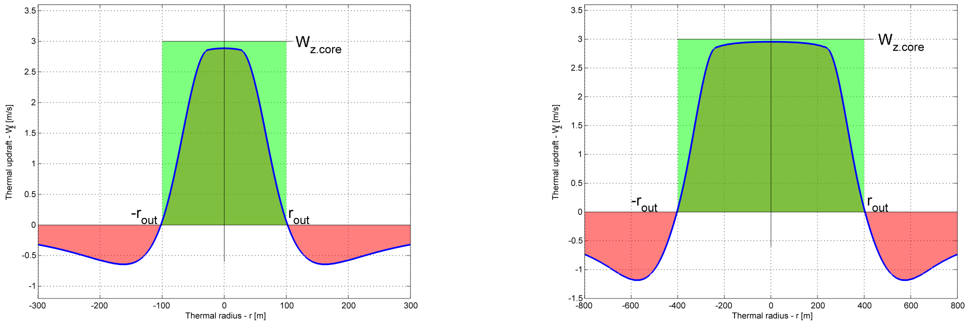
(d) Vertical flow field with $\delta_{Flat} = 0.4$

Figure 2.35: Vertical flow field based on Lawrance updraft field, with an updraft outer radius of 100m.



(a) Lawrance vertical flow field model

(b) Gedeon vertical flow field model



(c) Plateau Updraft vertical flow field model with $R_T = 100$

(d) Plateau Updraft vertical flow field model with $R_T = 400$

Figure 2.36: Comparison between vertical flow field models. Lawrance and Gedeon updraft models yield very similar results. The Plateau Updraft model presents an extra modeling degree-of-freedom.

The last model, *Plateau* updraft, is an extension of the second, with a core updraft plateau:

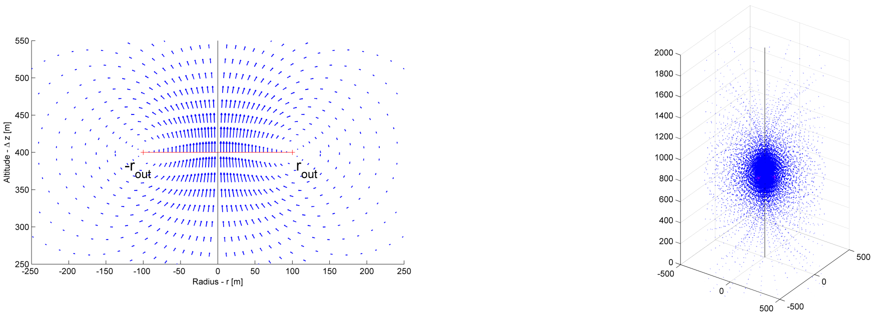
$$W_{z,d_H,\Delta z=0} = \begin{cases} w_{core} & d_H \leq R_{Plat} \\ w_{core} e^{-\delta_R} \cdot [1 - \delta_R] & d_H \in (R_{Plat}, R_T] \end{cases} \tag{2.91a}$$

$$with\ \delta_R = \left(\frac{d_H - R_{Plat}}{R_T - R_{Plat}} \right)^{2\delta_{Flat}} \tag{2.91b}$$

where R_{Plat} is the core updraft plateau radius. This may be defined as in the *Chimney model* by equation (2.77):

$$\frac{R_{Plat}}{R_T} = \begin{cases} 0.0011 R_T + 0.14 & R_T < 600 \\ 0.8 & otherwise \end{cases} \tag{2.92}$$

Figure 2.36 illustrates all three models showing that there is almost no difference between the Lawrance and the Gedeon models. It also shows that the plateau size control on the last model gives the user an extra degree-of-freedom, possibly allowing a better match between the model and real observations.



(a) 2D flow field

(b) 3D flow field

Figure 2.37: Flow field in a *Bubble Thermal* conservative model. The vortex is noticeable around the thermal updraft outer radius (100m) at the mean altitude (400m).

The resulting *Bubble Thermal* flow field is depicted in figure 2.37. It is very clear that the flow closest to the bubble axis is almost vertical, while near the updraft outer radius it turns into a vortex. The continuous nature of this model is also apparent. Figure 2.38 illustrates the main components of the *Bubble Thermal*, including the core updraft and the surrounding vortex, as proposed by Cone [45].

As *Bubble Thermals* are usually completely detached from the ground they are more affected by the prevailing wind than the *Chimney Thermals*. The *Bubble Thermals* center velocity is defined as:

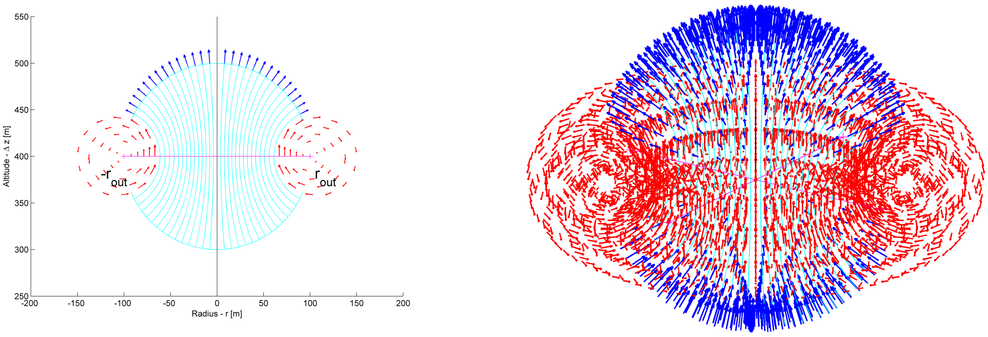
$$\dot{x}_T = u_T = V_T \cos \psi_T \tag{2.93a}$$

$$\dot{y}_T = v_T = V_T \sin \psi_T \tag{2.93b}$$

$$\dot{z}_T = w_T \tag{2.93c}$$

$$V_T \sim \mathcal{N}\left(\mu_{V_T}, \sigma_{V_T}\right), \mu_{V_T} = \|\mathbf{w}\| \tag{2.93d}$$

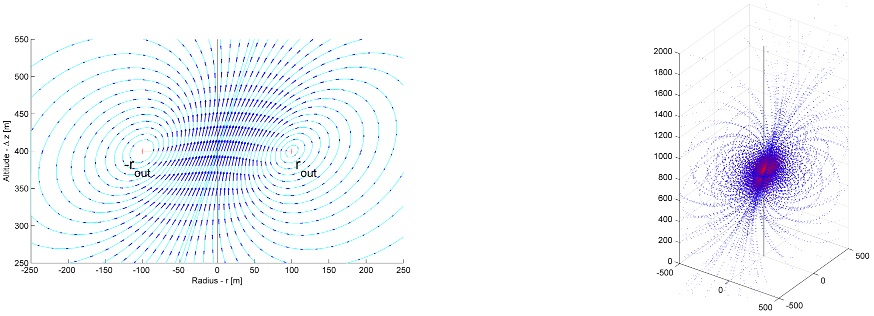
$$\psi_T \sim \mathcal{N}\left(\mu_{\psi_T}, \sigma_{\psi_T}\right), \tag{2.93e}$$



(a) 2D diagram (b) 3D diagram

Figure 2.38: *Bubble Thermal* conservative model structure diagram. The vortex is illustrated by red arrows around the updraft outer radius. The blue arrows illustrate the updraft flow which birds and gliders usually take advantage of.

where x_T, y_T, and z_T are the *Bubble Thermal* center coordinates, u_T, v_T, and w_T are the *Bubble Thermal* center drift velocities, μ_{V_T} and σ_{V_T} are the thermal drift speed mean and standard deviation and μ_{ψ_T} and σ_{ψ_T} are the thermal drift direction probability parameters.



(a) 2D flow field (b) 3D flow field

Figure 2.39: Flow field in a leaning *Bubble Thermal*. It is clear that the flow crosses the bubble center at a certain angle, which depends both on the bubble and the wind velocity. In the 3D picture the strongest flow is depicted in red.

If the *Bubble Thermal* moves with a difference velocity from the prevailing wing its structure leans. The combined flow field from a *Bubble Thermal* and the prevailing wind is defined by:

$$\mathbf{w} = W_{z,r_{min},\Delta z=0}\frac{r_{min}}{d'_H}\begin{bmatrix}\sin\zeta'\frac{\Delta x'}{d'_H}+l_x\cos\zeta'\\ \sin\zeta'\frac{\Delta y'}{d'_H}+l_y\cos\zeta'\\ \cos\zeta'\end{bmatrix}+\begin{bmatrix}\overline{W}_x\\ \overline{W}_y\\ 0\end{bmatrix}. \tag{2.94}$$

where

$$l_x = \frac{u_T - \overline{W}_x}{w_T - w_{core}} \tag{2.95a}$$

$$l_x = \frac{v_T - \overline{W}_y}{w_T - w_{core}}, \tag{2.95b}$$

and the streamlines are defined by:

$$\Delta x' = \Delta x - l_x \Delta z \tag{2.96a}$$

$$\Delta y' = \Delta y - l_y \Delta z \tag{2.96b}$$

$$d'_H = \sqrt{\Delta x'^2 + \Delta y'^2} \tag{2.96c}$$

$$d'_H = r_{max} - \Delta z \tan \zeta' \tag{2.96d}$$

$$\Delta z = (d'_H - r_{min}) \tan \zeta' \tag{2.96e}$$

$$\frac{R_T^2}{r_{min}} + r_{min} = \frac{\Delta z^2 + {d'_H}^2 + R_T^2}{d'_H}, \tag{2.96f}$$

2.5.2.6 Thermal Models Comparison

We now summarize the main similarities and differences among the presented thermal models. The models described above have some important similarities - the updraft speed is maximized at the thermal core and updraft field presents a *bell-like* shape. The main differences among the models are the manner in which the flow field changes with altitude, the main parameters governing the *bell-like* shape of the updraft speed, the dependencies on the thermal outer radius, the thermal core movement, and how the thermal flow field leans with the wind. Table 2.4 summarizes these differences among models, which are explained in more detail next.

Table 2.4: Thermal models comparison

Model	*Dim*	*Downdraft*	*Flat core*	*Mix. layer*^{1*}	*Type*
Gaussian [27]	2D	No	No	Independent	No type
Gedeon [9]	2D	Constrained^{3*}	No	Independent	No type
Allen [4]	3D	Extended^{4*}	Yes	Dependent	Chimney
Bencatel Mov. Chimn.^{6*}	3D	Extended^{4*}	Yes	Dependent	Chimney
Lawrance [3]	3D	Constrained^{3*}	No	Independent	Bubble
Bencatel Cons. Bubb.^{7*}	3D	Extended^{4*}	Choice^{5*}	Independent	Bubble

Model	*Flow Field*^{2*}	*Conservative*	*Leaning*	*Movement*
Gaussian [27]	Vertical	No	No	No
Gedeon [9]	Vertical	No	No	No
Allen [4]	Vertical	Yes	No	No
Bencatel Mov. Chimn.^{6*}	Vertical	Yes	Yes	Yes
Lawrance [3]	3D	No	No	No
Bencatel Cons. Bubb.^{7*}	3D	Yes	Yes	Yes

1⋆ *Mixing layer parameters - Thermal flow field dependence on the Mixing layer parameters.*

2⋆ *Flow Field representation: Only vertical component or full 3D flow field representation.*

3⋆ *Downdraft constrained to the thermal rim.*

4⋆ *Downdraft extended to the thermal surrounding area, beyond the thermal rim.*

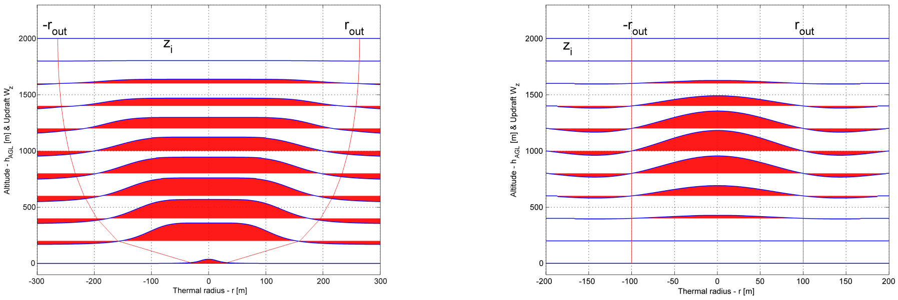
5⋆ *The user can choose if there is a flat plateau and/or the flatness/abruptness of the whole bell-shape updraft.*

6⋆ *Bencatel Moving Chimney*

7⋆ *Bencatel Conservative Bubble*

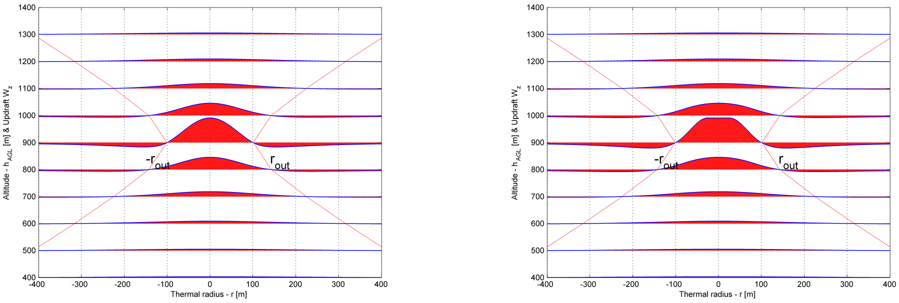
Altitude handling - Both *Chimney Thermal* and *Bubble Thermal* models incorporate variation with altitude. By contrast, the Gaussian and the Gedeon models are invariant with altitude, as shown in figure 2.40. The Allen model simulates a *Chimney* type thermal, where the thermal extends from the ground to the top of the mixing-layer (fig. 2.40a). In figure 2.40a it is possible to see the updraft speed and outer radius dependence over the altitude range. In the *Bubble Thermal* models the updraft region is limited to the mixing-layer, but may not reach either the ground or the top of the mixing layer (fig. 2.40b). In the Allen *Chimney Thermal* model, the updraft speed depends on the altitude. By contrast, the Lawrance *Bubble Thermal* model defines a fixed outer radius. The conservative *Bubble Thermal* models present a variable outer updraft radius, but this variation is distinctly different from the one defined by the Allen model.

***Bell-like* shape** - All models present a *Bell-like* shape with some differences. Then Allen *Chimney Thermal* model (fig. 2.41a) presents an updraft core with skirt downdrafts. In volumetric terms, the updraft and downdraft cancel each other, presenting a conservative airmass flow. Further, it shows a flat core for large diameter thermals. This shape is based on real observations as described by Irving [49]. The maximum updraft is not constant, decreasing as the outer diameter increases. The Lawrance *Bubble Thermal* model (fig. 2.41b) also presents a central updraft and exterior downdrafts. Unlike the downdrafts from the other models, in this model the negative vertical flow is constrained to a distance twice the outer radius. Moreover, the flow is only conservative at the bubble mean altitude, no flat core is modeled, and the maximum updraft only varies with altitude and not with the outer diameter. The conservative *Bubble Thermal* models present a mass conservative flow, with the downdraft extending to infinite, but decreasing with the distance to the bubble center. The core updraft also changes with altitude, although it covers different radii for both models. The conservative *Bubble Thermal* model with a Lawrance updraft field (fig. 2.41c) shows no flat core, while the one with the *Plateau* updraft (fig. 2.41d) defines a near flat core. The Gaussian-shape and Gedeon models are simplifications which present the core updraft. The only difference between these two models is the presence of skirt downdraft on the Gedeon model. These two models are not flow conservative and their maximum updraft is always constant, with respect to the altitude and outer diameter.



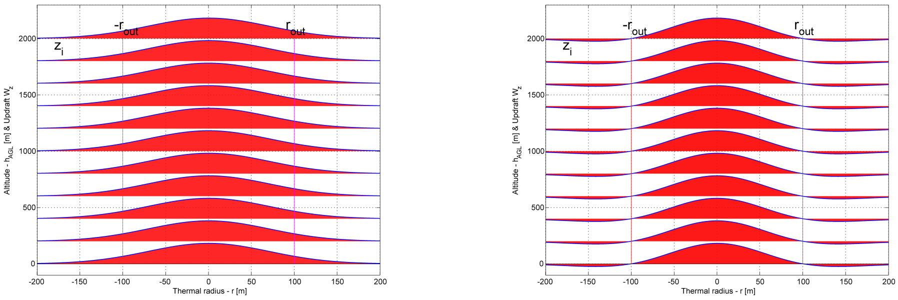
(a) Allen *Chimney Thermal* model - The outer diameter varies with the altitude as well as with the updraft speed.

(b) Lawrence *Bubble Thermal* model - The updraft speed varies with the altitude, but not with the outer diameter.



(c) Conservative *Bubble Thermal* model, with Lawrance updraft field - The outer diameter varies with the altitude as well as with the updraft speed.

(d) Conservative *Bubble Thermal* model, with *Plateau* updraft field - The outer diameter varies with the altitude as well as with the updraft speed.

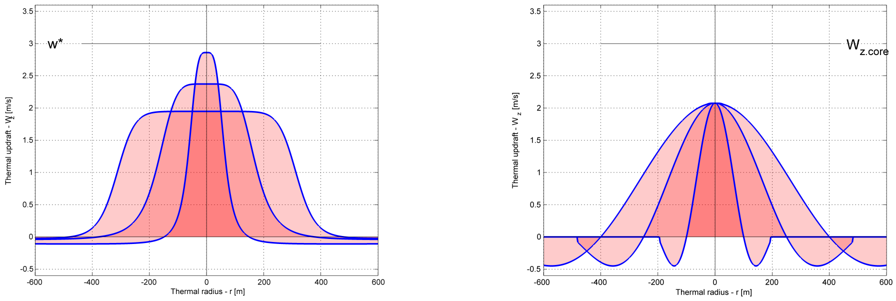


(e) Gaussian Thermal model - The altitude doesn't affect the outer radius or the updraft speed.

(f) Gedeon Thermal model - The altitude doesn't affect the outer radius or the updraft speed.

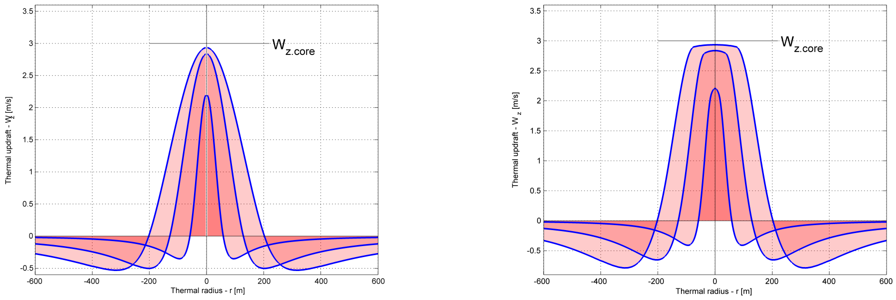
Figure 2.40: Differences on altitude dependence of the different models.

Outer radius - For most of the models the outer radius represents the distance to the core where the vertical flow is inverted, turning from updraft to downdraft. That is the case for the *Bubble Thermal* models and the Gedeon model. Further, in the *Bubble Thermal* models the outer radius at the *Bubble* mean altitude also indicates the center for the toroid vortex. Both the Allen and the Conservative *Bubble Thermal* models have the mean updraft strength as a function of the outer radius, and outer radius a function of the altitude. In the Allen model the flow field at the outer radius is almost null, but not quite. In the Gaussian model the outer radius is just a measure of the radial spread of the updraft.



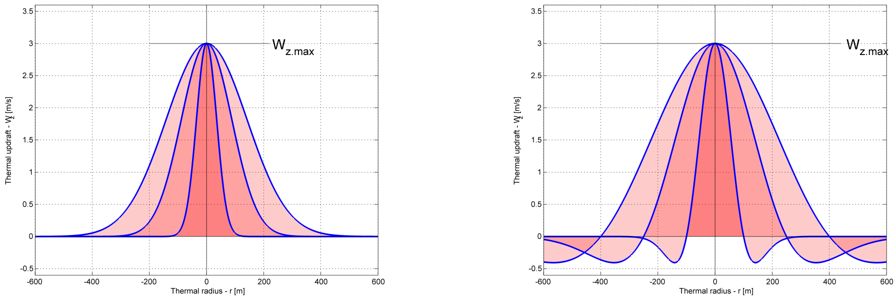
(a) Allen *Chimney Thermal* model - Presents a constant updraft speed at the core for large diameter thermals. Further, the updraft speed is inversely correlated with the thermal diameter.

(b) Lawrance *Bubble Thermal* model - Presents a maximum updraft speed constant with the altitude and no flat core for large diameter thermals.



(c) Conservative *Bubble Thermal* model, with Lawrance updraft field - Presents a correlation between the updraft speed and the thermal diameter.

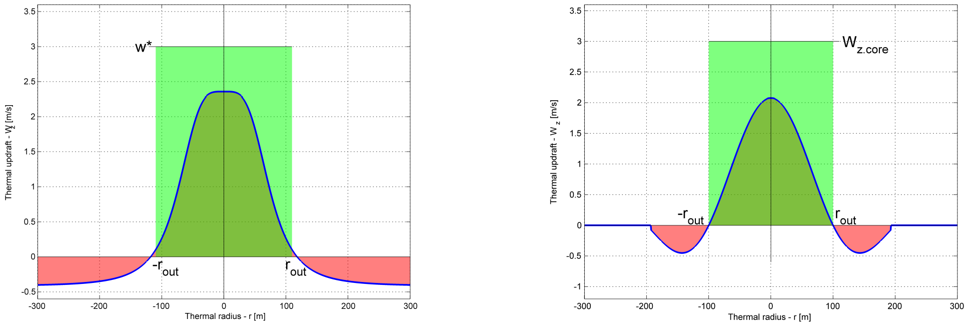
(d) Conservative *Bubble Thermal* model, with *Plateau* updraft field - Presents an almost constant updraft speed at the core for large diameter thermals and a correlation between the updraft speed and the thermal diameter.



(e) Gaussian Thermal model - Presents a constant maximum updraft speed and no flat core for large diameter thermals.

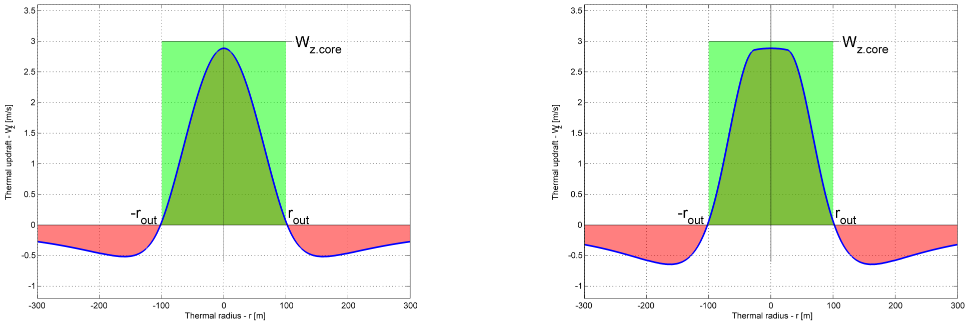
(f) Gedeon Thermal model - Presents a constant maximum updraft speed and no flat core for large diameter thermals.

Figure 2.41: Updraft function shape among the different models.



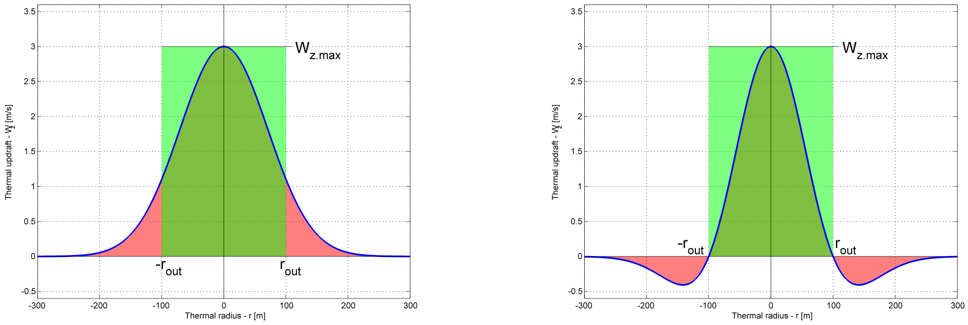
(a) Allen *Chimney Thermal* model.

(b) Lawrance *Bubble Thermal* model.



(c) Conservative *Bubble Thermal* model, with Lawrance updraft field.

(d) Conservatibe *Bubble Thermal* model, with *Plateau* updraft field.

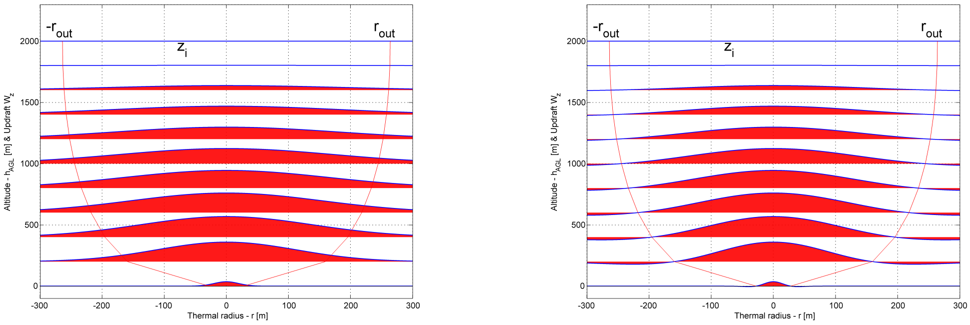


(e) Gaussian Thermal model.

(f) Gedeon Thermal model

Figure 2.42: Differences in outer radius dependence of the different models. The outer radius presents a null updraft speed for most of the models. The exceptions are the Allen *Chimney Thermal* model, where the updraft is almost null at the outer radius, and the Gaussian model, which always presents a positive updraft speed, about 37% of the maximum updraft speed at the outer radius.

We may extend the Gaussian and the Gedeon models into more realistic *Chimney* models, if we use the Allen model's outer radius function (2.74) and core updraft function (2.78). This would make the outer radius a function of altitude and the updraft speed a function of both altitude and outer radius (fig. 2.43). Even so, these models would remain non conservative in terms of air mass exchange.



(a) Allen-Gaussian Thermal model

(b) Allen-Gedeon Thermal model

Figure 2.43: Adaptation of the Gaussian shape and Gedeon models to simulate a *chimney* type thermal - The altitude affects the outer radius and the maximum updraft speed in the same manner as in the Allen model.

2.5.2.7 Thermals Development and Fading

Individual thermals' appearance rate may be modeled by a Poisson distribution. Based on Lenschow and Stephens data collected over the ocean [36], the average number of thermals encountered over a path of length s, normalized by the Mixed-Layer altitude z^i, is

$$\overline{N_T} \approx \frac{1.2s}{z^i}, z \in \left[0.1z^i, z^i\right], \tag{2.97}$$

where z is the measurement altitude. As such, the resulting Poisson distribution is

$$P_{Thermal}\left(s\right) = 1 - e^{-\lambda_T \cdot s}, \lambda_T \approx \frac{1.2}{z^i}. \tag{2.98}$$

Thermals present different strengths and sizes, even if they are developed in the same region with the same environmental conditions. The size probability distribution of an ensemble of thermals follows a Gamma distribution [5, 50]. Vulf'son and Borodin derived analytically the distribution of spontaneous convective jets and compared the results with several sets of empirical data. First they derived an invariant relation between the thermal diameter and its average vertical velocity, at each level h:

$$\hat{w} = \frac{3g\theta_T}{16\alpha_R}D. \tag{2.99}$$

where g is the gravity acceleration, θ_T is the environment temperature excess and $\alpha_R = 0.1$. The size and strength probability density function (pdf) was then defined as:

$$\frac{N_w}{N_0} = \frac{a}{\Gamma\left(a\right)\overline{\hat{w}^2}}\left[\frac{a\hat{w}^2}{\overline{\hat{w}^2}}\right]^{a-1} e^{\frac{-a\hat{w}^2}{\overline{\hat{w}^2}}} \tag{2.100a}$$

$$\frac{N_D}{N_0} = \frac{a}{\Gamma\left(a\right)\overline{D}}\left[\frac{aD}{\overline{D}}\right]^{a-1} e^{\frac{-aD}{\overline{D}}}, \tag{2.100b}$$

$$\tag{2.100c}$$

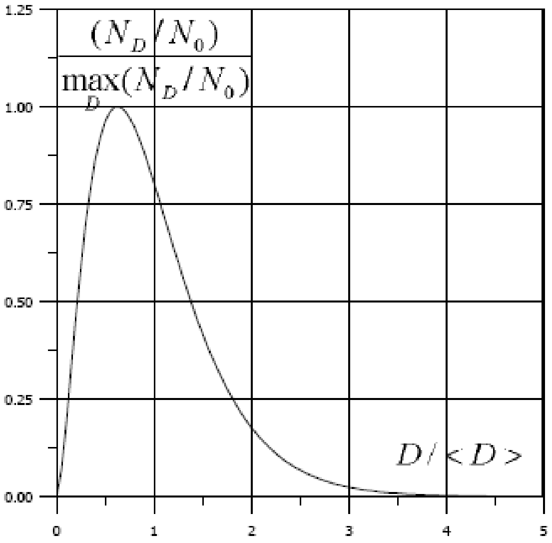


Figure 2.44: Diameter pdf with $a-1=5/3$ (illustrations from [5])

where $\overline{\hat{w}^2}$ and $\overline{D}$ are the average strength and diameter among all thermals, N_w/N_0 and N_D/N_0 are the fraction of thermals with strength or diameter in the interval $\left[\hat{w}^2, (\hat{w}+d\hat{w})^2\right]$ or $[D, D+dD]$, N_0 is the total number of thermals, and a is the pdf shape parameter (fig. 2.44). The pdf differs greatly from the surface layer to the rest of the Mixed-Layer. The surface layer extends from the ground to 100-300 meters. At these altitudes the size distribution is characterized by $a-1=1.67$, while in the above layer the distribution shape is characterized by $a-1=2.13$ [5, 50].

After the development of a thermal, it may fade after some time or merge with nearby thermals, forming a bigger thermal that will itself fade away after some time. Thermal lifespan can range from 5 to 30 minutes [46, 4, 28], with 20 minutes as the mean lifespan.

2.5.2.8 Orographic Updrafts

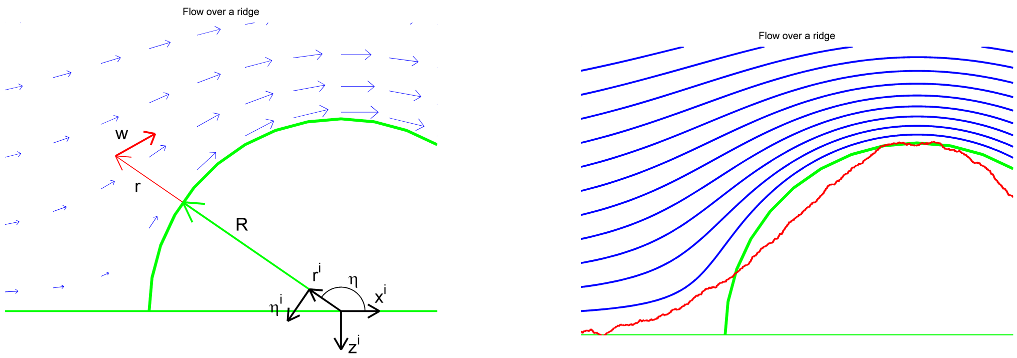
Ridge lift - Ridge or slope lift appears along the windward side of mountain ridges. It is generated when the flowing air mass collides with the mountain side and is forced to climb to overcome the obstacle. Langelaan described a general model for the ridge flow field based on a semi-cylindric obstacle over a flat surface [51] (fig. 2.45):

$$\mathbf{w} = w_\infty \hat{\mathbf{x}} - w_\infty \frac{R^2}{r^2}\left(\cos\eta \hat{\mathbf{r}} + \sin\eta \hat{\eta}\right) \tag{2.101a}$$

$$w_x = w_\infty \left[1 - \frac{R^2}{r^2}\left(\cos^2\eta - \sin^2\eta\right)\right] \tag{2.101b}$$

$$w_z = 2w_\infty \frac{R^2}{r^2}\cos\eta \sin\eta, \tag{2.101c}$$

The updraft is generated if the general wind direction is within 30° to 40° to the perpendicular to the ridge line [43]. Unlike thermals, the ridge updraft maintains a stable position, endures as long as the wind conditions are favorable, and can cover large areas, allowing long upward soaring legs. The main limitation of the ridge lift is that it is constrained to a limited altitude above the ridge crest.



(a) Ridge flow approximation by a semi-cylindrical obstacle. In the picture x^i, z^i, r^i, and η^i correspond to $\hat{\mathbf{x}}$, $\hat{\mathbf{z}}$, $\hat{\mathbf{r}}$, and $\hat{\eta}$, respectively.

(b) Ridge flow stream-lines.

Figure 2.45: Ridge updraft flow field model.

Wave lift - Wave lift is another effect of mountainous terrain. It develops after the air mass has passed over the undulating terrain at high altitudes. ”This lift is part of a large scale deflection of air mass, which is known as “lee wave” lift, first recognized in the 1930s and explored scientifically in the early 1950s” [52]. The lee wave flow field structure is similar to a sinusoidal wave (fig. 2.45b). ”Lee wave field structures the air mass sink rates in parallel bands having high cross-stream coherence” [53]. The wave length λ_W of this phenomenon depends on the atmospheric conditions, in particular the atmosphere stability. ”In the atmosphere, [...] $\lambda_W \sim \mathcal{O}\left(10n.mi.\right)$ for hydrostatic lee waves or $\lambda_W \sim \mathcal{O}\left(1n.mi.\right)$ for nonhydrostatic lee waves” [53]. Wave lift development depends greatly on the presence of relatively high winds and stable atmosphere conditions.

2.5.2.9 Experimental Verification of Flow Field Models

An important issue in models is their verification as representative of the reality. In the case of thermals and orographic updrafts one of the most important features is their internal flow field structure. We have not found any study referring to airflow data collected inside thermals or orographic updrafts that could be useful for the validation of these features. The studies we found referring to real data from thermals are from Lenschow and Stephens [36] and from Allen [4].

Allen [4] studied the distribution of two environmental parameters that influence the thermal size and strength, in Desert Rock, Nevada. These parameters are the maximum altitude a thermal may reach, i.e., the atmosphere mixing-layer altitude, and the convection intensity in the mixing-layer (z_i), represented by the convective velocity scale ($w^{\star}$), which is related to the thermals' updraft speed.

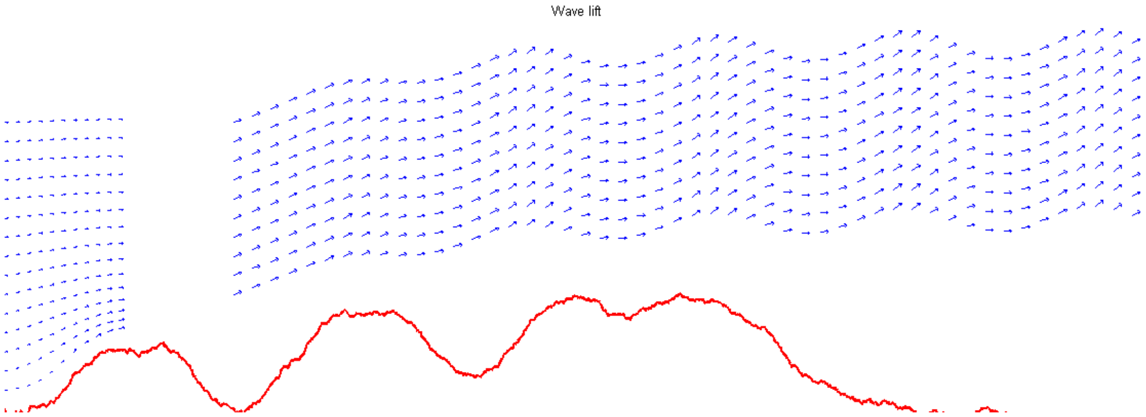


Figure 2.46: Wave flow.

Lenschow and Stephens's work [36] provides the best data to relate the altitude with the average thermal horizontal size and the updraft speed. The data was collected from a thermal field over the ocean, flying long lines and averaging the airflow data among all detected thermals at each altitude. This method is good to obtain the general relations between altitude and thermal size and strength. However, it is not good enough to evaluate the evolution of the thermal diameter within individual thermals. That would be important to evaluate the 3D shape of the thermals, identifying them as *Chimney* or *Bubble Thermals* and allowing the validation of the models presented in section 2.5.2.1. Furthermore, the averaging of the thermal airflow observations makes it impossible to validate the internal thermal flow field predicted by any thermal model.

2.5.3 Gust Models

Several models are used to simulate gust turbulence. The simplest model is a Gaussian noise model, where the mean wind is complemented by deviations generated from a Gaussian distribution:

$$W_x = \overline{W}_x + \Delta W \cos\psi_G \tag{2.102a}$$

$$W_y = \overline{W}_y + \Delta W \sin\psi_G \tag{2.102b}$$

$$\Delta W \sim \mathcal{N}\left(0, k_G \left\|\overline{w}\right\|\right) \tag{2.102c}$$

$$\psi_G \sim \mathcal{U}\left(0, \pi\right), \tag{2.102d}$$

where $\overline{W}_x$ and $\overline{W}_y$ are the mean wind velocities. ΔW is the total velocity variation and ψ_G is the velocity variation direction; they are generated by the Normal distribution $\mathcal{N}\left(0, k_G \left\|\overline{w}\right\|\right)$, with standard deviation proportional to the total wind speed $\sigma_G = k_G \left\|\overline{w}\right\|$, and the Uniform distribution $\mathcal{U}\left(0, \pi\right)$.

A more realistic model is the Gauss-Markov process random noise model. In this model each generated wind deviation doesn't only affect the wind at the generation moment, but its effect is also extended to the next moments by a decaying function. This yields a smoother and also more realistic wind variation. The deviations are still generated from a Gaussian distribution, but those are added to a decaying function from past deviations:

$$W_x = \overline{W}_x + \Delta W_x \tag{2.103a}$$

$$W_y = \overline{W}_y + \Delta W_y \tag{2.103b}$$

$$\Delta W_x = \Delta W_x' \left(1 - \tau_G \Delta t\right) + \Delta W^+ \cos \psi_{G+} \tag{2.103c}$$

$$\Delta W_y = \Delta W_y' \left(1 - \tau_G \Delta t\right) + \Delta W^+ \sin \psi_{G+} \tag{2.103d}$$

$$\Delta W^+ \sim \mathcal{N}\left(0, k_{G+} \left\|\overline{w}\right\|\right) \tag{2.103e}$$

$$\psi_{G+} \sim \mathcal{U}\left(0, \pi\right), \tag{2.103f}$$

where $\overline{W}_x$ and $\overline{W}_y$ are the mean wind velocities. ΔW_x and ΔW_y are the wind variation velocities. $\Delta W_x'$ and $\Delta W_y'$ are the past wind variation velocities. τ_G is the decay time constant. ΔW^+ is the total velocity extra variation and ψ_{G+} is the velocity extra variation direction; they are generated by the Normal distribution $\mathcal{N}\left(0, k_{G+} \left\|\overline{w}\right\|\right)$, with standard deviation proportional to the total wind speed $\sigma_{G+} = k_{G+} \left\|\overline{w}\right\|$, and the Uniform distribution $\mathcal{U}\left(0, \pi\right)$. The extra wind variation standard deviation (σ_{G+}) is computed by the combination of the decay time constant (τ_G) and the overall wind standard deviation (σ_G):

$$\sigma_{G+} = \sigma_G \sqrt{\frac{2}{\tau_G}} \tag{2.104}$$

The Military Specification MIL-F-8785C [44] defines three gust models. The simplest model is the "1-cosine" shape model. The more realistic and most accepted models for aircraft simulation are the *von Karman* and the *Dryden* gust fields.

The models presented are well suited for open area flow simulation. However they don't capture well complex flow environments, as an urban grid. Models with more detail simulate those kind of environments better. For that, Cybyk et al [54] implemented a real-time physics-based simulation tool to study UAV dynamics in a urban environment.

2.6 Conclusions

In this chapter we present models for aircraft dynamics and for atmospheric air flow phenomena dynamics and flow fields. We present aircraft dynamics models with increasing complexity and realism. The simpler models present lower computational complexity. As such, they are useful for path planning, preliminary studies of maneuver controllers, and to study simple interactions with the air flow phenomena. The more realistic models allow the development of better performing maneuver controllers and the testing of those controllers with more accurate performance results.

The presented atmospheric air flow models describe updraft phenomena, with emphasis on thermals, several wind shear phenomena, and gusts. We present several models described in the literature and several new models The new *Chimney Thermal* model improves the modeling of the thermal center dynamics and the interaction of the whole updraft column with the prevailing wind. The new *Bubble Thermal* model improves the realism of the vortex flow field. It also incorporates dynamics and interactions with wind, similar to those implemented in the *Chimney Thermal*.

The wind shear models describe the air flow velocity gradient generated by the interactions between atmospheric layers or between the prevailing wind and a water or ground surface. We present new models for both the *Layer Wind Shear* and the *Ridge Wind Shear*.

We also present several models for wind gusts. As with the aircraft models, the simpler gust models are suitable for simulations demanding less realism. The more complex gust models allow the simulation of more realistic wind conditions, yielding more accurate performance evaluations of flight maneuver controllers.

The models presented in this chapter will be used throughout the thesis. The next chapter will evaluate the necessary conditions to perform perpetual flight by combining the aircraft and the air flow phenomena models. The *Flow Field Estimation* chapter studies the estimation of the flow field phenomena taking into account the combination of the aircraft dynamics and the flow field phenomena models. The *Control* chapter makes use of several aircraft models in the synthesis and testing of the collision avoidance and formation flight controllers.

CHAPTER III

Perpetual Flight Conditions

3.1 Introduction

This chapter is central to this thesis. Our main goal is to show that aircraft may use the energy of the flow field to dramatically extend their flight endurance. For that we define the minimum atmospheric conditions and aircraft aerodynamic parameters that would allow perpetual flight. Further, we evaluate these conditions in the light of the reality.

We start by laying out the energy balance equations in section 3.2. These support the derivation of conditions for perpetual flight and the estimation methods developed in chapter IV. In section 3.3 we define perpetual flight, as well as several other terms which are important to the clarity of the rest of the chapter. In sections 3.4, 3.5, and 3.6 we derive the minimal conditions necessary for perpetual flight in thermal fields, wind shear, and gusts.

3.2 Energy Dynamics

While flying, airplanes may use several sources of energy which are used mainly to overcome its drag. The energy state can be analyzed either within the airmass or relative to a ground reference. As the forces and moments acting on the aircraft depend mainly on the airspeed, most of the analysis will consider the airmass-relative energy.

For the total energy (E) balance we take into account only the potential energy (E_P) and the kinetic energy (E_K). We do not regard the fuel tank or the battery in the aircraft total energy, unless we explicitly define it otherwise. In the subsequent equations we regard the possibility of a motor energy input (T) for the sake of the generality of the equations. This is regarded as an external energy input and is important for the *Estimation* chapter (IV). In the perpetual flight minimum conditions analysis, developed in the next sections of this chapter, this energy input is not regarded, as we use the energy conveyed by the airflow (sec. 3.2.4).

3.2.1 Airmass-relative Energy

For an airplane the total energy relative to the airmass is

$$E = E_P + E_K = -mgz + \frac{mV_a^2}{2} \tag{3.1}$$

The time derivative yields:

$$\dot{E} = -mg\dot{z} + mV_a\dot{V}_a = mg\left(V_a \sin\gamma_a - W_z\right) + mV_a\dot{V}_a \tag{3.2}$$

We may now substitute the term $\dot{V}_a$ by the equations developed in section 2.4.6 for the leveled wings and turning flight cases. A remark to note that both cases allow ascending or descending flight. In the case of wings leveled flight we substitute $\dot{V}_a$ from (2.27) in (3.2), leading to the cancellation of the altitude-airspeed energy transfer terms, resulting in:

$$\dot{E} = -mgW_z + V_a\left(T - D\right) - mV_a \begin{bmatrix} \cos\gamma_a \\ -\sin\gamma_a \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \dot{W}_x \\ \dot{W}_z \end{bmatrix} \tag{3.3}$$

In the case of turning flight we substitute $\dot{V}_a$ from (2.29) in (3.2), yielding:

$$\dot{E} = mgW_z + V_a \cos\phi\left(T - D\right) - mV_a \begin{bmatrix} \cos\gamma_a \\ -\sin\gamma_a \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \dot{W}_x \\ \dot{W}_z \end{bmatrix} + V_a L \dot{\gamma_a} \tan\gamma_a \left(\cos\phi - 1\right) \tag{3.4}$$

3.2.2 Goal-relative Energy

If the airplane aims to reach a specific target ($[x_T, y_T]$) in the horizontal plane, the energy evaluation equation should be derived using the velocity aligned with the target direction. The energy and the energy rate are:

$$E = -mgz + \frac{mV_T\left|V_T\right|}{2} \tag{3.5a}$$

$$\dot{E} = -mg\dot{z} + m\left|V_T\right|\dot{V}_T. \tag{3.5b}$$

with the target aligned velocity (V_T):

$$V_T = \begin{bmatrix} \cos\chi_T \\ \sin\chi_T \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \tag{3.6a}$$

$$\dot{V}_T = \begin{bmatrix} \cos\chi_T \\ \sin\chi_T \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} + \dot{\chi}_T \begin{bmatrix} -\sin\chi_T \\ \cos\chi_T \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix}, \tag{3.6b}$$

where the direction relative to the target (χ_T), its time derivative ($\dot{\chi}_T$), and the distance to the target (d_T) are defined by,

$$\cos\chi_T = \frac{\Delta x_T}{d_T} \tag{3.7a}$$

$$\sin\chi_T = \frac{\Delta y_T}{d_T} \tag{3.7b}$$

$$\dot{\chi}_T = \frac{\Delta x_T^2}{\Delta x_T^2 + \Delta y_T^2}\frac{\Delta \dot{y}_T}{\Delta \dot{x}_T} \tag{3.7c}$$

$$d_T = \sqrt{\Delta x_T^2 + \Delta y_T^2} \tag{3.7d}$$

$$\Delta x_T = x_T - x \tag{3.7e}$$

$$\Delta y_T = y_T - y. \tag{3.7f}$$

3.2.3 Airplane Nominal Power

For airplanes, the nominal power, which depends on the propulsion setting and the airplane attitude, is $P_n = \dot{E}_n = V_a\,(T - D)$. The motor contribution may be defined in terms of its power (P) or thrust (T). The energy dissipation is caused by the drag, which is usually represented by its coefficient (C_D), yielding:

$$\dot{E}_n = P - \frac{\rho V_a^3}{2} S \cdot C_D = T \cdot V_a - \frac{\rho V_a^3}{2} S \cdot C_D \tag{3.8}$$

3.2.4 Flow Conveyed Power

Assuming there is no other power source than the aircraft motor, the power conveyed by the airflow to the aircraft is simply the aircraft total power ($\dot{E}$) reduced by the aircraft nominal power ($\dot{E}_n$):

$$\dot{E}_a = \dot{E} - \dot{E}_n. \tag{3.9}$$

3.3 Perpetual and Sustainable Flight Definitions

Definition III.1 (Perpetual Flight)**.** A flight is perpetual if the aircraft, starting with an initial altitude h_i and speed V_i, is able to fly always above or at a safety altitude h_{min} for infinite time.

Definition III.2. A *perpetual flight* initial energy is:

$$E_i := mgh_i + m\frac{V_i^2}{2}, \tag{3.10}$$

Definition III.3. A *perpetual flight* minimum energy is:

$$E_{min} := mgh_{min} + m\frac{V_{min}^2}{2}, \tag{3.11}$$

where h_{min} is the safety altitude and V_{min} is the minimum velocity required to maintain sustained flight, i.e., the stall speed at h_{min}.

Definition III.4 (Safe Flight)**.** A flight is safe if the aircraft altitude is always greater or equal to the safety altitude h_{min},

$$h \geq h_{min}, \forall t. \tag{3.12}$$

Definition III.5 (Sustainable Flight)**.** A flight is sustainable if the aircraft final total energy (E_f) is greater or equal to its initial total energy (E_i). That also implies that the average of the aircraft total energy rate must be positive or null,

$$\overline{\dot{E}} \geq 0, t \in [t_i, t_f]. \tag{3.13}$$

Lemma III.6. *If a flight is perpetual it is sustainable.*

Proof. From definitions III.1 and III.3 the energy variation along a *perpetual flight* is:

$$\Delta E \geq E_{min} - E_i. \tag{3.14}$$

The mean energy rate is

$$\overline{\dot{E}} := \frac{\int_{t_i}^{t_f} \dot{E} dt}{\int_{t_i}^{t_f} dt} = \frac{\Delta E}{\Delta t}. \tag{3.15}$$

As $\Delta t = \infty$ and $E_{min} - E_i$ is bounded, $\overline{\dot{E}} \geq 0$. □

Definition III.7 (Flight Cycle)**.** A flight cycle is a partition of a flight where one or more of the aircraft state or energy variables present the same final value as the initial and the same trend at the beginning and the end of the partition:

$$\kappa_i = \kappa_f \tag{3.16a}$$

$$\dot{\kappa}_f = a\dot{\kappa}_i, a \geq 0. \tag{3.16b}$$

Remark III.8. For thermal soaring cycles the cyclic variable is usually the total energy, while for wind shear soaring cycles the cyclic variables are usually the course, the bank, and the pitch angles.

Definition III.9 (Sustainable Flight Cycle)**.** A sustainable flight cycle is a partition of a sustainable flight, with:

$$\overline{\dot{E}_j} \geq 0, \tag{3.17}$$

where j is the cycle index.

Lemma III.10. *If a flight is only composed by an infinite sequence of flight cycles, it is a sustainable flight if all the cycles are sustainable (III.9).*

Proof. If a flight is only composed by sustainable and safe cycles, the whole flight mean energy rate will be

$$\overline{\dot{E}} = \frac{\sum_{i=1}^{N} \overline{\dot{E}_i}}{N}, \tag{3.18}$$

where N is the number of cycles. From its definition (III.9) a sustainable and safe cycle will have a null or positive mean energy rate ($\overline{\dot{E}_i} \geq 0$), yielding $\overline{\dot{E}} \geq 0$. □

Definition III.11 (Sustainable and Safe Flight Cycle)**.** A sustainable and safe flight cycle is a partition of a *perpetual flight*. It respects the *sustainable flight cycle* condition (III.9) and is safe by flying above the above the safety altitude h_{min}, yielding:

$$\overline{\dot{E}}_j \geq 0 \tag{3.19a}$$

$$h \geq h_{min}, \forall t, \tag{3.19b}$$

where j is the cycle index.

Lemma III.12. *If a flight is only composed by an infinite sequence of flight cycles, it will be a* perpetual flight *if all the cycles are sustainable and safe (III.11).*

Proof. If a flight is only composed by sustainable and safe cycles, the whole flight altitude will be greater or equal to the safety altitude, as each cycle altitude respects $h \geq h_{min}$. This condition extends for infinite time, as the flight is composed only by the infinite sequence of sustainable and safe cycles. □

3.4 Perpetuity Conditions in Thermal Fields

In this section we discuss the necessary conditions to achieve *perpetual flight* in thermal fields. In the literature there are some studies about the optimal flight settings for flight inside and between thermals. Two good examples are the MacCready Speed-to-Fly theory [55] and Cochrane's [56] extension to uncertain thermal activity and operation altitude limits. That said, we couldn't find any study about the necessary conditions to fly perpetually, with a neutral energy balance, or just to fly during a day, from morning to late afternoon.

We will focus on gliding flights to define the necessary conditions to fly perpetually in thermal fields. In a gliding flight the aircraft does not use the propulsion system to energize its flight. From here on, we will refer to a gliding aircraft as a glider. When the glider is flying inside a thermal it is propelled up by the thermal updraft, climbing. If it is flying outside any thermal it maintains the necessary speed to fly by losing altitude. To study the flight through thermal fields, also called thermaling, we need to take into account thermals' constraints, both in time and in space. We define the main factors and constraints to the problem of energy harvesting flight through a thermal field from the thermals' properties and dynamics described in the thermal models (sec. 2.5.2.1). The main factors affecting the ability to achieve *sustainable flight* are:

- The thermal field average updraft speed;
- The thermal field appearance rate;
- The thermals' average lifespan;
- The aircraft efficiency ($\frac{L}{D}_{max}$ and RC_{max}),

while the most relevant constraints to the energy harvesting process are the:

- Thermal lifespan (t_T);

- Thermal maximum altitude, i.e., the mixed-layer thickness z_i;
- Ground altitude.

As the thermal existence is limited to a finite lifespan, the glider is required to execute thermaling cycles, i.e., a sequence of climbs inside thermals intertwined with descents searching for active thermals. That is why we need to study the energy balance of flight cycle to define the necessary conditions to achieve *perpetual flight*.

We introduce some general assumptions about the flight settings and environmental conditions affecting the thermaling cycle, before analyzing conditions for *perpetual flight* in a thermal field:

Assumption III.13. *The wind horizontal speed is null or the thermal's drift is the same as the wind horizontal velocity.*

Assumption III.14. *During the thermaling cycle we assume the flight settings and flight path result in:*

1. *The airplane airspeed while it circles inside a thermal is fixed as* $V|_{RC_{max}}$*, meaning that the vertical speed relative to the air is* RC_{max}*,*

$$V_a = V|_{RC_{max}} \tag{3.20a}$$
$$\dot{h} = RC_{max}; \tag{3.20b}$$

2. *The air-climb angle* γ_a *is small enough to allow the approximations:*

$$\cos\gamma_a \approx 1 \tag{3.21a}$$
$$\sin\gamma_a \approx \gamma_a \tag{3.21b}$$
$$V_x \approx V_a; \tag{3.21c}$$

3. *The airplane airspeed while in transit between thermals is fixed as* $V|_{L/D_{max}}$*, meaning that the ratio between the lift and drag forces will be* L/D_{max}*, which with* (3.21) *yields:*

$$V_a = V|_{L/D_{max}} \tag{3.22a}$$
$$\dot{h} = \frac{V_a}{L/D_{max}}; \tag{3.22b}$$

Assumption III.15. *As proposed in section 2.5.2.7, the thermal appearance probability is defined by*

$$1 - e^{-\lambda_T \cdot s}, \tag{3.23}$$

where λ_T *is the thermal appearance rate, the inverse of the average distance among thermals, and* s *is the traveled distance since the last observed thermal.*

Furthermore, the thermal height and the ground altitude add altitude constraints to the energy harvesting process. These constraints, together with the stochastic nature of the thermal searching process, the thermal updraft strength, and the thermal lifespan, result in a stochastic process. This process is a chain of stochastic events, which include finding thermals at random altitudes, leaving thermals with a random gained altitude, and land-outs, i.e., unwanted landings, when no thermal is found before reaching the ground altitude. That means that the general thermaling process is quite complex to analyze. Therefore, we will analyze the simpler case of *sustainable flight*, to define necessary conditions for *perpetual flight.* Below, we present the further assumptions for the *sustainable flight* case analysis:

Assumption III.16 (Gaussian Thermal Lifespan)**.** *The thermal lifespan is a Gaussian:*

$$t_T \sim \mathcal{N}\left(\mu_t, \sigma_t\right); \tag{3.24}$$

Assumption III.17. *The thermal has no altitude bounds, z_i or ground altitude.*

Assumption III.18 (Updraft Daily Variation)**.** *w^* daily variation is given by,*

$$w^* \approx \frac{w^*_{max}}{2}\left[1 - \cos\left(2\pi \frac{t_d - t_{d0}}{t_{Day}}\right)\right], \tag{3.25}$$

where w^*_{max} is the day maximum w^*, t_d is the time of day, t_{d0} is the time of day when there is no convective activity (dawn), and t_{Day} is the time-length of a day.

With these assumptions we may define the necessary conditions for *sustainable flight* through a thermal field.

Theorem III.19 (Necessary Conditions for Sustainable Thermaling)**.** *It is only possible to perform a* sustainable flight *through a thermal field if:*

$$\lambda_T \mu_t \left(RC_{max} + \frac{\overline{w}_{max}}{2}\right) L/D_{max} \geq 1, \qquad \text{(Conditions for Sustainable Thermaling)}$$

where λ_T is the mean spacing among thermals, μ_t is the mean thermal lifespan experienced by aircraft, $\overline{w}_{max}$ is the day maximum of mean updraft speed ($\overline{w}$), RC_{max} is the aircraft minimum sink rate, and $(L/D)_{max}$ is the aircraft efficiency, i.e., the maximum ratio between the lift and drag forces.

Note that closer thermals ($\uparrow \lambda_T$), longer lifespans ($\uparrow \mu_t$), increased strength ($\uparrow \overline{w}_{max}$), lower aircraft sink rate ($\uparrow RC_{max}$), or higher aircraft efficiency ($\uparrow (L/D)_{max}$), lead to improved conditions for *perpetual flight.*

We proved theorem III.19 both through an altitude variation analysis and an energy balance analysis. In the next sections we present both derivations.

3.4.1 Altitude Hold Derivation

We may prove theorem III.19 by analyzing altitude variation during a thermaling cycle. As stated before, the thermaling cycle is composed by climb phase, inside a thermal, and a sink phase, while searching for other thermals. In order for an airplane to stay aloft the average sink ($\overline{\Delta h_S}$) should be compensated by the average thermaling climb ($\overline{\Delta h_T}$): $\overline{\Delta h_T} \geq -\overline{\Delta h_S}$, with

$$\overline{\Delta h_T} = \sum_{i=1}^{N} \frac{\int_{t_{T0}}^{t_{T1}} \dot{h} dt}{N} = \sum_{i=1}^{N} \frac{\Delta h_T}{N}, N \to \infty \tag{3.26a}$$

$$\overline{\Delta h_S} = \sum_{i=1}^{N} \frac{\int_{t_{S0}}^{t_{S1}} \dot{h} dt}{N} = \sum_{i=1}^{N} \frac{\Delta h_S}{N}, N \to \infty, \tag{3.26b}$$

where t_{T0} and t_{T1} are the time interval limits for each climb phase and t_{S0} and t_{S1} are the time interval limits for each sink phase. Taking into account (3.20) from assumption III.14 and the fact that while on a thermal the aircraft vertical speed (RC) sums to the thermal updraft speed, the gained height is,

$$\Delta h_T = \dot{h} \Delta t_T = \left(RC_{max} + \overline{w}\right) \Delta t_T, \tag{3.27}$$

where $\Delta t_T = t_{T1} - t_{T0}$. Given the independence between $\overline{w}$ and Δt_T, the average climb may also be defined as,

$$\overline{\Delta h_T} = \left(RC_{max} + \overline{w}_d\right) \mu_t, \tag{3.28}$$

where $\overline{w}_d$ is the updraft daily average.

From the *Updraft Daily Variation* assumption (III.18) and because the thermal average updraft is,

$$\overline{w} \propto w^*. \tag{3.29}$$

the daily updraft average ($\overline{w}_d$) is defined by,

$$\overline{w^*} = \frac{w^*_{max}}{2} \Rightarrow \overline{w}_d = \frac{\overline{w}_{max}}{2}, \tag{3.30}$$

where $\overline{w}_{max}$ is the day maximum for $\overline{w}$.

When in transit between thermals, the aircraft may lose an altitude of:

$$\Delta h_S = s \cdot \frac{RC}{V_a}, \tag{3.31}$$

where s is the flown distance relative to the air. Equation (3.22), in assumption III.14:

$$\frac{L}{D}_{max} = \frac{V_a}{RC}, \tag{3.32}$$

which means that lost altitude during the gliding phase is:

$$\Delta h_S = -\frac{s}{L/D_{max}}. \tag{3.33}$$

From assumptions III.13 and III.15, the average sink may now be defined as,

$$\overline{\Delta h_S} = -\frac{\overline{s}}{L/D_{max}} = -\frac{1}{\lambda_T L/D_{max}} \tag{3.34}$$

resulting in,

$$\begin{aligned} \left(RC_{max} + \frac{\overline{w}_{max}}{2}\right)\mu_t \geq \frac{1}{\lambda_T L/D_{max}} \Leftrightarrow \\ \Leftrightarrow \lambda_T \mu_t \left(RC_{max} + \frac{\overline{w}_{max}}{2}\right) L/D_{max} \geq 1, \end{aligned} \tag{3.35}$$

and proving theorem III.19.

3.4.2 Energy Balance Derivation

Let us now prove theorem III.19 through the energy balance analysis of the thermaling cycle. We will first define the energy balance in terms of the relation between the power source and power dissipation. Then we will define in more detail both the power source and power dissipation functions. To finish, we need to define the time spent in each power setting to evaluate the energy balance.

An unpowered aircraft can only maintain a *sustainable flight* if it harvests the same or more energy from the flow field then needed to stay aloft, i.e., if the overall energy variation is positive (def. III.9),

$$\Delta E = \int_{t_0}^{t_f} P_{Total} \geq 0, \tag{3.36}$$

where t_f is the final thermaling time and $P_{Total} = P_{in} - P_{out}$. In this case,

$$t_f \to \infty \tag{3.37a}$$

$$P_{in} = P_{FF} \tag{3.37b}$$

$$P_{out} = P_{Fly}, \tag{3.37c}$$

where P_{FF} is the power harvested from the flow field and P_{Fly} is the dissipated power needed to keep the aircraft aloft.

Now we will explain and define the power dissipation process. To stay aloft the aircraft needs to produce lift. In the particular case of airplanes, lift is created by the interaction between the air flow and the wing. To produce enough lift to stay aloft the aircraft needs to fly faster than a minimum airspeed, the stall speed. The interaction of the air flow with the wing also produces aerodynamic drag. Therefore, to maintain the airspeed the aircraft spends energy to overcome the drag force. For constant airspeeds the power required to keep the aircraft aloft is the power needed to overcome the aerodynamic drag force:

$$P_{Fly} = V_a D = C_D \rho S \frac{V_a^3}{2}, \tag{3.38}$$

where C_D is the drag coefficient, ρ is the air density, S is the aircraft wing area and V_a is the aircraft airspeed. In unaccelerated flight the aircraft forces are all balanced (fig. 3.1), and so:

$$0 = L \sin\gamma + D \cos\gamma \tag{3.39a}$$

$$mg = L \cos\gamma - D \sin\gamma, \tag{3.39b}$$

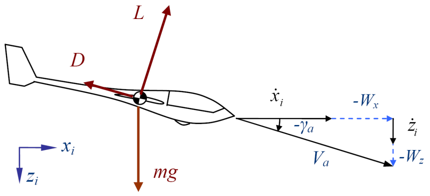


Figure 3.1: Forces affecting an aircraft (illustration from [3])

Taking into account (3.21) from assumption III.14 and because for the flight settings under analysis $L\cos\gamma >> D\sin\gamma$, the dissipated power may be approximated by,

$$P_{Fly} \approx -V_a mg\sin\gamma. \tag{3.40}$$

The dissipated power can then be defined in terms of the vertical velocity, $RC = V_a\sin\gamma$, as

$$P_{Fly} = -RCmg. \tag{3.41}$$

Note that in unpowered flight, i.e., gliding, γ is always negative, and so the required power is positive.

Similarly to the power dissipation, the power source is a function of a vertical velocity. In this case, where the energy is provided by the updraft, the power source is a function of the vertical airflow (w),

$$P_{FF} = wmg. \tag{3.42}$$

To conclude the definition of the power functions we now define the aircraft vertical velocity for each flight setting. If we consider the thermaling cycle, i.e., a search & climb flight path, we may assume the power to fly is constant for each of the two settings. As defined in (3.20) from assumption III.14 the search setting maximizes the flight range, with the maximum lift-over-drag ratio (L/D) and the corresponding C_D. The thermal climb setting maximizes the endurance, with the minimum C_D, i.e., minimum sink rate ($-RC_{max}$), also defined in assumption III.14 by (3.22). These settings yield,

$$RC_{C_{Dmin}} = RC_{max} = -C_{D_{min}}\rho S\frac{V^3_{a_{C_{Dmin}}}}{2mg} \tag{3.43a}$$

$$RC_{L/Dmax} \approx -\frac{V_{a_{L/Dmax}}}{L/D_{max}}. \tag{3.43b}$$

resulting in,

$$P_{Total} = \begin{cases} P_{FF} - P_{Fly} = wmg + RC_{max}mg & Climb \\ -P_{Fly} = -\frac{V_{a_{L/Dmax}}}{L/D_{max}}mg & Search \end{cases} \tag{3.44}$$

Now that we have the power functions we need to define the time spent in each flight setting, to finish the energy balance derivation. The condition $\Delta E \geq 0$ means that

$$\begin{aligned} & t_T \left(P_{FF} - P_{Fly}\right) - t_S P_{Fly} \geq 0 \Leftrightarrow \\ & \Leftrightarrow t_T w m g \geq -t_T RC_{max} m g + t_S \frac{V_{a_{L/Dmax}}}{L/D_{max}} m g, \end{aligned} \tag{3.45}$$

where t_T and t_S are respectively the sum of all the time spent climbing on a thermal and the sum of all the time spent sinking, searching for new thermals. The expected value of t_T for N_T exploited thermals is:

$$E\left(t_T\right) N_T = \mu_t N_T. \tag{3.46}$$

t_S is the sum of all time flying between thermals, which is also the ratio between the distance between thermals and the search speed:

$$t_S = \frac{\sum s_T}{V_{a_{L/Dmax}}} \approx \frac{N_T}{\lambda_T V_{a_{L/Dmax}}}, \tag{3.47}$$

If we rearrange equation (3.45) with (3.46) and (3.47), we get:

$$\begin{aligned} & \mu_t N_T w \geq -\mu_t N_T RC_{max} + \frac{N_T}{\lambda_T L/D_{max}} \Leftrightarrow \\ & \Leftrightarrow \lambda_T \mu_t L/D_{max} \left(w + RC_{max}\right) \geq 1. \end{aligned} \tag{3.48}$$

When we substitute the daily average of the updraft speed (w) from (3.30) in (3.48), we obtain the same necessary conditions for a *sustainable flight* (3.49) as in section 3.4.1:

$$\lambda_T \mu_t L/D_{max} \left(RC_{max} + \frac{\overline{w}_{max}}{2}\right) \geq 1, \tag{3.49}$$

further proving theorem III.19.

3.4.3 Discussion

If we consider a specific aircraft the aerodynamic parameters RC_{max} and $(L/D)_{max}$ are constant. The lifespan distribution may also be considered constant for anywhere in the world. This is because we assume it independent of the thermal strength. This assumption holds if we also assume soft wind shear, as the thermal's disruption depends on the ratio between the thermal strength and wind shear [4].

According to Schuemann [57], the typical condition in terms of thermal updraft speed for the eastern U.S. is about 430 ft/min (2.2 m/s). A model glider like the SB-XC presents an $\frac{L}{D}_{max} = 27$ and $RC_{max} = -0.58 m/s$ [9, 31]. The thermals' lifespan mean is usually around 20 minutes [46, 4, 28]. If we take $\overline{w}_{max} = 2.2 m/s$, we have:

$$\lambda_T \geq \frac{1}{1200 \cdot 27 \left(\frac{2.2-0.58}{2}\right)} = 5.9 \cdot 10^{-5} m^{-1}. \tag{3.50}$$

Taking the data for Desert Rock in Nevada shown in table 2.2, the thermals height is likely to be in the interval from $150m$ to $3.638m$. By (2.98), the appearance rate

(λ_T) is then likely to be in the interval from $3.3 \cdot 10^{-4} m^{-1}$ to $8.0 \cdot 10^{-3} m^{-1}$, which is well above $5.9 \cdot 10^{-5} m^{-1}$. According to theorem III.19 this means that a *sustainable flight* would be possible in that area with this model glider with a very good margin.

This does not assure a *perpetual flight* is possible, because the *sustainable flight* relates only to the energy and disregards the altitude constrains. This is a necessary condition for *perpetual flight*, meaning that if this condition is not met, it wont be possible to achieve a *perpetual flight.* Moreover, if the aircraft is able to harvest energy and collect it internally (batteries) instead of in potential energy (altitude), a *sustainable flight* condition could also be a lower bound condition for *perpetual flight.*

3.5 Perpetuity Conditions in Wind Shear

By lemma III.10, a *perpetual flight* is possible if the necessary conditions to perform a *sustainable and safe flight cycle* are present. As such we may state that:

Proposition III.20. *The main factors affecting the ability to achieve a* sustainable and safe flight cycle *by harvesting wind shear energy, and therefore enabling the possibility to perform a* perpetual flight*, are:*

- *The wind shear top and bottom altitudes;*
- *The wind shear gradient magnitude;*
- *The aircraft efficiency* ($\frac{L}{D}_{max}$).

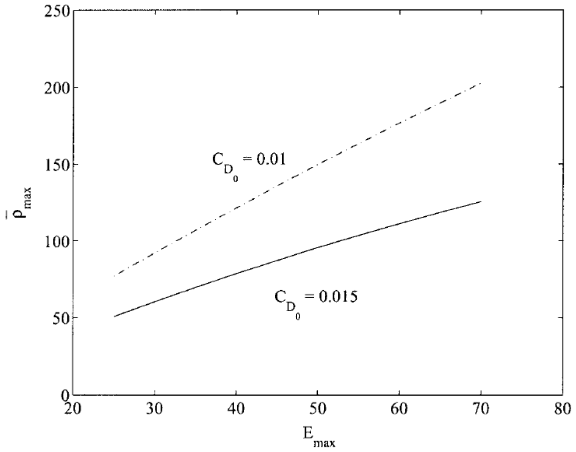


Figure 3.2: Minimum required wind gradient for basic soaring pattern (illustration from [6]). $\overline{\rho} = \frac{\rho g^2}{2(mg/S)\frac{dWx}{dz}^2}$ is an adimensional value, representing the wind shear gradient, the wing loading, the air density and the gravity. $E_{max} = L/D_{max}$ is the best lift over drag ratio and C_{D_0} is the minimum drag ratio.

Zhao [6] studied several optimal soaring cyclic trajectories. Those trajectories minimized the cycle time, maximized the gained altitude or minimized the required wind shear gradient required to maintain flight (fig. 3.2). This latest provides one of the most important results from the study, the minimum required conditions to maintain a *perpetual flight* by harvesting energy from a linear wind shear gradient.

Taking the results illustrated in figure 3.2, we can approximate the inequality defining the minimum conditions necessary to perform *sustainable and safe flight cycles* in wind shear:

$$\frac{\overline{dWx}^2}{dz} \geq \frac{\rho g^2}{(mg/S)} \frac{C_{D_0}}{\left(0.0644 - 1.12 \cdot 10^{-4} \frac{L}{D_{max}}\right) \frac{L}{D_{max}}} \approx \frac{\rho g^2}{(mg/S)} \frac{C_{D_0}}{0.059 \frac{L}{D_{max}}}, \quad (3.51)$$

where $\overline{\frac{dWx}{dz}}$ is the average wind shear gradient, ρ is the air density, g is the gravity acceleration, C_{D_0} is the minimum drag ratio, L/D_{max} is the best lift over drag ratio, which defines the aircraft aerodynamic efficiency, and $\frac{mg}{S}$ is the wing loading (m is the aircraft mass and S is the wing area).

In the case of *Layer Wind Shear* (sect. 2.5.1.2) the phenomenon gradient is almost constant throughout the shear layer and its magnitude is defined by the wind velocity vector at the top and bottom altitudes of the layer. The wind shear top and bottom altitudes depend on the phenomenon which originated the shear, as *inversion layers* and the *Jet Stream.*

Sachs and Costa [7] present some results for the minimum required wind shear gradient below the *Jet Stream* to allow *sustainable and safe flight cycles* (fig. 3.3). In this example they take into account the air density variation with the altitude, as the wind shear layer thickness is quite large. Further, the flight path optimization respected constrains to the lift coefficient (C_L) and the load factor (n).

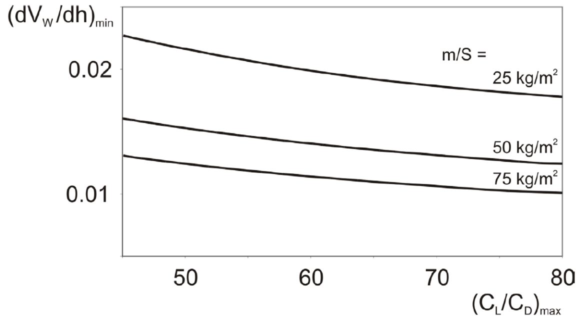


Figure 3.3: Minimum required wind gradient ($\frac{dV_w}{dh}_{min} = \frac{dW_z}{dz}_{min}$) for dynamic soaring over the wind shear below the *Jet Stream*, with the lift and load factor constrained to $C_{L,max} = 1.5$ and $n_{max} = 4.5G$. (illustration from [7])

These results are consistent with (3.51), where the minimum required wind shear gradient ($\frac{dW_z}{dz}_{min}$) grows with decreasing aerodynamic efficiency (L/D_{max}) or wing loading $\frac{mg}{S}$.

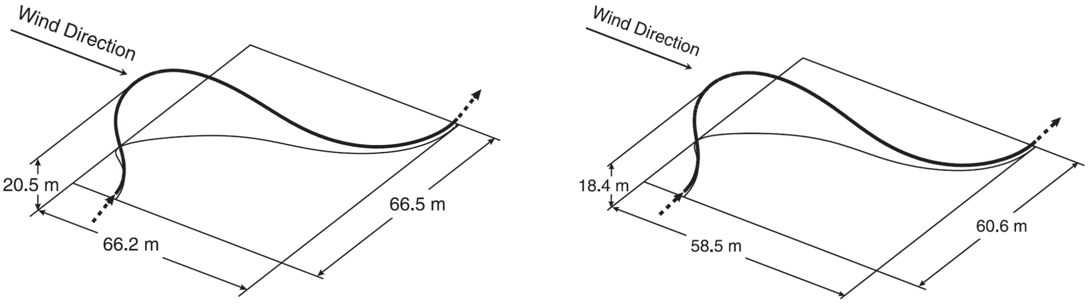
In the case of *Surface Wind Shear* (sect. 2.5.1.1) the phenomenon gradient is defined by the wind velocity at the reference altitude, for the average wind speed gradient, and by the *roughness altitude*, for the gradient steepness near the surface. The wind shear is only limited below by the surface altitude, but because the gradient gets almost null above the reference altitude, we can take this as an upper gradient limit.

Sachs [8] computed the minimum wind velocity required to generate a strong enough flow gradient to propel the albatross flight (fig. 3.4). He concluded that albatross would require a wind of $\|\mathbf{w}_H\| = 8.6m/s$, at a reference altitude (h_{Ref}) of 10m, if they could control the lift coefficient optimally, or $\|\mathbf{w}_H\| = 8.9m/s$, if they would hold a constant lift coefficient. In these computations the estimated albatross wing aerodynamic parameters were:

$$\frac{L}{D}_{max} = 20 \tag{3.52a}$$

$$\frac{mg}{S} = 128N/m^2 \tag{3.52b}$$

$$C_{D_0} = 0.033. \tag{3.52c}$$



(a) Dynamic soaring optimal path with variable lift coefficient. To propel the albatross flight, the minimum wind speed required is $\|\mathbf{w}_H\| = 8.6m/s$.

(b) Dynamic soaring optimal path with fixed lift coefficient. To propel the albatross flight, the minimum wind speed required is $\|\mathbf{w}_H\| = 8.9m/s$.

Figure 3.4: Dynamic soaring optimal path for *Surface Wind Shear*. (Illustrations from [8])

3.6 Perpetuity Conditions in Gusts

Proposition III.21. *The main factors affecting the ability to achieve* perpetual flight *by harvesting gust energy are:*

- *The gust magnitude and wavelength;*
- *The aircraft efficiency* ($\frac{L}{D}_{max}$).

Langelaan [9, 31] studied the possibility of harvesting energy from wind gusts. One of his results shows clearly that harvesting gust energy is possible with a controller designed accordingly. Figure 3.5 shows the energy balance of flights over a sinusoidal gust field with a normal controller, i.e., regulating the velocity to be one that provides maximum L/D with steady wind, and a controller specifically synthesized to harvest gust energy. These controllers are tested with 3 sets of sinusoidal wind gust wavelengths (25m, 50m and 100m). As expected both controllers perform similarly when no gusts are present. Although, for non-null gusts the normal controller loses more energy, while the controller designed to exploit the gusts gains energy. In fact, the flight energy balance becomes positive for gusts above 2 m/s for all gust wavelengths, in the presented simulations. As a note, the simulations used the parameters of a SB-XC model glider with $\frac{L}{D}_{max} = 27$ and $\frac{mg}{S} = 98N/m^2$ [9, 31].

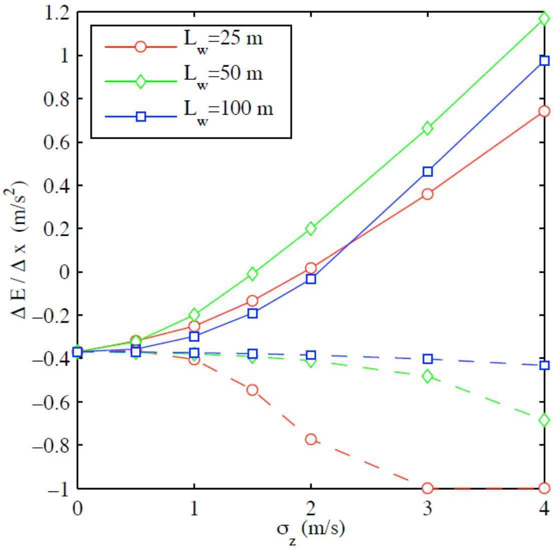


Figure 3.5: Comparison of energy balance of two different flight controllers, with different gust conditions. (illustration from [9])

The main conclusion we may take from the example above is that the main factors which affect the capacity to perform perpetual flight are: the gusts' magnitude and wavelength, the aircraft efficiency ($\frac{L}{D}_{max}$), and of course the flight controller. The aircraft efficiency defines the starting point for the energy balance, i.e., the energy loss without gusts. The harvested energy grows with the gusts' magnitude, but the growth rate seems to depend on the gusts' wavelength. This may be due to resonant dynamics, which would make the aircraft mass, the pitch damping, the response to the elevator deflection and the response to the gusts important factors in the conditions for *perpetual flight* with gusts.

3.7 Conclusions

This chapter defines the necessary conditions for *perpetual flight*. An interesting and intuitive conclusion from the perpetuity flight conditions analysis is that the aircraft aerodynamic efficiency is a key factor in the conditions to achieve *perpetual flight* with any of the three described phenomena. Another key factor is the magnitude of each phenomena strength, i.e., the updraft speed in thermals, the flow gradient in wind shear, and the gust magnitude in gusts. Both these conclusions make a lot of sense, as a more efficient aircraft loses less energy and a stronger phenomenon is capable of conveying more energy.

All the studied phenomena harvest energy in a cyclic manner. The gust phenomenon has cycles that can last a few seconds at most. An energy harvesting flight cycle in wind shear may last for a few minutes. Thermals may last half an hour and the search flight to find other thermals tens of minutes, so it is long cycle, but still a cycle. Therefore, the concept of *sustainable and safe flight cycle* is central to the study of *perpetual flight*.

Because of the short cycle length and the required maneuverability, the exploitation of gusts and wind shear may be limited to smaller aircraft, as small Unmanned Aerial Vehicles (UAVs). Thermals are usable by any aircraft that is able to stay inside its updraft core, as gliders do. As such, if there is controller prepared to exploit these phenomena, the aircraft exploiting the phenomena is efficient and maneuverable enough, and the phenomenon itself is strong enough, it is possible to execute *perpetual flight*.

CHAPTER IV

Flow Field Estimation

4.1 Introduction

This chapter describes the estimation processes developed to allow the tracking and characterization of the studied flow field phenomena. The studied estimation methods assume that the flow field observations are taken by an aircraft, through the sensors that compose its avionics suite. The information collected about these phenomena allows us to optimize the flight trajectories through the flow field. In this work we are mainly focused on the energy harvesting optimization, increasing the flight endurance or reducing required energy storage for the same flight endurance. But the gathered information may also improve other performance variables, such as in-route time.

There are path control methods that use implicit localization to guide the aircraft. These methods use a direct feedback from the flow field measurements, or its energy, to control the flight path. One such method is the thermal centering controller presented by Andersson and Kaminer [29] and the gust exploration described by Langelaan [9, 31]. In the case of gusts the direct feedback may be the only feasible method due to the transitory and random nature of turbulence, which is the main factor in gusts. In the case of phenomena that exist almost unchanged for longer periods, such as thermals and wind shear, the explicit estimation enables a better and wider exploration, i.e., an optimization of the flight path over the surrounding region. As stated before, an explicit estimation enables an optimized path control, mitigating the effects of the random perturbations in the phenomena flow or energy measurements and enabling the use of path planing strategies. Furthermore, the explicit estimation process may also enable the optimization of the estimation itself, by the identification of flight paths that yield the most informative measurements. But maybe even more important, the explicit estimation allows a team of aircraft to share that information, extending the potential for energy harvesting by the whole team.

The estimation methods described in this chapter use the aircraft as a sensor, providing inputs to the estimators. The estimators use data such as the vertical flow measurement or the difference between the airspeed and the ground-speed in different directions. Because of that, the estimators input data depends on the aircraft dynamics. As the aircraft dynamics are nonlinear (sec. 2.4), the relation between the estimators input data and the main variables driving the aircraft motion is nonlinear. Moreover, we intend to verify whether it is possible to estimate the parameters of a flow field phenomenon by checking if these are observable with the available input data. As such, we describe the method for checking the observability of nonlinear systems (sec. 4.2). We then apply this method to the problems of observing thermals (sec. 4.3) and wind shear (sec. 4.4).

The *Thermal Estimation* section (4.3) first deals with the observability of thermal parameters (sec. 4.3.1), such as the center position and speed, and the updraft flow field parameters. Further, it describes a Particle Filter estimator for *Chimney Thermals* and the estimation results (sec. 4.3.2). To finish this section we describe the problems associated with estimating *Bubble Thermals* (sec. 4.3.4).

The *Wind Shear Estimation* section (4.4) starts by the study of the observability conditions to estimate the wind components (sec. 4.4.1) and then focuses on a Particle Filter estimator for wind shear and its estimation results.

4.2 Observability of Nonlinear Systems

In this section we present the method to study the flow field phenomena observability. Because the phenomena will be observed by an aircraft, the relation between the driving variables, i.e., the main variables of the aircraft dynamics, and the observed parameters is nonlinear. Therefore, to test the observability of the phenomena's parameters we cannot use the same methods as for linear systems, such as the Grammian rank matrix test [58]. However, for these nonlinear systems we can determine local weak observability [59] using Lie derivatives [60]. Similarly to linear systems, we build an observability matrix and check its rank to test the observability of a nonlinear system. The Lie derivatives are the rows of this observability matrix.

Let us consider a dynamic system described by:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}\left(\mathbf{x}, \mathbf{u}\right) \\ \mathbf{y} = \mathbf{h}\left(\mathbf{x}\right) \end{cases} \tag{4.1}$$

where $\mathbf{f}\left(\mathbf{x}, \mathbf{u}\right)$ is the process function and $\mathbf{h}\left(\mathbf{x}\right)$ is the measurement function.

By convention the zeroth-order derivative satisfies:

$$\mathcal{L}^0 \mathbf{h} = \mathbf{h}, \tag{4.2}$$

The first-order Lie derivative of a scalar function h with respect to $\mathbf{f}$ is the inner product of $\mathbf{f}$ with the function (h) gradient:

$$\mathcal{L}^1_{\mathbf{f}} h = \nabla h \cdot \mathbf{f} = \frac{\partial h}{\partial x_1} f_1 + ... + \frac{\partial h}{\partial x_N} f_N, \tag{4.3}$$

where ∇ is the gradient operator and $\mathbf{f} = [f_1, ..., f_N]^\intercal$. The higher order Lie derivatives are first-order Lie derivatives applied over lower order Lie derivatives:

$$\mathcal{L}^l_{\mathbf{f}} h = \nabla \mathcal{L}^{l-1}_{\mathbf{f}} h \cdot \mathbf{f} = \mathcal{L}^1_{\mathbf{f}} \left(\mathcal{L}^{l-1}_{\mathbf{f}} h\right) = \mathcal{L}^1_{\mathbf{f}} \left(\mathcal{L}^1_{\mathbf{f}} \left(... \left(\mathcal{L}^1_{\mathbf{f}} h\right)\right)\right). \tag{4.4}$$

For example, the second order Lie derivative is

$$\mathcal{L}^2_{\mathbf{f}} h = \mathcal{L}^1_{\mathbf{f}} \left(\mathcal{L}^1_{\mathbf{f}} h\right) = \nabla \mathcal{L}^1_{\mathbf{f}} h \cdot \mathbf{f}. \tag{4.5}$$

The observability matrix rows are the increasing order Lie derivatives of the observation function ($\mathbf{h} = [h_1, ..., h_m]^\intercal$),

$$\mathcal{O} := \begin{bmatrix} \nabla h_1 \\ \vdots \\ \nabla h_m \\ \nabla \mathcal{L}^1_{\mathbf{f}} h_1 \\ \vdots \\ \nabla \mathcal{L}^1_{\mathbf{f}} h_m \\ \nabla \mathcal{L}^2_{\mathbf{f}} h_1 \\ \vdots \\ \nabla \mathcal{L}^2_{\mathbf{f}} h_m \\ \vdots \end{bmatrix} = \begin{bmatrix} \nabla \mathbf{h} \\ \nabla \dot{\mathbf{h}} \\ \nabla \ddot{\mathbf{h}} \\ \vdots \end{bmatrix}. \tag{4.6}$$

When the process function ($\mathbf{f}$) can be divided in independent parts $\mathbf{f}_i, i = 1, ..., n$ excited by each component of the control vector ($\mathbf{u}$), the dynamic system may described by:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \mathbf{f}_1(\mathbf{x}, u_1) + \mathbf{f}_2(\mathbf{x}, u_2) + ... + \mathbf{f}_N(\mathbf{x}, u_N) \\ \mathbf{y} = \mathbf{h}(\mathbf{x}) \end{cases} \tag{4.7}$$

where $\mathbf{f}_0$ is the zero-input process function. With the dynamic system (4.7), there is an nth-order Lie derivative for each $\mathbf{f}_i$ [61]. Therefore the first-order Lie derivatives of the function h with respect to $\mathbf{f}_i$ are the inner products of $\mathbf{f}_i$ with the function (h) gradient:

$$\mathcal{L}^1_{\mathbf{f}_i} h = \nabla h \cdot \mathbf{f}_i = \frac{\partial h}{\partial x_1} f_{i1} + ... + \frac{\partial h}{\partial x_N} f_{iN}, \tag{4.8}$$

where $\mathbf{f}_i = [f_{i1}, ..., f_{iN}]^\intercal$. As for the general case described above, the higher order Lie derivatives are first-order Lie derivatives applied over lower order Lie derivatives:

$$\mathcal{L}^l_{\mathbf{f}_i \mathbf{f}_j ... \mathbf{f}_k} h = \nabla \mathcal{L}^{l-1}_{\mathbf{f}_j ... \mathbf{f}_k} h \cdot \mathbf{f}_i = \mathcal{L}^1_{\mathbf{f}_i} \left(\mathcal{L}^{l-1}_{\mathbf{f}_j ... \mathbf{f}_k} h \right) = \mathcal{L}^1_{\mathbf{f}_i} \left(\mathcal{L}^1_{\mathbf{f}_j} \left(... \left(\mathcal{L}^1_{\mathbf{f}_k} h \right) \right) \right) \tag{4.9}$$

where $\mathbf{f}_i$, $\mathbf{f}_j$, and $\mathbf{f}_k$, with $i, j, k = 1, ..., n$, may be the same part of the process function or not.

As such, the observability matrix is,

$$\mathcal{O} := \left\{ \nabla \mathcal{L}^l_{\mathbf{f}_i ... \mathbf{f}_j} h_k : i, j = 1, ..., n; l \in \mathbb{N} \right\}. \tag{4.10}$$

This matrix can now be used to study the observability of the wind state and the thermal state and parameters.

4.3 Thermal Estimation

Some thermal soaring controllers require the knowledge of the thermals localization. Updraft effects can be detected by aircraft. Both the vertical airflow and the energy balance can be computed through the aircraft sensing data. The energy balance allows us to quantify how much airflow energy is being harvested by the aircraft, particularly in the vertical axis (sec. 4.3.2.7). As such, the aircraft can be regarded as a sensor, and the thermals as the objects whose parameters and state have to be estimated. This is a global localization problem. Further, the nonlinear nature of the problem is emphasized by the necessity to estimate the thermal shape and strength. Thermal characterization encompasses position, size, shape, vertical speed gradient, dynamics, number, etc. This characterization depends on the underlying model. To characterize the position we can assume the flow has a geometric center, which can be localized. The shape can be modeled as a circle or an ellipse on the horizontal plane. On the vertical plane, the thermal radius is a function of the altitude above the ground. The vertical speed usually varies from the highest upward speed in the center of the thermal to a downward speed on the skirts, and a slight downward velocity in the area between thermals. Furthermore, thermal spots can change, moving around, shrinking or enlarging, becoming stronger or fading. These dynamics depend on the terrain characteristics (plain or rough, hot spot type, etc.), the wind, the sun, the cloud coverage, etc. Section 2.5.2.1 describes in more detail the models used to develop the thermal estimators.

Also important are the number and relative position between thermal spots (sec. 3.4). Updrafts are only useful to an aircraft if they are spread around the desired path.

The process of localizing thermals is sometimes compared to the localization of hydrothermal vents, which create a hydrothermal plume. But, unlike hydrothermal vents [62], thermal updrafts are fairly large features, with a diameter between 100m and 1000m [63]. Further, the thermal energy or flow manifestations are mostly contained within the thermal itself. Cumulus clouds created above a thermal can be identified from larger distances, but the computation of imagery data for this purpose is not trivial. As such, to localize thermals we need to observe the air flow inside the updraft itself.

Allen [34] presented a centroid-based method for dynamic localization of thermal centers. The developed system was demonstrated in real flight tests over Edwards Air Force Base. The algorithm had performance problems due to filtering delays, resulting in considerable thermal center localization errors [11]. Edwards [11] extended Allen's approach, mixing it with Wharington's [27] neural-network based locator. The flight results were quite good, demonstrating highly effective thermal energy harvesting. Both methods assumed the estimation of a single thermal at each time.

4.3.1 Thermal Observability

In this section we study the observability of thermals by aircraft flying through the updraft and measuring it. We will start by the simplest 2D localization. Next we will study the observability of more detailed models and conclude with the study of the complete thermal model parameters observability (sec. 2.5.2.1).

The main characterizing parameters of the updraft flow field over a horizontal plane are the mean strength or mean vertical velocity ($\overline{w}$), the peak strength (w_{peak}), the thermal radius (r_2), also designated outer radius, and the inner radius (r_1). These variables are related by the equations (2.77) and (2.78), which we rewrite here:

$$\frac{r_1}{r_2} = \begin{cases} 0.0011r_2 + 0.14 & r_2 < 600 \\ 0.8 & otherwise \end{cases} \tag{4.11a}$$

$$w_{peak} = \overline{w}\frac{3r_2^2\left(r_2 - r_1\right)}{r_2^3 - r_1^3}. \tag{4.11b}$$

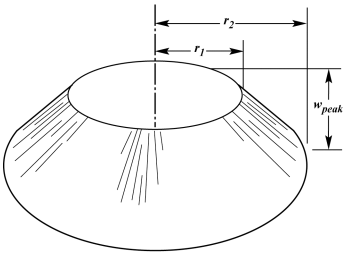


Figure 4.1: Updraft velocity trapezoidal model (illustration from [4])

For sake of simplicity in the next analysis we will use the trapezoidal shape model (fig. 4.1) to describe the updraft flow field over a horizontal plane,

$$w = \begin{cases} r_2 \geq d \geq r_1 & \delta w_d\left(d - r_2\right) \\ d \leq r_1 & w_{peak} \\ d \geq r_2 & 0 \end{cases} \tag{4.12a}$$

$$\delta w_d = -\frac{3\overline{w}}{r_2\left(1 - k_r^3\right)} \tag{4.12b}$$

This trapezoidal updraft model presents the most significant updraft field features of the thermal models (sec. 2.5.2.1). It presents a positive updraft gradient from the thermal skirts to the core and a constant core updraft.

From (4.12a) it is easy to notice that the local observability problem should be divided according to d. In the two cases where $d \leq r_1$ and $d \geq r_2$ the thermal center coordinates are unobservable, as the observation w is constant for any flight trajectory, yielding a rank deficient observability matrix (4.10). Therefore, we will focus on the case where the aircraft is flying through a varying updraft ($w = \delta w_d\left(d - r_2\right) \Leftarrow r_2 \geq d \geq r_1$).

4.3.1.1 2D Thermal Center Localization

The 2D localization is the sole observation of the thermal center coordinates (x_T, y_T). This does not include the characterization of the updraft flow field. As such, the 2D localization is only possible if the aircraft measuring the updraft is flying at a constant radius from the thermal center. That is due to the fact that there is no knowledge about the updraft field, and so the variation of the updraft measurement cannot be interpreted. Only the measurement of a constant updraft speed is meaningful, as it means that the aircraft is flying at a constant radius from the thermal center. Then, it is possible to observe the thermal position from the flight path curvature. In this context the thermal centering controller presented by Andersson and Kaminer [29] may be interpreted as an implicit thermal localization algorithm. It does not explicitly estimate the thermal center, but it uses constant updraft measurement to identify a constant radius to the thermal center.

To enable the localization of a thermal center with paths not centered on the thermal we need to characterize the updraft flow field, i.e., the updraft outer radius and radial gradient. The 2D localization and characterization is the estimation of the thermal center coordinates (x_T, y_T) and the updraft flow field parameters over a horizontal plane, also called updraft flow field plane parameters. The relevant equations are then:

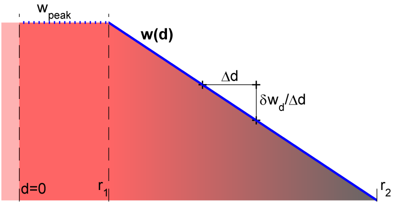


Figure 4.2: Updraft observation function.

$$w = -\delta w_d \left(r_2 - d\right) \tag{4.13a}$$

$$\delta w_d = -\frac{3\overline{w}}{r_2\left(1-k_r^3\right)} \tag{4.13b}$$

$$d = \sqrt{\Delta x^2 + \Delta y^2} \tag{4.13c}$$

$$k_r = \frac{r_1}{r_2} = \begin{cases} 0.0011 r_2 + 0.14 & r_2 < 600 \\ 0.8 & otherwise \end{cases}, \tag{4.13d}$$

where d is the aircraft distance to the thermal center, and Δx and Δy are the thermal position coordinates relative to the aircraft:

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} x - x_T \\ y - y_T \end{bmatrix} = d \begin{bmatrix} \cos\gamma \\ \sin\gamma \end{bmatrix}, \tag{4.14}$$

where x and y are aircraft position coordinates, and γ is the relative azimuth, i.e., the angle of the thermal center relative to the aircraft position. When the aircraft is measuring the updraft speed (w), equations (4.13) present seven unknown variables (δw_d, r_2, $\overline{w}$, d, Δx, Δy, and k_r). Most of these variables are illustrated in figure 4.2. This means that we need to observe three of the variables to fully define the state.

Considering (4.14) and the aircraft kinematics described in section 2.4.2, the kinematic equations for the thermal/aircraft system are:

$$\begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} = \begin{bmatrix} V_a \cos\left(\psi\right) + W_x - u_T \\ V_a \sin\left(\psi\right) + W_y - v_T \end{bmatrix} = \begin{bmatrix} \Delta V \cos\left(\varphi\right) \\ \Delta V \sin\left(\varphi\right) \end{bmatrix}, \tag{4.15}$$

where u_T and v_T are the thermal center velocity components, W_x and W_y are the wind velocity components, V_a is the aircraft airspeed, ψ is the aircraft heading, ΔV is the norm of the relative velocity and φ is the relative course. Notice that the thermal velocity influences the kinematic equations exactly as an offset of the wind velocity.

For the observability study we define the relevant state vector and observation function (fig. 4.2) as:

$$\mathbf{x} = \begin{bmatrix} d \\ \gamma \\ r_2 \end{bmatrix} \tag{4.16a}$$

$$h\left(\mathbf{x}\right) = w = \delta w_d \left(d - r_2\right), \tag{4.16b}$$

yielding the process function defined below,

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{d} \\ \dot{\gamma} \\ \dot{r_2} \end{bmatrix} = \begin{bmatrix} \Delta V \cos\left(\varphi\right) \\ \Delta V \sin\left(\varphi\right)/d \\ 0 \end{bmatrix}. \tag{4.17}$$

Therefore, the observability matrix is:

$$\mathcal{O} = \begin{bmatrix} \nabla h \\ \nabla \dot{h} \\ \nabla \ddot{h} \end{bmatrix} = \delta w_d \begin{bmatrix} 1 & 0 & -\frac{d}{r_2} \\ 0 & d\dot{\gamma} & -\frac{\dot{d}}{r_2} \\ -\dot{\gamma}^2 & \dot{d}\left(\dot{\varphi} - 2\dot{\gamma}\right) & -\frac{\ddot{d}}{r_2} \end{bmatrix}. \tag{4.18}$$

The determinant is,

$$\det\left(\mathcal{O}\right) = \delta w_d^3 \frac{\dot{\gamma}^3 d^2 - \dot{\varphi}\dot{d}^2}{r_2}. \tag{4.19}$$

Theorem IV.1. *The thermal position and all the updraft flow field plane parameters* (4.13) *are locally weakly observable by an aircraft flying trajectories with* $\dot{\varphi} \neq \dot{\gamma}\tan^2\left(\varphi - \gamma\right)$*, as long as the trajectory is included in the area defined by* $r_2 \geq d \geq r_1$*. This holds for the trapezoidal model (Fig. 4.1, eq.* (4.12a)*).*

Proof. For $d \geq r_2$ and $d \leq r_1$ w_W is constant producing a rank deficient observability matrix. This means that the full state $\mathbf{x}_T$ can't be observed if $d > r_2$ or $d < r_1$.

When $r_2 \geq r_V \geq r_1$ the observability matrix determinant (4.19) is not null if:

$$\dot{\gamma}^3 d^2 \neq \dot{\varphi}\dot{d}^2 \tag{4.20a}$$

$$\delta w_d \neq 0 \tag{4.20b}$$

$$r_2 \neq \infty. \tag{4.20c}$$

The last two conditions are true by the definition of thermals in the area defined by $r_2 \geq d \geq r_1$, as the radius is limited and there is an updraft speed variation with the distance to the thermal center. The first condition (4.20a) is equivalent to $\dot{\varphi} \neq \dot{\gamma}\tan^2\left(\varphi - \gamma\right)$, as $\dot{\gamma}^2 d^2 = \Delta V^2 \sin^2\left(\varphi - \gamma\right)$ and $\dot{d}^2 = \Delta V^2 \cos^2\left(\varphi - \gamma\right)$. □

From theorem IV.1 we can define the necessary flight conditions for the observability of the thermal position and the updraft flow field plane parameters:

The aircraft cannot fly at a constant distance from the thermal center. This condition is easily verified as $\dot{d} = 0$ would zero all rows but the first of the observability matrix (4.18). It is also intuitive that the aircraft should need to sweep different distances from the thermal center to be able to estimate the flow field shape.

The aircraft should be flying around the thermal or turning This is also intuitive, as the aircraft needs to observe several points around the thermal to be able to estimate its center. If it is not circling around the thermal ($\dot{\gamma} = 0$), but it is turning ($\dot{\varphi} \neq 0$), the condition $\dot{\gamma} = 0$ will only be true for an instant.

4.3.1.2 2D Thermal Center Tracking and Characterization

The thermal planar tracking and characterization is an extension of the previous problem. By that we denote the estimation of the thermal center and speed (x_T, y_T, u_T, v_T) and the estimation of the updraft flow field parameters over the horizontal plane.

The relevant state is:

$$\mathbf{x}_T = \begin{bmatrix} x_T \\ y_T \\ u_T \\ v_T \\ r_2 \end{bmatrix}. \tag{4.21}$$

We divide this problem into two parts: the first is the observability of x_T, y_T, r_2, and the second is observability of u_T, v_T. The first part is solved by theorem IV.1. The second part is an observability problem where we can consider x_T, y_T as observations, and so:

$$\mathbf{x} = \begin{bmatrix} u_T \\ v_T \end{bmatrix} \tag{4.22a}$$

$$\mathbf{h} = \begin{bmatrix} x_T \\ y_T \end{bmatrix}. \tag{4.22b}$$

Lemma IV.2. *If a system can estimate a position of a point, it can also observe the velocity of the same point.*

Proof. The observability matrix for (4.22b) is:

$$O = \nabla \begin{bmatrix} x_T \\ y_T \\ u_T \\ v_T \end{bmatrix} = C \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.23}$$

which is full rank, proving lemma IV.2. □

We now may extend theorem IV.1 to moving thermals.

Theorem IV.3. *The thermal position, velocity and updraft flow field planar parameters* (4.21) *are locally weakly observable by an aircraft flying trajectories with* $\dot{\varphi} \neq \dot{\gamma} \tan^2(\varphi - \gamma)$*, as long as the trajectory is included in the area defined by* $r_2 \geq d \geq r_1$*. This holds for the trapezoidal model (Fig. 4.1, eq.* (4.11)*).*

Proof. Theorem IV.3 together with lemma IV.2 prove theorem IV.1. ☐

4.3.2 Chimney Thermal Estimator

When flying over a thermal, an aircraft is affected by the thermal local vertical flow. As such, it may estimate the thermal state through its energy measurements. If we compare this problem with the widely studied robot localization problem, we may interpret the thermal as the agent being localized and the aircraft as the sensor.

Due to the global localization nature of the problem and the nonlinear nature of the thermal model (sec. 2.5.2.2) we choose to implement a Particle Filter. We present four Particle Filter (PF) versions (sec. 4.3.2.1, 4.3.2.2, 4.3.2.3, and 4.3.2.4) to estimate the thermal parameters, and the respective *Propagation* and *Observation* models (sec. 4.3.2.6 and 4.3.2.7).

The aircraft is regarded as the sensor. It is assumed that it has an avionics suite which will provide a pose estimate to localize the energy measurements. The pose estimate,

$$\mathbf{p}_{UAV} = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi, p, q, r]^{\intercal}, \tag{4.24}$$

is a vector with the position and velocity, in the ground reference frame, and the Euler angles and angular rates (sec. 2.3). This estimation is obtained by fusing data from a Global Navigation Satellite System (GNSS), an Inertial Measurement Unit (IMU), and a Pitot pressure probe, which tends to yield low relative error in the short term. Because of that, and because we are not trying to enhance the aircraft position estimate, we will consider that estimate as ground truth for the flow measurements. In addition to the pose, several other aircraft measurements or parameters are used to compute the air flow energy rate.

We will now deduce the measurement equation in terms of the aircraft on-board sensor measurements. We define $\dot{E}_a$ as the aircraft energy rate variation caused by the air flow, i.e., the difference between the total energy rate (3.2) and the predicted nominal energy rate (3.8):

$$\dot{E}_a := -mg\dot{z}_V + mV_a\dot{V}_a - P + \frac{1}{2}\rho V_a^3 S \cdot C_D. \tag{4.25}$$

We may approximate the speed rate of change, by $\dot{V}_a \approx \ddot{x}_B \cos\alpha$, if the rate of change of the wind speed is small, and because the sideslip-angle (β) is assumed null. $\ddot{x}_B$ is the acceleration in the UAV x axis. Now, the measurement of $\dot{E}_a$ may be computed directly from

$$\dot{E}_{a,UAV} = m\left(V_a\ddot{x}_B \cos\alpha - g\dot{z}\right) - P + \frac{1}{2}\rho V_a^3 S \cdot C_D. \tag{4.26}$$

This equation provides the aircraft measurement of the thermal updraft effect. It will be further developed with the *Observation Model* (sec. 4.3.2.7).

4.3.2.1 Particle Filter (PF)

The thermal estimation is implemented through a Particle Filter. The belief distribution at each estimation step is represented by particles with 6 state variables ($\mathbf{x}_t = [x_T, y_T, u_T, v_T]^\intercal, \mathbf{y}_t = [z_i, w^\star]^\intercal$):

$$Bel\left(\mathbf{x}_t, \mathbf{y}_t\right) = P\left(\mathbf{x}_t, \mathbf{y}_t \middle| \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t-1}, \dot{E}_{a1:t}\right), \tag{4.27}$$

where x_T and y_T are the thermal center coordinates, u_T and v_T are the respective velocity components, and z_i and $w^\star$ are the mixing-layer state parameters (sec. 2.5.2.2).

Each particle is a hypothesis of the current state. At each step, particles are propagated, evaluated and resampled, to create a new estimate. The particles are propagated through the propagation model described in section 4.3.2.6. The observation model, also presented in section 4.3.2.7, provides a measurement of likelihood of the hypothesis represented by each particle. The resampling prunes the unlikely particles (hypotheses).

In this implementation, the standard particle filter suffers from degeneracy problem, i.e., the particles converge to a hypothesis that is not the correct one. This happens due to the small number of particles used to represent the initial belief distribution, when compared to the state size.

4.3.2.2 Adaptive Particle Filter (APF)

Thrun et al. [64] present an adaptive version of the Particle Filter (in section 8.3.5). This version is able to cope with the global localization problem. It uses a combination of resampled particles and random particles at each update step. In this work, the method is used to increase the sample diversity when the quality of the estimate decreases. It is assumed that the particles' average likelihood evaluates the estimation quality at each step. This is used as a driver to set the relation between resampled and random particles. The implemented algorithm (Appendix A, lines 10 to 13 and 18) uses the average likelihood to set how many of the particles are resampled or randomly generated. Decreasing likelihoods lead to more random particles. No random particles are generated when the filter converged to a likely estimate.

4.3.2.3 Regularized Particle Filter (RPF)

Ristic et al. [65] describe the Regularized Particle Filter (in section 3.5.3). This method was developed to reduce the degeneracy problem. The RPF does not resample directly from the propagated particle set, which is a discrete approximation of the prior probability distribution. RPF resamples from a continuous approximation of the posterior probability distribution, by actively jittering the resampled particle values:

$$x_t^{[i]\star} = x_t^{[i]} + h_{opt} \cdot \mathbf{D}_t \epsilon_i \tag{4.28}$$

where ϵ_i is a sample from a Gaussian distribution, $\mathbf{D}_t$ is such that $\mathbf{D}_t \mathbf{D}_t^\intercal = \mathbf{S}_t$, $\mathbf{S}_t$ is the empirical covariance matrix, and $h_{opt} = \left[\frac{4}{(n_x+2)N}\right]^{\frac{1}{n_x+4}}$ is the optimal Gaussian kernel bandwidth, function of the state dimension (n_x) and the number of particles N. On the current implementation (Appendix A, lines 15 to 22) the kernel bandwidth was set to $h = k_B \cdot \left[\frac{1}{2N_{RSpl}}\right]^{0.1}$. The tuning gain k_B ($= 0.15$) was needed to guarantee the estimation convergence, as the initial distribution of particles was leading to a divergence of the filter.

4.3.2.4 Regularized Adaptive Particle Filter (RAPF)

The current implementation mixes both Regularized and Adaptive methods (Appendix A). The regularization increases sample diversity, allowing a smaller generation of random particles by the adaptive part. This leads to a stronger convergence, yet enables the filter to avoid wrong convergences, i.e., the convergence of the particle positions to one far from the real position.

4.3.2.5 Particle Generation

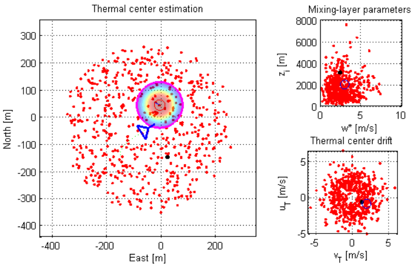


Figure 4.3: Particle generation

When a thermal is first detected, e.g., by the observation of an updraft, the PF is initialized. As the thermal localization is a global localization problem, no initial state can be assumed. As such, a particle generation function was developed (Appendix A, line 23) to generate an initial estimate with random particles. Three rules define the particle generation. The new particles' position (x_T, y_T) distribution is uniform on a bounded radius around the aircraft. The drift velocity (u_T, v_T) distribution is also bounded by the drift speed constraint. The mixing-layer altitude (z_i) and the convective velocity scale $(w^\star)$ are sampled from Gamma distributions, derived from table 2.2 data. Figure 4.3 illustrates the typical particle generation distribution.

4.3.2.6 Propagation Model

The thermal center motion dynamics are described by:

$$\dot{\mathbf{x}}_T = A \cdot \mathbf{x}_T + \varepsilon_T \tag{4.29}$$

where ε_T is the thermal center disturbance vector. The dynamics can be discretized in time, yielding

$$\mathbf{x}_{T,t} = A_t \cdot \mathbf{x}_{T,t-1} + \varepsilon_T \tag{4.30}$$

with

$$A_t = \begin{bmatrix} 1 & 0 & \frac{1}{\Delta t} & 0 \\ 0 & 1 & 0 & \frac{1}{\Delta t} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.31a}$$

$$\varepsilon_T = \begin{bmatrix} 0 \\ 0 \\ \varepsilon_{uT} \\ \varepsilon_{vT} \end{bmatrix} \tag{4.31b}$$

$$\sqrt{\varepsilon_{uT}^2 + \varepsilon_{vT}^2} \sim \mathcal{N}\left(0, \delta_{Drift}^2\right), \tag{4.31c}$$

where ε_T is the thermal center disturbance vector and δ_{Drift} is the drift velocity rate of change.

The mixing-layer dynamics are described by:

$$\dot{\mathbf{y}}_T = \varepsilon_M \tag{4.32}$$

where ε_M is the mixing-layer disturbance vector. The dynamics can be discretized in time, yielding

$$\mathbf{y}_{T,t} = \mathbf{y}_{T,t-1} + \varepsilon_M \tag{4.33}$$

with

$$\varepsilon_M = \left[\varepsilon_{z_i}, \varepsilon_{w*}\right]^\mathsf{T} \tag{4.34a}$$

$$\varepsilon_{z_i} \sim \mathcal{N}\left(0, \delta_{z_i}^2\right) \tag{4.34b}$$

$$\varepsilon_{w*} \sim \mathcal{N}\left(0, \delta_{w*}^2\right), \tag{4.34c}$$

where ε_M is the mixing-layer disturbance vector, δ_{z_i} and δ_{w*} are the mixing-layer altitude and convective velocity scale rates of change, respectively.

4.3.2.7 Observation Model

The aircraft can observe the thermal through its *unexpected* energy variation. The estimate of the aircraft energy variation depends on the engine model, the drag at the current speed and attitude, the aircraft acceleration, and the altitude variation.

The updraft energy rate observation ($\dot{E}_a$) may be described by

$$\dot{E}_a = \dot{E} - \dot{E}_n \approx h\left(\mathbf{x}_T, \mathbf{y}_T, \mathbf{p}_{UAV}\right) + \varepsilon_{\dot{E}}. \tag{4.35}$$

This is derived from (3.2) and (3.8). It follows that,

$$h\left(\mathbf{x}_T, \mathbf{y}_T, \mathbf{p}_{UAV}\right) = -mgW_z - mV_a \begin{bmatrix} \cos\gamma_a \\ -\sin\gamma_a \end{bmatrix}^\mathsf{T} \mathbf{J}_W \begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix}. \tag{4.36}$$

This equation constrains the movement to the XZ plane. The expansion to the general 6DOF case is done by expanding x to the XY plane. Similar to what was discussed in section **??**, the flow field Jacobian matrix is now:

$$\mathbf{J}_W = \begin{bmatrix} \frac{\partial W_x}{\partial x} & \frac{\partial W_x}{\partial y} & \frac{\partial W_x}{\partial z} \\ \frac{\partial W_y}{\partial x} & \frac{\partial W_y}{\partial y} & \frac{\partial W_y}{\partial z} \\ \frac{\partial W_z}{\partial x} & \frac{\partial W_z}{\partial y} & \frac{\partial W_z}{\partial z} \end{bmatrix}. \tag{4.37}$$

So, the expansion of (4.36) yields:

$$h\left(\mathbf{x}_T, \mathbf{y}_T, \mathbf{p}_{UAV}\right) = -mgW_z - mV_a \begin{bmatrix} \cos\psi\cos\gamma_a \\ \sin\psi\cos\gamma_a \\ -\sin\gamma_a \end{bmatrix}^{\mathsf{T}} \mathbf{J}_W \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}. \tag{4.38}$$

The vertical wind velocity is the thermal updraft speed ($W_z = w$) calculated from equations (2.76), (2.77), (2.78), (2.79), and (2.80). To compute this speed we need to define d, the distance to the thermal center:

$$d = \sqrt{\left(x_T - x\right)^2 + \left(y_T - y\right)^2}, \tag{4.39}$$

and the aircraft altitude in the thermal (z). In this case we are considering constant horizontal wind speed, yielding:

$$\mathbf{J}_W = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{\partial W_z}{\partial x} & \frac{\partial W_z}{\partial y} & \frac{\partial W_z}{\partial z} \end{bmatrix}, \tag{4.40}$$

and so

$$h\left(\mathbf{x}_T, \mathbf{y}_T, \mathbf{p}_{UAV}\right) = mV_a \sin\gamma_a \begin{bmatrix} \frac{\partial W_z}{\partial x} \\ \frac{\partial W_z}{\partial y} \\ \frac{\partial W_z}{\partial z} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} - mgW_z. \tag{4.41}$$

The observation uncertainty ($\varepsilon_{\dot{E}}$) has two components. The component due to thermal updraft variability has a similar magnitude to that of h, when the UAV is inside the thermal:

$$\varepsilon_{\dot{E},Thermal} \sim N\left(0, k_U \cdot h\left(\mathbf{x}_T, \mathbf{y}_T, \mathbf{z}\right)\right) \tag{4.42}$$

In addition to this, the uncertainty depends on the aircraft sensing noise and the wind disturbances:

$$\varepsilon_{\dot{E},Sens+Wind} \sim N\left(0, k_{SW}\right). \tag{4.43}$$

As such we can model $\varepsilon_{\dot{E}}$ as:

$$\varepsilon_{\dot{E}} = \varepsilon_{\dot{E},Thermal} + \varepsilon_{\dot{E},Sens+Wind} \sim N\left(0, k_U \cdot h\left(\mathbf{x}_T, \mathbf{y}_T, \mathbf{p}_{UAV}\right) + k_{SW}\right). \tag{4.44}$$

4.3.2.8 Simulation Results

A simulator was built to test the thermal estimator. The simulation includes an Unmanned Aerial Vehicle (UAV) measuring the vertical flow speed. The UAV motion was simulated by a standard unicycle model (sec. 2.4.2). In the simulation the UAV is not commanded to circle inside the thermal, but rather to execute random passes inside the thermal from time to time. This shows that the estimator doesn't require the UAV to be inside the thermal permanently nor any constrained trajectory. The waypoint controller randomly selects waypoints in a grid moving with the center of the simulated thermal. This grid was a square with 100 by 100 meters. The thermal simulation included noise in the drift direction and speed. The mixing-layer parameters (Z_i and w^*) noise was simulated by a Random Walk. Additionally, the updraft speed was also affected by a gain factor, which was set by a Scalar Gauss-Markov Process. No other measurement noise was added.

Table 4.1: Estimators performance

	Localization error	Strength error	Update time
Filter	$(\mu_{err} \pm \sigma_{err})$	$(\mu_{err} \pm \sigma_{err})$	$(\mu_T \pm \sigma_T)$
PF	∞	-	-
APF (600part)	102 ± 16 m	$17,9 \pm 1,9$ W	$0,038 \pm 0,0012$ s
APF (2000part)	40 ± 12 m	$12,7 \pm 1,2$ W	$0,152 \pm 0,0154$ s
RPF (600part)	127 ± 97 m	26 ± 11 W	$0,044 \pm 0,0028$ s
RPF (2000part)	27 ± 17 m	$13 \pm 7,5$ W	$0,184 \pm 0,0096$ s
RAPF(600part)	$14.6 \pm 9,7$ m	$9.7 \pm 3,5$ W	$0,042 \pm 0,0013$ s

Filter	Quality
PF	Converges to a wrong solution
APF (600part)	Produces an unstable solution
APF (2000part)	Produces an unstable solution
RPF (600part)	Sometimes converges to a wrong solution
RPF (2000part)	Converges to a good solution
RAPF(600part)	Converges to a good solution

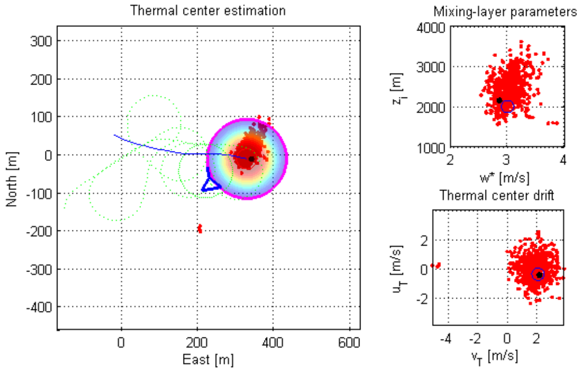


Figure 4.4: RAPF convergence

Table 4.1 compares the performance of the various filters. As indicated the standard Particle Filter (PF) is not a good solution for this problem since, with a reasonable number of particles, it presents unreliable convergence. The Adaptive Particle Filter (APF) is not a good alternative, although its estimates are closer to the true state than the standard Particle Filter (PF) estimates. The average estimation error is close to the average thermal radius, providing poor information. Further, the estimate is very unstable, jumping all around the true state. The Regularized Particle Filter (RPF) is a better solution. With a low quantity of particles (600) it is not very reliable, often suffering from the same problems as the standard PF. With more particles the confidence level on a correct lock-on is also higher, showing good reliability with 2000 particles. The Regularized Adaptive Particle Filter (RAPF) is the best choice. It converges on a good solution and is stable as illustrated in figure 4.4. The regularization enables a good convergence and the adaptive part allows it to identify and exit erroneous convergence processes. The reliability is a lot higher than RPF for the same number of particles. The required computation time is smaller than for any of the simpler Particle Filters (PFs), because fewer particles are needed for a good convergence, as illustrated by the last line of table 4.4. It may also be adjusted to the computation capacity at hand, affecting primarily the convergence time and not the final estimation error.

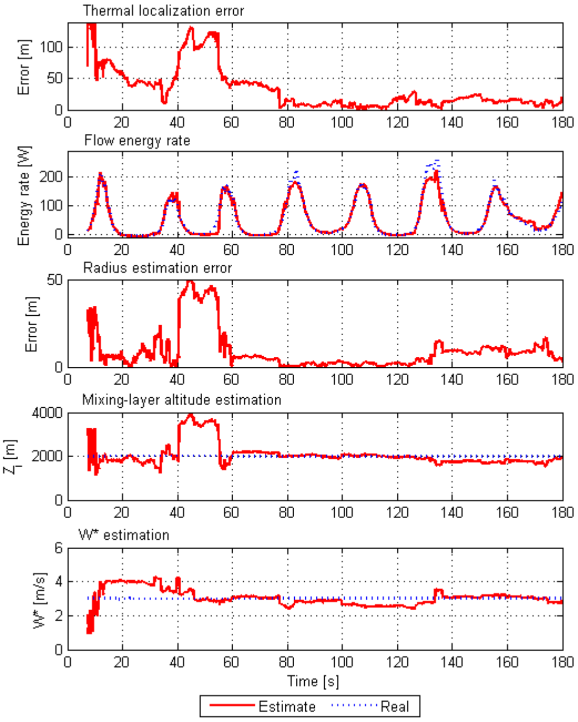


Figure 4.5: RAPF estimation performance

Figure 4.5 shows some performance evaluation plots over a simulated flight. The top plot illustrates an initial convergence of the thermal center position estimate towards the real position. This is followed by strong divergence after which the estimator converges to the correct solution, with a position error below 25 m. The thermal radius and mixing-layer altitude (Z_i) estimates (3^{rd} and 4^{th} plots) also show the same behavior. The convection velocity scale (w^*, last plot) shows a stable convergence throughout the simulation. The energy rate prediction (2^{nd} plot) is quite close to the measured one, providing a good input for a future trajectory controller.

Good estimation results were obtained with a 5Hz update rate, although lower rates may be used with low estimation degradation. Table 4.1 also shows that the RPF and RAPF results were obtained with a computation time quite suitable for real time estimation ($\approx \frac{1}{5}$ of the update time, in MatLab®, on a Intel® Core™2 Duo CPU, T9300 @ 2.5GHz).

4.3.3 Flight Results

The thermal estimator requires the knowledge of the lift, drag, and thrust forces. To estimate these forces we implemented an aircraft parameter identification system. This identification system observes the parameters of the full 6 Degrees-of-Freedom (DOF) model (sec. 2.4.7), using a time domain recursive least-squares observer, described in [40].

The preliminary flight tests showed that the system needs to be improved as the lift, drag, and thrust estimates are too noisy for the thermal estimator. The poor performance is probably due to lack of a sensor for the Angle-of-Attack (AOA). One solution described by Morelli [66] requires the implementation of an extra estimation layer for the air flow angles, i.e., the AOA and the side-slip angle. The solution described is based on a frequency domain observer, which by itself yields less noisy and more accurate results. We intend to improve our aircraft parameter identification system with this solution and then continue with the thermal estimator flight tests.

4.3.4 Bubble Thermal Estimation

To localize and estimate the parameters of *Bubble Thermals* we implemented a similar system to the one described above. The propagation and observation models are quite similar to the ones described for the *Chimney Thermals* estimator (sec. 4.3.2.6 and 4.3.2.7), with some adaptations based on the *Bubble Thermal* model (sec. 2.5.2.5). However, the results were not as good as for the *Chimney Thermals*. The *Bubble Thermal* Particle Filter version requires a lot more particles than the *Chimney Thermal* Particle Filter, for similar results, or even convergent estimation processes. The number of required particles is higher due to the increased complexity of the *Bubble Thermal* model. Therefore, a similar implementation to one for the *Chimney Thermals* is not suitable for real time estimation, as the larger number of particles with more model state variables requires a lot more processing time.

A future solution for the estimation of *Bubble Thermals* is the use of two combined estimators. The nuclear one is a Regularized Adaptive Particle Filter (RAPF) to estimate the 2D thermal updraft parameters. This estimator was tested and yields localization results similar to those of the *Chimney Thermal* RAPF, with slightly more error when the aircraft changes altitude. Its main task would be to provide preliminary information to allow the aircraft to maintain a flight path around the thermal, and also to the more complex 3D estimator. With the data from the 2D estimator, the complete 3D *Bubble Thermal* estimator would run with a slower update rate, using measurements batches, i.e., sets of measurements accumulated between updates. Its output should improve the overall estimation predictions of the *Bubble* vertical motion and shape.

4.4 Wind Shear Estimation

This section deals with the estimation of wind shear phenomena. We focus on vertical wind shear, as surface, inversion, and jet stream shear are quite steady phenomena. Birds learn how to use these phenomena. UAVs may be programmed to do the same. The main requirements are: knowing the wind shear characteristics and controlling the aircraft to execute an energy harvesting flight path. The control methods developed by Lawrance and Sukkarieh [3, 67, 35] require knowledge of the surrounding flow field. These phenomena occur over large areas, which makes it difficult to characterize them as a whole.

Lawrance and Sukkarieh's solution for this characterization is a Gaussian process (GP) regression that describes the flow field over spatially distributed points. The approach we propose is a model-based estimation, requiring few characterization parameters. As such, we simplify the phenomena to uniaxial (z) wind vector variations, i.e., define the wind vector as a function of the altitude only. We further distinguish the vertical wind shear phenomena in Surface and Layer Shear Wind, as the flow gradient is different for each phenomenon.

To estimate the wind shear parameters we developed a Particle Filter. Particle Filters handle nonlinearities well and may be extended to simultaneously localize several different wind shear layers. We describe the adequate propagation and the observation models for Surface and *Layer Wind Shear* phenomena to use with Bayesian filters. These models are based on the wind shear models described in 2.5.1. We then present a Particle Filter estimator for *Surface Wind Shear*, which is similar to the simpler Particle Filter version presented above for thermals estimation. Further, we describe the estimator mechanics and the results obtained, which are quite promising.

As in the case of thermals, the fact that an aircraft is affected by the air flow makes it a potential source of information about the wind shear. The aircraft may estimate the flow field state through its sensor measurements. This estimate is usually obtained through a flow field observer. If we regard the aircraft with its sensors and a flow field observer as a single system, this can now be considered as a sensor for the wind shear phenomena.

In this context, the relevant aircraft measurements are the local wind vector $\mathbf{w} = [W_x, W_y]^\intercal$ and the aircraft position vector $\mathbf{x}_{AC} = [x, y, h]^\intercal$, where h is the altitude above ground level. It is assumed the aircraft can estimate its position with good enough accuracy to take it as ground truth for the wind shear estimator. We prove next that the wind components are observable so that these can be used to estimate the wind shear.

4.4.1 Wind Observability

The wind estimation is essential for wind shear estimators. In this section we prove that it is possible to build a wind observer.

Let us consider the aircraft kinematic equations described in section 2.4.3,

$$\dot{x} = V_a \cos\psi + W_x \tag{4.45a}$$

$$\dot{y} = V_a \sin\psi + W_y \tag{4.45b}$$

$$\dot{\psi} = \omega \tag{4.45c}$$

$$\dot{V}_a = a_x, \tag{4.45d}$$

where W_x and W_y are the wind velocity components, V_a is the aircraft airspeed, ψ is the aircraft heading, and a_x is longitudinal acceleration, i.e, the acceleration over the axis that results from the projection of the aircraft body longitudinal axis (x_B) onto the the horizontal plane (XY_G).

Assumption IV.4. *The aircraft avionics include a GNSS localization system, an IMU, and a Pitot pressure probe, which can output* $\dot{x}$, $\dot{y}$, V_a, ω, *and* a_x.

If we have a heading sensor, such as a magnetometer, the equations in (4.45) are enough to compute W_x and W_y. If there is no explicit heading sensor we have to estimate W_x, W_y, and ψ. Therefore we define the relevant state and the measurement function as:

$$\mathbf{x} = \begin{bmatrix} W_x \\ W_y \\ \psi \end{bmatrix} \tag{4.46a}$$

$$\mathbf{h} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix}. \tag{4.46b}$$

The observability matrix is then:

$$\mathcal{O} = \nabla \begin{bmatrix} \dot{x} \\ \dot{y} \\ \ddot{x} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -V_a \sin\psi \\ 0 & 1 & V_a \cos\psi \\ 0 & 0 & -a_x \sin\psi - V_a\omega \cos\psi \\ 0 & 0 & a_x \cos\psi - V_a\omega \sin\psi \end{bmatrix}, \tag{4.47}$$

and the determinant of $\mathcal{O}^\intercal\mathcal{O}$ is:

$$\det\left(\mathcal{O}^\intercal\mathcal{O}\right) = \det\left(\begin{bmatrix} 1 & 0 & -V_a \sin\psi \\ 0 & 1 & V_a \cos\psi \\ -V_a \sin\psi & V_a \cos\psi & a_x^2 + V_a^2\left(1+\omega^2\right) \end{bmatrix}\right) = a_x^2 + V_a^2\omega^2. \tag{4.48}$$

Theorem IV.5. *The wind components are locally weakly observable by an aircraft as long as its kinematics observe* (4.45)*, and its sensors can measure* $\dot{x}$, $\dot{y}$, V_a, ω, *and* a_x. *Further, the aircraft needs to be turning* ($\omega \neq 0$) *or accelerating forward* ($a_x \neq 0$).

Proof. From (4.48) it is easy to verify that the observability matrix is full rank either if the aircraft is accelerating forward ($a_x \neq 0$), or if it is turning $\omega \neq 0$, as the aircraft must have a minimum velocity ($V_a \geq V_{Stall} > 0$) to fly. □

4.4.2 Wind Shear Propagation Models

We now describe the *Surface* and *Layer Wind Shear* propagation models, used with the Bayesian estimators. The models are based on wind shear models described in 2.5.1. In these models we also define the phenomena state vectors that are used in the observation models.

4.4.2.1 Surface Wind Shear

The *Surface Wind Shear* state vector is $\mathbf{x}_{SWS} = [W_{SWS,6}, \psi_{SWS}, h_{SWS,0}]^\intercal$, where $W_{SWS,6}$ is the wind speed at 6 meters above ground level due to the *Surface Wind Shear*, as specified in the Military Specification MIL-F-8785C [44]. ψ_{SWS} is the *Surface Wind Shear* mean direction, which we assume constant from the ground level to the reference altitude, 6 meters in this case. $h_{SWS,0}$ is *roughness* altitude, which defines the *Surface Wind Shear* gradient shape. The gradient dynamics are described by:

$$\dot{\mathbf{x}}_{SWS} = \varepsilon_{SWS}, \tag{4.49}$$

where ε_{SWS} is a disturbance vector. The dynamics can be discretized in time, yielding

$$\mathbf{x}_{SWS,t} = \mathbf{x}_{SWS,t-1} + \varepsilon_{SWS}, \tag{4.50}$$

with $\varepsilon_{SWS} \sim \mathcal{N}\left(0, \left[\delta^2_{W_{SWS,6}}, \delta^2_{\psi_{SWS}}, \delta^2_{h_{SWS,0}}\right]^\intercal\right)$ and $\delta_{(\cdot)}$ are the horizontal wind velocity, the wind shear direction, and *roughness* altitude rates of change, respectively. These rates of change account for the parameters' time variation and for the spatial variation of the aircraft position.

An alternative propagation model for the *Surface Wind Shear* is one which allows the reference altitude to vary from that of the MIL-F-8785C [44]. The state vector would then be $\mathbf{x}_{SWS} = [h_{SWS,ref}, W_{SWS,ref}, \psi_{SWS}, h_{SWS,0}]^\intercal$, where $h_{SWS,ref}$ is the gradient reference altitude, $W_{SWS,ref}$ is the average horizontal wind velocity at $h_{SWS,ref}$, ψ_{SWS} is the *Surface Wind Shear* main direction, and $h_{SWS,0}$ is *roughness* altitude. The gradient dynamics are now described by:

$$\begin{aligned}
h_{SWS,ref,t} &= h_{SWS,ref,t-1} + \varepsilon_{h_{SWS,ref}} && \text{(4.51a)}\\
W_{SWS,ref,t} &= W_{SWS,ref,t-1}\frac{\ln\left(h_{SWS,ref,t}/h_{SWS,0,t-1}\right)}{\ln\left(h_{SWS,ref,t-1}/h_{SWS,0,t-1}\right)} + \varepsilon_{W_{SWS,ref}} && \text{(4.51b)}\\
\psi_{SWS} &= \psi_{SWS,t-1} + \varepsilon_{\psi_{SWS}} && \text{(4.51c)}\\
h_{SWS,0,t} &= h_{SWS,0,t-1} + \varepsilon_{h_{SWS,0}} && \text{(4.51d)}\\
& && \text{(4.51e)}
\end{aligned}$$

with $\varepsilon_{h_{SWS,ref}} \sim \mathcal{N}\left(0, \delta^2_{h_{SWS,ref}}\right)$, $\varepsilon_{W_{SWS,ref}} \sim \mathcal{N}\left(0, \delta^2_{W_{SWS,ref}}\right)$, $\varepsilon_{\psi_{SWS}} \sim \mathcal{N}\left(0, \delta^2_{\psi_{SWS}}\right)$, and $\varepsilon_{h_{SWS,0}} \sim \mathcal{N}\left(0, \delta^2_{h_{SWS,0}}\right)$. $\delta_{(\cdot)}$ are the *Surface Wind Shear* reference altitude, the horizontal wind velocity, the wind shear direction, and the *roughness* altitude rates of change, respectively.

4.4.2.2 Layer Wind Shear

The *Layer Wind Shear* state vector is $\mathbf{x}_{LWS} = \left[\overline{h}_{LWS}, \Delta h_{LWS}, \overline{\mathbf{w}}_{LWS}, \Delta \mathbf{w}_{LWS}\right]^\intercal$, where $\overline{h}_{LWS}$ is the mean altitude of the wind shear layer, Δh_{LWS} is wind shear layer height, $\overline{\mathbf{w}}_{LWS}$ and $\Delta \mathbf{w}_{LWS}$ are the horizontal wind vector mean and the variation vector over the wind shear layer. As with the *Surface Wind Shear*, this is only a correct representation of the *Layer Wind Shear* if we assume a constant wind gradient direction. The gradient dynamics are described by:

$$\dot{\mathbf{x}}_{LWS} = \varepsilon_{LWS}, \tag{4.52}$$

where ε_{LWS} is a disturbance vector. The dynamics can be discretized in time, yielding

$$\mathbf{x}_{LWS,t} = \mathbf{x}_{LWS,t-1} + \varepsilon_{LWS}, \tag{4.53}$$

with $\varepsilon_{LWS} \sim \mathcal{N}\left(0, \left[\delta^2_{\overline{h}_{LWS}}, \delta^2_{\Delta h_{LWS}}, {\delta^2_{\mu W,LWS}}^\intercal, {\delta^2_{\Delta W_{LWS}}}^\intercal\right]^\intercal\right)$ and $\delta_{(\cdot)}$ are the rates of change of all the *Layer Wind Shear* state parameters, respectively. As with the *Surface Wind Shear* parameters, these rates of change account for the parameters' time variation and the change with spatial variation of the aircraft position.

4.4.3 Wind Shear Observation Models

We now describe the *Surface* and *Layer Wind Shear* observation models, to use with Bayesian estimators. As the propagation models, these are based on wind shear models described in 2.5.1.

The horizontal wind vector observation ($\mathbf{w}_H$) may be described by

$$\mathbf{w}_H = \left[W_x, W_y\right]^\intercal \approx \mathbf{h}\left(\mathbf{x}_{WS}, \mathbf{x}_{AC}\right) + \varepsilon_{W_H}, \tag{4.54}$$

where W_x and W_y are estimated with a wind observer.

The observation uncertainty (ε_{W_H}) is caused mainly by gusts, but also by the aircraft sensing noise, i.e., the wind observer errors. As such, it may be defined as:

$$\varepsilon_{W_H} = \varepsilon_{W_{H,Gust}} + \varepsilon_{W_{H,Sens}} \tag{4.55a}$$

$$\varepsilon_{W_H,Gust} \sim N\left(0, k_{Gust} \cdot \hat{\overline{\mathbf{w}}}_H\right) \tag{4.55b}$$

$$\varepsilon_{W_H,Sens} \sim N\left(0, \sigma_{WObs}\right), \tag{4.55c}$$

where $\hat{\overline{\mathbf{w}}}_H$ is the estimate of the average horizontal wind, σ_{WObs} is the wind observer uncertainty vector, and k_{Gust} is a dimensionless ratio relating the wind variation with the wind average:

$$k_{Gust} = \frac{\sqrt{\mathsf{E}\left[\left(\overline{\|\mathbf{w}_H\|} - \|\mathbf{w}_H\|\right)^2\right]}}{\overline{\|\mathbf{w}_H\|}}. \tag{4.56}$$

4.4.3.1 Surface Wind Shear

The *Surface Wind Shear* observation model is derived from (2.59):

$$\hat{\overline{\mathbf{w}}}_H = \mathbf{h}\left(\mathbf{x}_{SWS}, \mathbf{x}_{AC}\right) = W_{SWS,ref} \frac{\ln\left(h/h_{SWS,0}\right)}{\ln\left(h_{SWS,ref}/h_{SWS,0}\right)} \begin{bmatrix} \cos \psi_{SWS} \\ \sin \psi_{SWS} \end{bmatrix}. \tag{4.57}$$

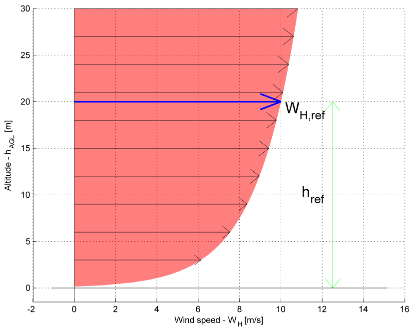


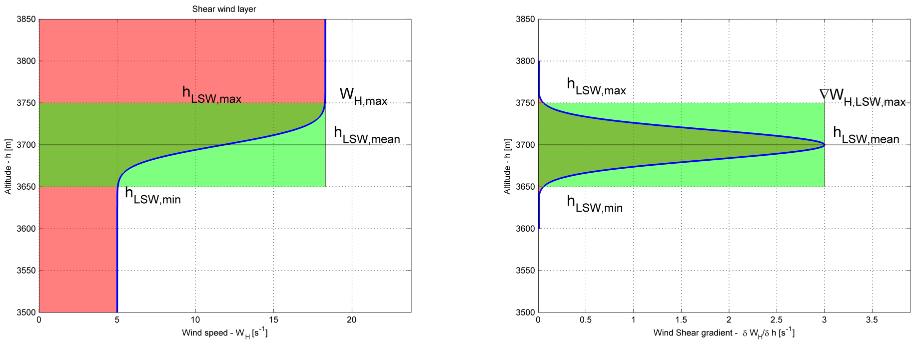
Figure 4.6: *Surface Wind Shear* profile (sec. 2.5.1.1).

4.4.3.2 Layer Wind Shear

The *Layer Wind Shear* observation model is derived from (2.62):

$$\hat{\overline{\mathbf{w}}}_H = \mathbf{h}\left(\mathbf{x}_{LWS}, \mathbf{x}_{AC}\right) = \mathbf{w}_{h_{min},LWS} + \frac{\Delta\mathbf{w}_{LWS}}{2}\left[1 + \operatorname{erf}\left(4\frac{h - \overline{h}_{LWS}}{\Delta h_{LWS}}\right)\right], \qquad (4.58)$$

where $\mathbf{w}_{h_{min},LWS} = \overline{\mathbf{w}}_{LWS} - \frac{\Delta\mathbf{w}_{LWS}}{2}$.



(a) Wind speed profile

(b) Wind speed gradient

Figure 4.7: *Layer Wind Shear* Gaussian model.

4.4.4 Particle Filter (PF)

We now describe the Particle Filter implemented to estimate the *Surface Wind Shear* parameters. The belief distribution at each estimation step is represented by particles defined by the state vector $\mathbf{x}_{SWS}$. Each particle is a hypothesis of the current state. At each step, particles are propagated, evaluated and resampled to create a new estimate. The particles are propagated through the propagation model described in section 4.4.2. This step allows the filter to track the state evolution and represent the uncertainty. The observation model, presented in section 4.4.3, generates the expected observation for each particle. This is combined with the aircraft observation to provide a likelihood measurement of the hypothesis represented by each particle. The resampling prunes the unlikely particles (hypothesis).

4.4.4.1 Particle generation

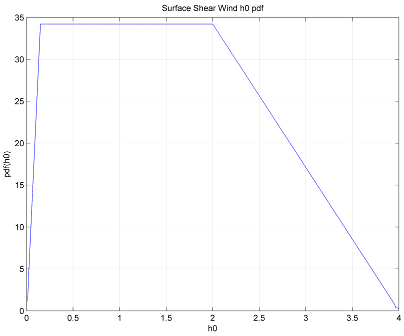


Figure 4.8: *Surface Wind Shear* ”roughness” altitude distribution generated

The estimation particles are generated when the aircraft is low enough to measure the effect of the *Surface Wind Shear*. The current application of Particle Filter assumes a fixed reference altitude as specified in MIL-F-8785C [44]. The generated particles represent the distribution of the ”roughness” altitude as illustrated in figure 4.8. The reference wind vector distribution is centered on the current wind measurement adjusted for the reference altitude (2.59), with a Gaussian distribution. Figure 4.9 illustrates the typical particle generation distribution, in terms of wind vector.

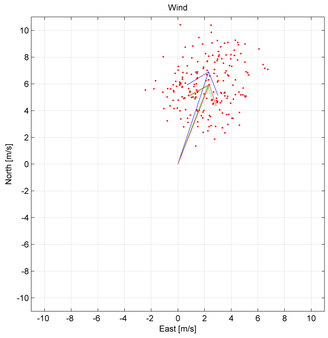


Figure 4.9: Particle generation

4.4.4.2 Simulation Results

The results presented next were obtained through multiple simulations of aircraft flights sweeping the wind shear altitude. The aircraft was simulated by an extended unicycle model, which included altitude variations (sec. 2.4.4). In the simulation the aircraft is commanded to sweep a range of altitudes, including those affected by wind shear phenomena. The wind simulation summed the wind shear effects to gusts. The gusts were generated by a Gauss-Markov process, which simulates well the kind of sensing noise the system would be subject to in reality.

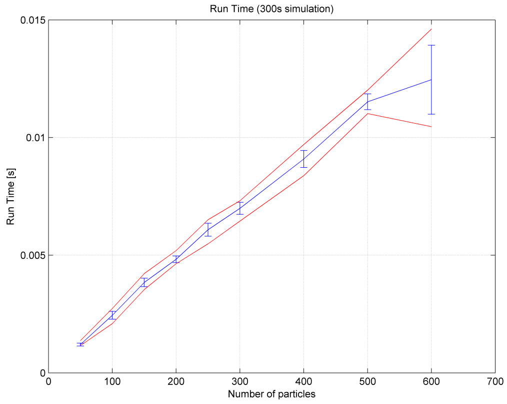
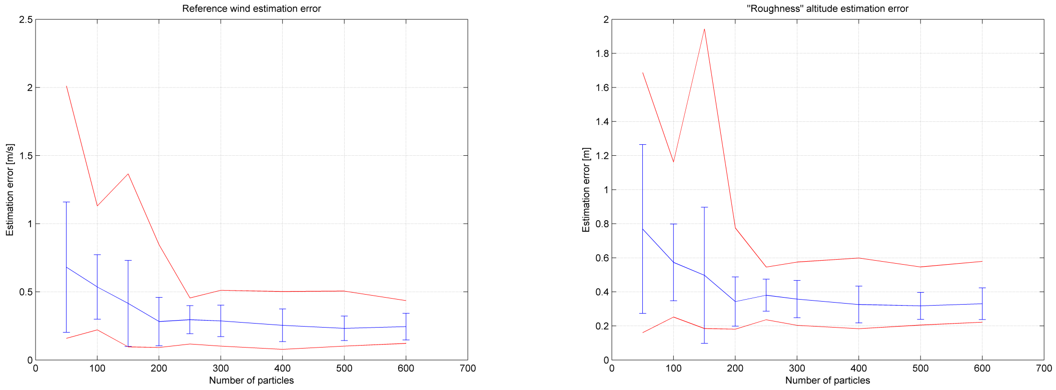


Figure 4.10: *Surface Wind Shear* Particle Filter cycle run time with the number of estimation particles.



(a) Reference wind vector estimation error

(b) ”Roughness” altitude estimation error

Figure 4.11: *Surface Wind Shear* parameters estimation error as a function of the number of particles in the PF. The higher variances and maxima for the test sets with less than 200 particles result from convergence instability presented by the estimator, due to the deficient representation of the probability distributions by the particle sets.

We run the 20 independent simulations with each particle quantity choice to statistically characterize the developed Particle Filters. As expected the time required by the Particle Filter for each estimation cycle grows almost linearly with the number of particles (fig. 4.10). However the estimation quality doesn't improve linearly with the number of particles. In fact, for the *Surface Wind Shear* the estimation quality is almost the same for 200 estimation particles or more (fig. 4.11). In MatLab®, on a Intel® Core™2 Duo CPU, T9300 @ 2.5GHz, a Particle Filter estimation cycle takes less then $5 \times 10^{-3}s$. Further, the Particle Filter doesn't need to run at more than 5Hz to obtain good quality estimations. This means that this Particle Filter with 200 particles can run in real time, with almost no extra load to the processor, and good quality estimates.

The average wind prediction error (fig. 4.12) is mostly under half a meter per second. This error is mostly due to the gusts, as the filter is estimating the effect of the steady wind shear and not the gusts.

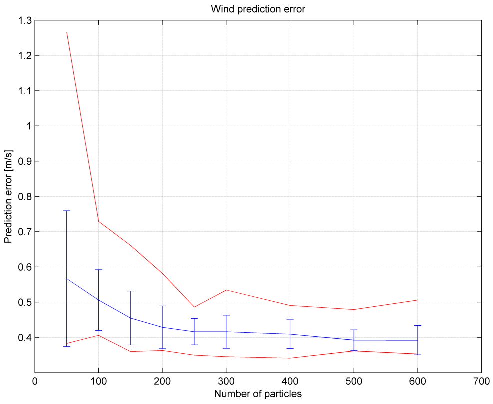


Figure 4.12: Wind prediction error of *Surface Wind Shear* Particle Filter with the number of estimation particles.

4.5 Conclusions

We showed that it is feasible to estimate the relevant parameters of the flow field phenomena, in particular: thermals, wind, and wind shear. We proved that the phenomena parameters are observable and developed estimation methods based on Particle Filters. The Particle Filter estimation methods simulation showed processing times that allow a real time estimation. The computation requirements seem quite suitable for small UAVs with low computational power, as was intended. This is important to these UAVs as it enables them to harvest energy from the observed phenomena.

The *Chimney Thermal* estimator presented in section 4.3.2 is capable of localizing and characterizing atmospheric thermal flows in real time. The global localization nature of the problem and the nonlinear nature of the model pointed to the Particle Filters as the most suited estimators. The implemented Particle Filter uses a detailed thermal updraft model that provides a better position tracking. From the four Particle Filter versions presented and tested, the Regularized Adaptive Particle Filter (RAPF) yielded the best performance, running in real time with very good reliability.

The *Surface Wind Shear* estimator developed in section 4.4.4 also presented very promising performance. We showed that a simple Particle Filter with 200 particles is very well suited for real time estimation of the *Surface Wind Shear* parameters. We were able to estimate and track the *Surface Wind Shear* effects using a Particle Filter with only three parameters. This avoids the need to maintain a time history of the observed flow field or a spatially distributed characterization of it.

CHAPTER V

Control

5.1 Introduction

This chapter describes two control tools developed to support the study of flow fields. We identified the need for airflow data to characterize the flow field in individual thermals, and verify the internal flow field models described in section 2.5.2.1. We have not found any study referring to airflow data collected inside thermals that could be useful for this validation. As mentioned in section 2.5.2.9, the studies we found referring to real data from thermals are from Lenschow and Stephens [36] and from Allen [4], and can't provide any information on the thermal's internal flow field structure.

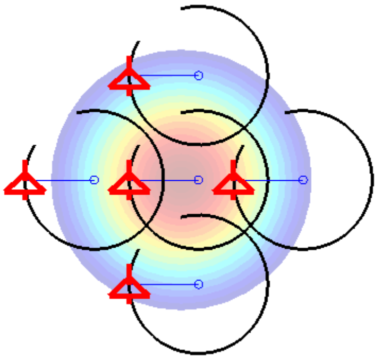


Figure 5.1: Spatially distributed data collection of a thermal updraft airflow.

To acquire useful data to validate the different thermal models we envisioned a system that could observe the flow field in several spatially distributed points simultaneously. We may use several Unmanned Aerial Vehicles (UAVs) to observe the airflow in several points simultaneously. Small UAVs fly slowly enough to maneuver inside individual thermals and to obtain a dense spatial sampling. As we need to characterize the flow field vertically but also horizontally, we need the UAVs to be able to fly safely in close proximity at the same altitude (fig. 5.1).

We study and implement algorithms that control a safe flight in close proximity. We started by studing Collision Avoidance Systems (CASs) and the challenges involved (sec. 5.2). We implemented a CAS control system in an indoor testbed with helicopters and studied how to prove the system safety. With the experience gained with initial CAS implementation, and taking into account that we wanted to have fixed wing UAVs flying at the same altitude, we decided to design a formation flight controller with collision avoidance (sec. 5.3).

5.2 Collision Avoidance System for Close Proximity Operations

5.2.1 Related Work

Airborne aircraft collision avoidance is a current and challenging control topic. Bicchi and Pallottino presented a tutorial at ICRA 2006 with an extensive overview of collision avoidance methods [68]. The Traffic Alert and Collision Avoidance System (TCAS) was created to avoid aircraft collisions, acting only as a last resort system [69]. A common research approach was the use of the optimal control formulation framework which allowed researchers to gain valuable insight into the problem structure and possible solutions [70, 71, 72, 73, 74]. The biggest handicap of these methods is the centralized formulation and its scalability. Jardin presents an optimal control method to guide two aircraft and avoid collisions between them [75]. This method should maintain near optimal paths even in the presence of collision avoidance maneuvers, which disturb the individual aircraft paths. In reference [76], Borrelli et al. describe distributed Receding Horizon Control (RHC) as a feasible solution even for a large number of aircraft. Tomlin et al. [77] studied multiagent conflicts under a hybrid systems framework. They describe a method based on the Hamilton-Jacobi-Isaacs PDE to generate conflict resolution trajectories that are provably safe for two aircraft. This method computes the reachable set of the hybrid system accounting for nonlinear dynamics of the vehicles and the intent uncertainty of one of the aircraft. Hoffmann and Tomlin developed and demonstrated decentralized collision avoidance algorithms for Vertical Takeoff and Landing (VTOL) vehicles [78]. They present one algorithm for two vehicles and another for more vehicles. These algorithms have proven safety and low computational cost. Real tests were performed with STARMAC [79]. Pallottino et al. address the problem of complex hybrid automata verification for decentralized control systems [80]. The authors use probabilistic verification, as classical approaches are difficult to apply. The goal is to prove safety and liveness properties, assuring collision avoidance and goal completion.

Another approach to collision avoidance is the application of SoftWalls. This type of methods was proposed by Edward Lee in response to the September 11, 2001 terrorist attacks with airline aircraft [81]. The goal was to create a flight control system to prevent aircraft from entering no-fly zones. Cataldo presents a comprehensive study on this topic as well as an implementation for “half world” no-fly zones [82]. This method can also be applied to moving objects.

5.2.2 General Problem

The system under analysis [83] is a pair of autonomous helicopters flying on the same horizontal plane, referred to as UAVs from here on. A CAS was implemented to prevent collisions between both vehicles. The CAS controls each UAV through a lower level Flight Control System (FCS). The CAS is composed of a Path Deconflicting Algorithm (PDA) and by a Close Proximity Collision Avoidance Algorithm (CPCAA).

The PDA detects probable conflicting trajectories and generates alternative collision free ones, if needed. The CPCAA detects if the UAVs are too close to each other and forces the separation.

This system can be described mathematically by

$$\dot{\mathcal{C}} = f\left(\mathcal{C}\right) + u\left(\mathcal{C}\right), \tag{5.1}$$

where $\mathcal{C} = \{\mathbf{x}_1, \mathbf{x}_2\} \in \mathbb{R}^4$ is the system state, with $\mathbf{x}_1 = \{x_1, y_1\}$ and $\mathbf{x}_2 = \{x_2, y_2\}$. $f\left(\mathcal{C}\right)$ models the system natural dynamics and $u\left(\mathcal{C}\right)$ introduces the effects of the controls.

The goal of this work was to create and describe a control system which would enable the operation of two vehicles in close proximity while avoiding any collision. The CAS methods are formally described in section 5.2.6. Section 5.2.7 describes the hybrid automata modeling the whole system dynamics. These models differ in the motion and control dynamics. We complete these models with the proofs of safety and liveness in appendix B. These proofs cover some of the dynamic models, such as single integrator or double integrator dynamics, and double integrator dynamics with control delays.

Although the goal of this work is to prove that the CAS avoids any and every collision, history tells us that such a constraint should sometimes be relaxed. The deconfliction method for British bombers in the World War II allowed an average of 1 collision among 1000 bombers during the bombing stage, reducing the average shutdown of aircrafts [84]. As such, we should evaluate very carefully the hard constraints we impose on our systems, "*else we could someday do the equivalent of what the British pilots* [from WWII] *were about to do; avoid accidents and die in the process due to another problem*" [84].

5.2.3 Nomenclature

$\mathbf{x}_i$ - Position of UAV i

$\mathbf{x}_{WPi}$ - Position of current UAV i waypoint

$\Delta\mathbf{x}_{ij}$ - Relative position between the two UAVs $(\mathbf{x}_j - \mathbf{x}_i)$

$\Delta\mathbf{x}_{i\_WP}$ - Relative position between the UAV i and its current waypoint $(\mathbf{x}_{WPi} - \mathbf{x}_i)$

$\Delta\mathbf{x}_{ij}^{i\_WP}$ - Relative position between the two UAVs, in the reference frame attached to $\mathbf{x}_{i\_WP}$

$\Delta\mathbf{x}_{j\_WP}^{i\_WP}$ - Relative position between the UAV j and its current waypoit, in the reference frame attached to $\mathbf{x}_{i\_WP}$

$\mathbf{v}_i$ - Velocity of UAV i

$\mathbf{v}_{cmd,i}$ - Commanded velocity of UAV i

$\Delta\mathbf{v}_{ij}$ - Relative velocity between the two UAVs $(\mathbf{v}_j - \mathbf{v}_i)$

V_{a_i} - Avoidance maneuver speed of UAV i

V_{c_i} - Cruising speed of UAV i

r_{c_i} - Collision radius of UAV i

r_{D_i} - Deconflicting radius of UAV i

r_{s_i} - Safety radius of UAV i

d_{WPi} - Arrival assessment distance (from UAV i to the target waypoint)

5.2.4 Mathematical Background

5.2.4.1 Temporal Logic

Temporal logic is a logical basis for proving system properties [85, 86, 87]. To implement this logic we use immediate and temporal assertions. To assert a state we make use of Propositional Logic (PL) and First-Order Logic (FOL) primitives and add Temporal Logic (TL) primitives.

Immediate assertions: $s_k \models p\left(t\right)$ - p has the value *true* for state s_k at time t: $p\left(t\right)$ *holds for* s_k, or s_k *satisfies* $p\left(t\right)$. As an example, $s_k \models \mathbf{x} = \left(\left[1,\ 0.5\right]^{\mathsf{T}}\right)\left(10\right)$ means that for state s_k the vector $\mathbf{x}$ components are $x_1 = 1$ and $x_2 = 0.5$ at time *10*.

Temporal assertions: $\sigma \models p\left(t\right)$ - p has the value *true* for the string of states σ. Furthermore, $\sigma \models p\left(t\right)\ IFF\ s_0 \models p\left(t\right)$, where s_0 is the initial state of σ.

Propositional logic primitives:

$\sim$ - not

$\wedge$ - and

$\vee$ - or

$\Rightarrow$ - implies

$\Leftrightarrow$ - equivalent

First-order logic primitives:

$\forall$ - for all

$\exists$ - there exists

Temporal logic primitives

$\left(\Box p\right)\left(t_0\right) \Leftrightarrow \forall t \geq t_0, p\left(t\right)$ - "p is true now and forever" (always)

$\left(\Diamond p\right)\left(t_0\right) \Leftrightarrow \exists t \geq t_0, p\left(t\right)$ - "p is true now or in the future" (eventually)

$\left(\circ p\right)\left(t_0\right) \Leftrightarrow p\left(t_0^+\right)$ - "p is true at the next instant"

$\left(\mu q, p\right)\left(t_0\right) \Leftrightarrow \exists t \geq t_0, q\left(t\right) \wedge \forall \tau \in \left[t_0, t\right), p\left(\tau\right)$ - "p is true until q is true"

Derived relations (in reference [85])

1. $\Box$ and $\Diamond$ are duals of each other by

$$\Diamond P \equiv \sim \Box \sim P \tag{5.2}$$

2. $(p \rightarrow q)(t) \equiv (\Box (p \Rightarrow \Diamond q))(t)$ - $p(t)$ leads to $q(t)$
3. If $s_k \models p(t), \forall s_k \in \mathcal{Q}$ and $\mathcal{Q}$ is the state space of the system S, then $S \models p(t)$
4. $S \models p(t) \Rightarrow S \models (\Box p)(t)$
5. $(\Box (p \Rightarrow q))(t) \Rightarrow (p \rightarrow q)(t)$
6. $(\Box (p \wedge q))(t) \equiv (\Box p \wedge \Box q)(t)$
7. $(\Diamond (p \vee q))(t) \equiv (\Diamond p \vee \Diamond q)(t)$
8. $(\Box p \vee \Box q)(t) \Rightarrow (\Box (p \vee q))(t)$
9. $(\Box p \wedge \Box (p \Rightarrow q))(t) \Rightarrow (\Box q)(t)$
10. $(\Diamond p)(t) \vee (\Box \sim p)(t)$
11. $((p \rightarrow q)(t) \wedge (q \rightarrow r)(t)) \Rightarrow (p \rightarrow r)(t)$
12. $((p \rightarrow r)(t) \wedge (q \rightarrow r)(t)) \Rightarrow ((p \vee q) \rightarrow r)(t)$
13. $((p \wedge \Box q) \rightarrow r)(t) \Rightarrow ((p \wedge \Box q) \rightarrow (r \wedge \Box q))(t)$

5.2.4.2 Other Definitions

Definition V.1 (Internal Product)**.** The internal product of any two vectors $\mathbf{u} \in \mathbb{R}^n$ and $\mathbf{v} \in \mathbb{R}^n$ is represented here by $\langle \mathbf{u} | \mathbf{v} \rangle$ and is defined as:

$$\langle \mathbf{u} | \mathbf{v} \rangle = \mathbf{u}^{\mathrm{T}} \cdot \mathbf{v} = \sum_{i=1}^{n} u_i \cdot v_i \tag{5.3}$$

Definition V.2 (Parallel Projection)**.** The parallel projection of vector $\mathbf{v} \in \mathbb{R}^n$ onto a vector $\mathbf{u} \in \mathbb{R}^n$ is represented here by $\langle \mathbf{u} \| \mathbf{v} \rangle$ and is defined as:

$$\langle \mathbf{u} \| \mathbf{v} \rangle = \langle \hat{\mathbf{u}} | \mathbf{v} \rangle = \frac{\langle \mathbf{u} | \mathbf{v} \rangle}{\|\mathbf{u}\|} = \frac{\langle \mathbf{u} | \mathbf{v} \rangle}{\sqrt{\langle \mathbf{u} | \mathbf{u} \rangle}} \tag{5.4}$$

Definition V.3 (Perpendicular Projection)**.** The perpendicular projection of vector $\mathbf{v} \in \mathbb{R}^2$ onto a vector $\mathbf{u} \in \mathbb{R}^2$ is represented here by $\langle \mathbf{u} \perp \mathbf{v} \rangle$ and is defined as:

$$\langle \mathbf{u} \perp \mathbf{v} \rangle = \left\langle \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{u} \middle\| \mathbf{v} \right\rangle \tag{5.5}$$

The perpendicular projection is a parallel projection applied to a vector rotated 90 degrees from the original.

Definition V.4 (Absolute Perpendicular Projection)**.** The absolute perpendicular projection of vector $\mathbf{v} \in \mathbb{R}^2$ onto a vector $\mathbf{u} \in \mathbb{R}^2$ is represented here by $|\mathbf{u} \perp \mathbf{v}|$ and is defined as:

$$|\mathbf{u} \perp \mathbf{v}| = |\langle \mathbf{u} \perp \mathbf{v} \rangle| \tag{5.6}$$

Definition V.5. Property is something that is proven true for every possible execution.

Definition V.6 (Safety)**.** The safety property ensures that nothing "bad" happens to the system ([88]), meaning that the system never enters an unacceptable state. On the current case the system has the safety property if the distance between vehicles is always larger than the collision distance,

$$(\square \, \|\Delta \mathbf{x}_{12}\| > d_{c_{12}}) \, (t) \tag{5.7}$$

Definition V.7 (Liveness)**.** The liveness property ensures that something "good" eventually happens to the system ([85, 88]), which means that the system enters a desirable state in a finite time horizon. Liveness is proved for the current case if the UAV gets within a certain distance from the target waypoint,

$$(\lozenge \, \|\Delta \mathbf{x}_{i\_WP}\| \leq d_{WPi}) \, (t) \tag{5.8}$$

5.2.5 System Definitions

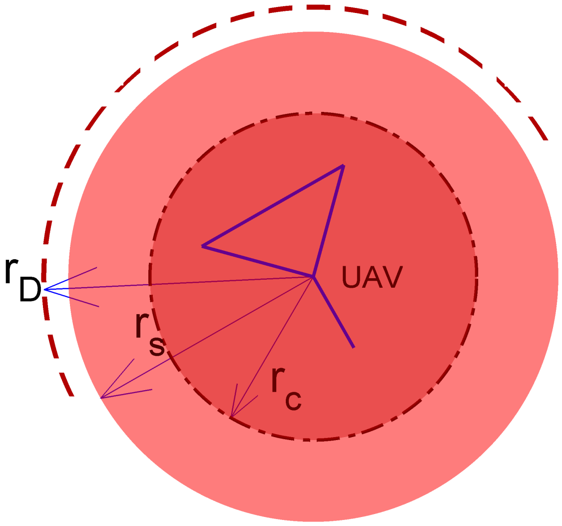


Figure 5.2: Radial layers

Definition V.8 (Collision radius)**.** r_c is the largest horizontal distance from any component of the UAV to its Center of Mass (CM) $\mathbf{x}$ (fig. 5.2). That means that a collision can occur only if

$$\|\Delta \mathbf{x}_{ij}\| \leq d_{c_{ij}} = r_{c_i} + r_{c_j}, \tag{5.9}$$

Definition V.9 (Safety radius)**.** r_s is the distance from an UAV center to any other object (fig. 5.2) above which it is considered a safe operation state. Therefore, r_s should be such that

$$r_{s_i} =: r_{c_i} \cdot a_i,\ a_i > 1, a_i \in \mathbb{R}, \tag{5.10}$$

and so the safety distance between UAVs is: $d_{s_{ij}} = r_{s_i} + r_{s_j}$.

Definition V.10 (Deconflition radius)**.** r_D sets the arc at which any other UAV will try to pass tangently, if the deconfliction is required (fig. 5.2). r_D should be such that

$$r_{D_i} =: r_{s_i} \cdot b_i,\ b_i \geq 1, b_i \in \mathbb{R}, \tag{5.11}$$

and so: $d_{D_{ij}} = r_{D_i} + r_{D_j}$.

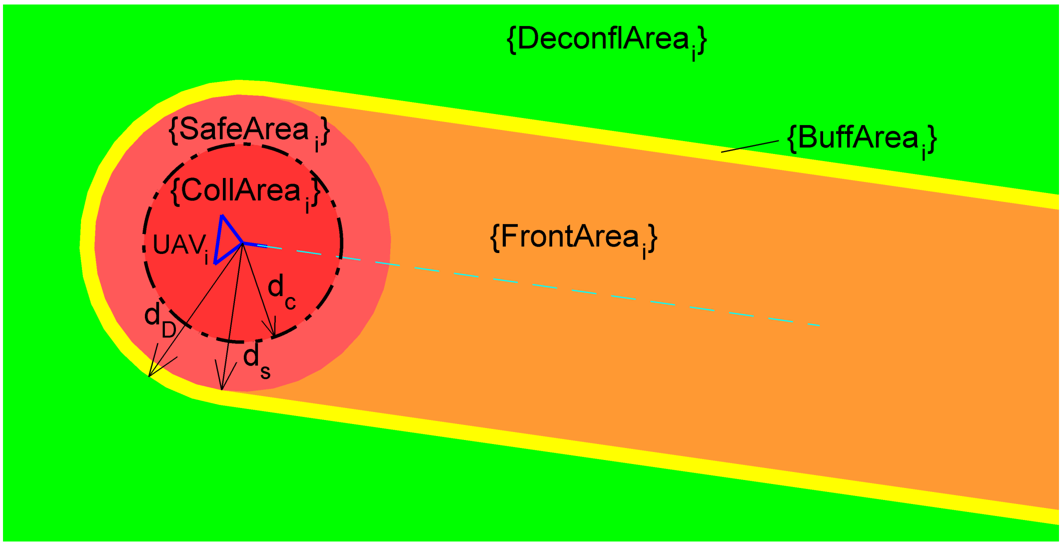


Figure 5.3: Operating areas

Definition V.11 (Collision Area)**.** As stated before a collision can occur if $\|\Delta\mathbf{x}_{ij}\| \leq d_{c_{ij}}$ (fig. 5.3). Therefore, the definition of the UAV i *Collision Area* (relative to UAV j) is:

$$CollArea_{ij} := \left\{\mathbf{x} : \|\mathbf{x} - \mathbf{x}_i\| \leq d_{c_{ij}}, \mathbf{x} \in \mathbb{R}^2\right\}. \tag{5.12}$$

Definition V.12 (Safety area)**.** The UAVs enter in each other's *Safety Area* if $\|\Delta\mathbf{x}_{ij}\| \leq d_{s_{ij}}$(fig. 5.3). Therefore, the definition of the UAV i *Safety Area* (relative to UAV j) is:

$$SafeArea_{ij} := \left\{\mathbf{x} : d_{c_{ij}} < \|\mathbf{x} - \mathbf{x}_i\| \leq d_{s_{ij}}, \mathbf{x} \in \mathbb{R}^2\right\}. \tag{5.13}$$

Definition V.13 (Frontal Area)**.** The *Frontal Area* of UAV i consists of all points in front of the UAV, within a distance $d_{s_{ij}}$ of the line connecting its CM to the waypoint, and outside the *Safety Area* (fig. 5.3). Therefore, the definition of the UAV i *Frontal Area* (relative to UAV j) is:

$$\begin{aligned} FrontArea_{ij} := \{\mathbf{x} : \langle \mathbf{x}_{WP_i} - \Delta\mathbf{x}_{ij} \parallel \mathbf{x} - \Delta\mathbf{x}_{ij}\rangle > 0 \wedge \ldots \\ \ldots \wedge |\mathbf{x}_{WP_i} - \Delta\mathbf{x}_{ij} \perp \mathbf{x} - \Delta\mathbf{x}_{ij}| \leq d_{s_{ij}} \wedge \|\mathbf{x} - \mathbf{x}_i\| > d_{s_{ij}}, \mathbf{x} \in \mathbb{R}^2\right\}. \end{aligned} \tag{5.14}$$

Definition V.14 (Buffer Area)**.** The *Buffer Area* of UAV i consists of all points within a distance $d_{D_{ij}}$ of its CM or the line connecting its CM to the waypoint, and outside the safety and the *Frontal Area* (fig. 5.3). Therefore, the definition of the UAV i *Buffer Area* (relative to UAV j) is:

$$\begin{aligned} &BuffArea_{ij} := \big\{\mathbf{x} : \|\mathbf{x}-\mathbf{x}_i\| < d_{D_{ij}} \vee ... \\ &... \vee \big(|\Delta\mathbf{x}_{i\_WP} \perp \mathbf{x}-\mathbf{x}_i| < d_{D_{ij}} \wedge \langle \Delta\mathbf{x}_{i\_WP} \, \|\mathbf{x}-\mathbf{x}_i\rangle > 0\big)\, , \, ... \\ &... \, \mathbf{x} \in \mathbb{R}^2 \backslash \left\{CollArea_{ij} \vee SafeArea_{ij} \vee FrontArea_{ij}\right\}\big\}\, , \end{aligned} \tag{5.15}$$

or

$$\begin{aligned} &BuffArea_{ij} := \big\{\mathbf{x} : \big(d_{s_{ij}} < \|\mathbf{x}-\mathbf{x}_i\| < d_{D_{ij}} \wedge \langle \Delta\mathbf{x}_{i\_WP} \, \|\mathbf{x}-\mathbf{x}_i\rangle < 0\big) \vee ... \\ &... \vee \big(d_{s_{ij}} < |\Delta\mathbf{x}_{i\_WP} \perp \mathbf{x}-\mathbf{x}_i| < d_{D_{ij}} \wedge \langle \Delta\mathbf{x}_{i\_WP} \, \|\mathbf{x}-\mathbf{x}_i\rangle > 0\big)\, , \mathbf{x} \in \mathbb{R}^2\big\}\, . \end{aligned} \tag{5.16a}$$

Definition V.15 (Deconfliction Area)**.** The *Deconfliction Area* of UAV i consists of all points outside $CollArea_{ij}$, $SafeArea_{ij}$, $FrontArea_{ij}$ and $BuffArea_{ij}$ (fig. 5.3). Therefore, the definition of the UAV i *Deconfliction Area* (relative to UAV j) is:

$$\begin{aligned} &DeconflArea_{ij} := \big\{\mathbf{x} : \mathbf{x} \in \mathbb{R}^2 \backslash \left\{CollArea_{ij} \vee SafeArea_{ij} \vee ... \right. \\ &\qquad ... \left. \vee FrontArea_{ij} \vee BuffArea_{ij}\right\}\big\} \end{aligned} \tag{5.17}$$

or

$$\begin{aligned} &DeconflArea_{ij} := \big\{\mathbf{x} : \big(\|\mathbf{x}-\mathbf{x}_i\| > d_{D_{ij}} \wedge \langle \Delta\mathbf{x}_{i\_WP} \, \|\mathbf{x}-\mathbf{x}_i\rangle < 0\big) \vee ... \\ &\qquad ... \vee |\Delta\mathbf{x}_{i\_WP} \perp \mathbf{x}-\mathbf{x}_i| > d_{D_{ij}}, \mathbf{x} \in \mathbb{R}^2\big\}\, . \end{aligned} \tag{5.18}$$

Definition V.16 (Conflict Area)**.** The *Conflict Area* of UAV i consists of all points within a distance $d_{D_{ij}}$ of its CM, its current waypoint, and the line connecting its CM to the waypoint (fig. 5.9):

$$\begin{aligned} &ConflArea_{ij} := \big\{\mathbf{x} : \|\mathbf{x}-\mathbf{x}_i\| \le d_{D_{ij}} \vee \|\mathbf{x}-\mathbf{x}_{WPi}\| \le d_{D_{ij}} \vee ... \\ &\qquad ... \vee \big(|\Delta\mathbf{x}_{i\_WP} \perp \mathbf{x}-\mathbf{x}_i| \le d_{D_{ij}} \wedge ... \\ &\qquad ... \wedge 0 < \left\langle \Delta\mathbf{x}_{i\_WP} \left\| \frac{\mathbf{x}-\mathbf{x}_i}{\|\Delta\mathbf{x}_{i\_WP}\|} \right.\right\rangle < 1\big)\, , \mathbf{x} \in \mathbb{R}^2\Big\}\, . \end{aligned} \tag{5.19}$$

Definition V.17 (Close Frontal Area)**.** The *Close Frontal Area* of UAV i consists of all points within the triangle formed by the points $\mathbf{x}_{CF}$, $\mathbf{y}_{CF1}$, $\mathbf{y}_{CF2}$ from figure 5.4 and out of the *Safety Area.* The definition of the UAV i *Close Frontal Area* (relative to UAV j) is:

$$\begin{aligned} &CloseFArea_{ij} := \{\mathbf{x} : \langle \mathbf{x}_{i\_CF} - \mathbf{y}_{i\_CF1} \perp \mathbf{x} - (\mathbf{x}_i + \mathbf{y}_{i\_CF1})\rangle \ge 0 \wedge ... \\ &... \wedge \langle \mathbf{x}_{i\_CF} - \mathbf{y}_{i\_CF2} \perp \mathbf{x} - (\mathbf{x}_i + \mathbf{y}_{i\_CF2})\rangle \le 0 \wedge ... \\ &... \wedge \langle \mathbf{y}_{i\_CF2} - \mathbf{y}_{i\_CF1} \perp \mathbf{x} - (\mathbf{x}_i + \mathbf{y}_{i\_CF1})\rangle < 0 \wedge \|\mathbf{x}-\mathbf{x}_i\| > d_{s_{ij}}, \mathbf{x} \in \mathbb{R}^2\big\}\, , \end{aligned} \tag{5.20}$$

where,

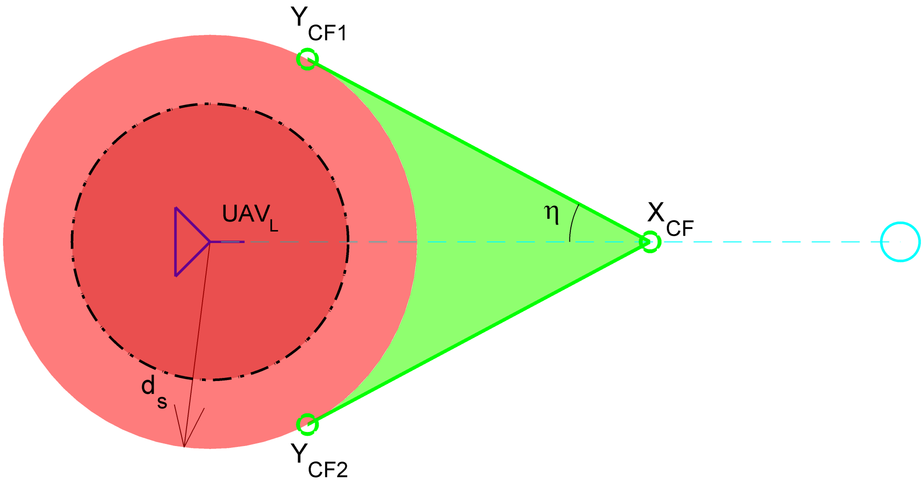


Figure 5.4: Close Frontal Area

$$\mathbf{x}_{i\_CF}^{i\_WP} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \frac{d_{s_{ij}}}{\sin\eta_{ij}} \Rightarrow \mathbf{x}_{i\_CF} = \frac{\Delta\mathbf{x}_{i\_WP}}{\|\Delta\mathbf{x}_{i\_WP}\|} \frac{d_{s_{ij}}}{\sin\eta_{ij}} \tag{5.21a}$$

$$\mathbf{y}_{i\_CF1}^{i\_WP} = \begin{bmatrix} \sin\eta_{ij} \\ -\cos\eta_{ij} \end{bmatrix} d_{s_{ij}} \Rightarrow \mathbf{y}_{i\_CF1} = \mathbf{R}_{i\_WP} \cdot \mathbf{y}_{i\_CF1}^{i\_WP} \tag{5.21b}$$

$$\mathbf{y}_{i\_CF2} = \begin{bmatrix} \sin\eta_{ij} \\ \cos\eta_{ij} \end{bmatrix} d_{s_{ij}} \Rightarrow \mathbf{y}_{i\_CF2} = \mathbf{R}_{i\_WP} \cdot \mathbf{y}_{i\_CF2}^{i\_WP} \tag{5.21c}$$

$$\eta_{ij} = \arctan\left(\frac{V_{c_j}}{V_{c_i}}\right) \tag{5.21d}$$

This area is important, because if the PDA method is active and the follower UAV for some reason falls in the leader's *Close Frontal Area*, the UAVs behavior would lead them to each other's *Safety Areas*, which is not desirable:

$$PDA \wedge \mathbf{x}_F \in CloseFArea_{LF} \Rightarrow \left(\Diamond \|\Delta\mathbf{x}_{12}\| \leq d_{s_{12}}\right)(t)\,, \tag{5.22}$$

and so the CPCAA should be activated.

Definition V.18 (Avoidance Area)**.** The *Avoidance Area* of UAV i consists of all points included in the *Safety* and *Close Frontal Areas*:

$$AvoidArea_{ij} := \{\mathbf{x} : \mathbf{x} \in SafeArea_{ij} \vee CloseFArea_{ij}\}\,. \tag{5.23}$$

5.2.6 Collision Avoidance Control

To solve the problem at hands, we created a collision avoidance system (CAS) with a hybrid control automaton. Figure 5.5 shows a simplified illustration of this automaton. The natural state is the *WP tracking*, where each vehicle tracks its own waypoint. If a future conflict is predicted, the *Path deconfliction* state is activated, giving right-of-way to one of the vehicles and commanding the other to avoid the first, going around it. If the vehicles are too close to each other the system switches to the

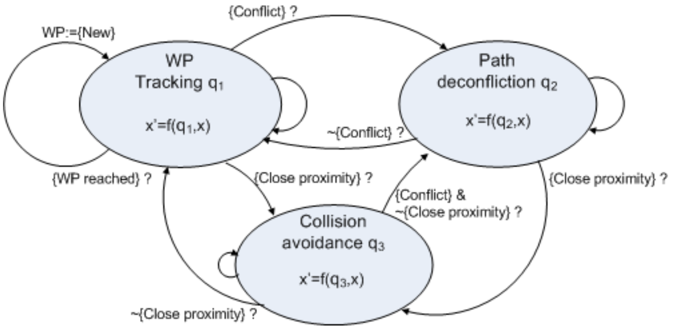


Figure 5.5: Hybrid automaton

Collision avoidance state. This state has priority over all other states, to avoid any collision.

Definition V.19 (CPCAA)**.** The *Close Proximity Collision Avoidance Algorithm* is activated if the UAVs are in each other's *Safety Area* (fig. 5.6):

$$\|\Delta\mathbf{x}_{12}\| \leq d_{s_{12}} \Rightarrow \{active\, CPCAA\}\,. \tag{5.24}$$

Further, it is activated if one UAV is in the other's *Close Frontal Area*, to keep the UAVs from reaching each other's *Safety Areas*:

$$\mathbf{x}_i \in CloseFArea_{ij}, i \neq j, i, j = 1, 2 \Rightarrow \{active\, CPCAA\}\,. \tag{5.25}$$

This means that the *CPCAA* is activated whenever one UAV is in the other's *Avoidance Area*:

$$\mathbf{x}_i \in AvoidArea_{ij}, i \neq j, i, j = 1, 2 \Rightarrow \{active\, CPCAA\}\,. \tag{5.26}$$

CPCAA forces the UAVs' commands to be

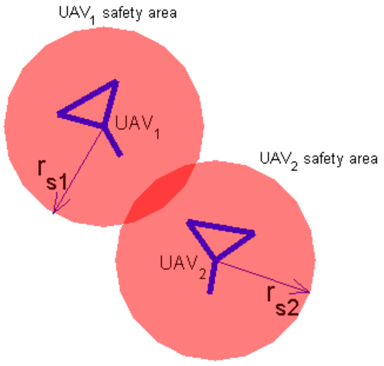


Figure 5.6: Close proximity detection

$$\{active\, CPCAA\} \Rightarrow \begin{cases} \mathbf{v}_{cmd_1} = -\frac{\Delta\mathbf{x}_{12}}{\|\Delta\mathbf{x}_{12}\|} \cdot V_{a_1} \\ \mathbf{v}_{cmd_2} = -\frac{\mathbf{x}_{21}}{\|\mathbf{x}_{21}\|} \cdot V_{a_2} = \frac{\Delta\mathbf{x}_{12}}{\|\Delta\mathbf{x}_{12}\|} \cdot V_{a_2} \end{cases}, \tag{5.27}$$

as shown in figure 5.7.

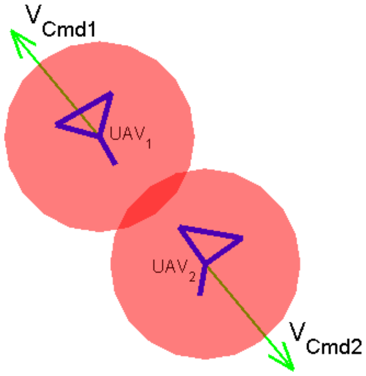


Figure 5.7: Collision avoidance commands

Definition V.20 (PDA)**.** The *Path Deconfliction Algorithm* should maintain a separation distance larger than $d_{D_{12}}$ between the UAVs. This algorithm was introduced to create smoother and more efficient paths for conflict resolution. The situations detected by the PDA are the UAVs' predicted path interceptions (fig. 5.8) or the intrusion of one UAV in the other's *Conflict Area* (fig. 5.9). An intersection detection occurs when the two lines drawn from each UAV's *virtual point* to its *virtual waypoint* intercept. The *virtual points* extend the detected interception range, allowing the detection of possible conflicting paths, related with the area occupied by the vehicle, which otherwise would not be detected. The *virtual points* are defined by:

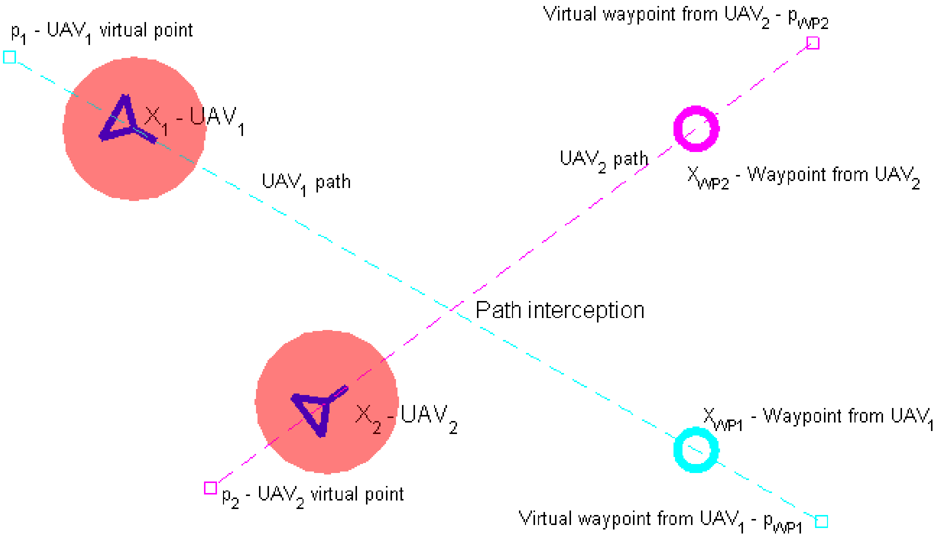


Figure 5.8: UAVs' path interception

$$\begin{cases} \mathbf{p}_i := \mathbf{x}_i - \frac{\Delta\mathbf{x}_{i\_WP}}{\|\Delta\mathbf{x}_{i\_WP}\|} \cdot d_{D_{12}} \\ \mathbf{p}_{WPi} := \mathbf{x}_{WPi} + \frac{\Delta\mathbf{x}_{i\_WP}}{\|\Delta\mathbf{x}_{i\_WP}\|} \cdot d_{D_{12}} \end{cases}, \ i = 1, 2. \tag{5.28}$$

The formal definition for the interception detection is:

$$
\begin{aligned}
&Interception \Leftarrow\sim \{(\langle \Delta\mathbf{x}_{1\_WP} | \Delta\mathbf{x}_{2\_WP} \rangle = 0 \wedge \sim \langle \Delta\mathbf{x}_{1\_WP} \perp \Delta\mathbf{x}_{12} \rangle = 0) \vee ..\\
&... \vee sign\left(\langle \Delta\mathbf{x}_{1\_WP} \perp \mathbf{p}_2 - \mathbf{p}_1 \rangle\right) = sign\left(\langle \Delta\mathbf{x}_{1\_WP} \perp \mathbf{p}_{WP2} - \mathbf{p}_1 \rangle\right) \vee ...\\
&... \vee sign\left(\langle \Delta\mathbf{x}_{2\_WP} \perp \mathbf{p}_1 - \mathbf{p}_2 \rangle\right) = sign\left(\langle \Delta\mathbf{x}_{2\_WP} \perp \mathbf{p}_{WP1} - \mathbf{p}_2 \rangle\right)\} .
\end{aligned}
\tag{5.29}
$$

The first verification excludes all cases where the paths are parallel and not coincident. The last two verifications detect the existing interceptions. When a path interception is detected the UAV closest to the interception will be designated the leader (UAV_L) with right-of-way, not changing its course. The other UAV is designated the follower (UAV_F). The distance from each UAV to the interception point is calculated by:

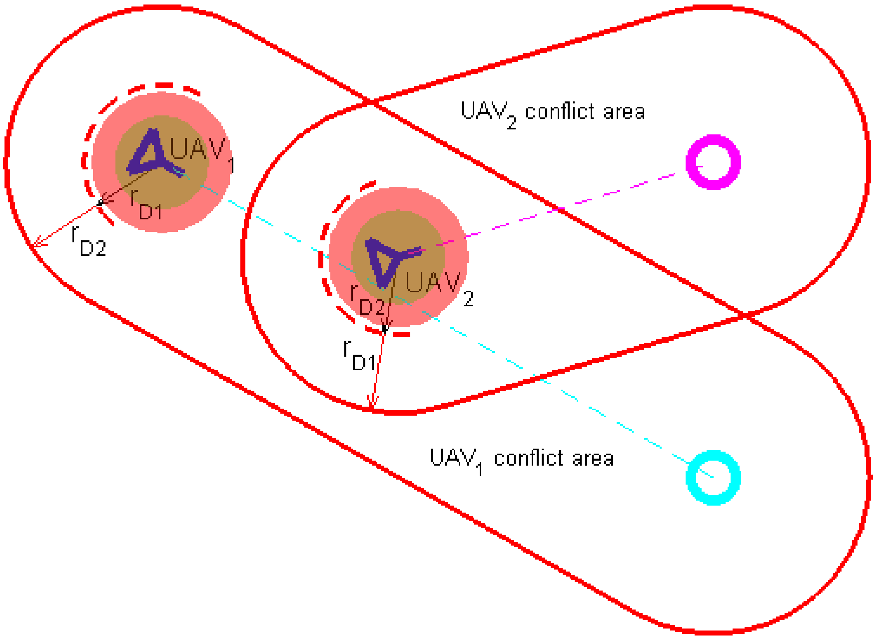


Figure 5.9: UAVs' *Conflict areas*

$$
d_{1Int} = x_{12}^{1\_WP} + y_{12}^{1\_WP} \left(- \frac{x_{2\_WP}^{1\_WP}}{y_{2\_WP}^{1\_WP}} \right) \tag{5.30a}
$$

$$
d_{2Int} = y_{12}^{1\_WP} \sqrt{1 + \left(\frac{x_{2\_WP}^{1\_WP}}{y_{2\_WP}^{1\_WP}} \right)^2} , \tag{5.30b}
$$

where

$$
\Delta\mathbf{x}_{12}^{1\_WP} = \mathbf{R}_{1\_WP} \cdot \Delta\mathbf{x}_{12} \tag{5.31a}
$$

$$
\Delta\mathbf{x}_{2\_WP}^{1\_WP} = \mathbf{R}_{1\_WP} \cdot \Delta\mathbf{x}_{2\_WP} \tag{5.31b}
$$

$$
\mathbf{R}_{1\_WP} = \frac{1}{\|\Delta\mathbf{x}_{1\_WP}\|} \begin{bmatrix} x_{1\_WP} & y_{1\_WP} \\ -y_{1\_WP} & x_{1\_WP} \end{bmatrix} , \tag{5.31c}
$$

That being, the role attribution function is:

$$L = \underset{i=1,2}{\arg\min} d_{iInt} \tag{5.32a}$$

$$F = i : i = 1, 2, i \neq L, \tag{5.32b}$$

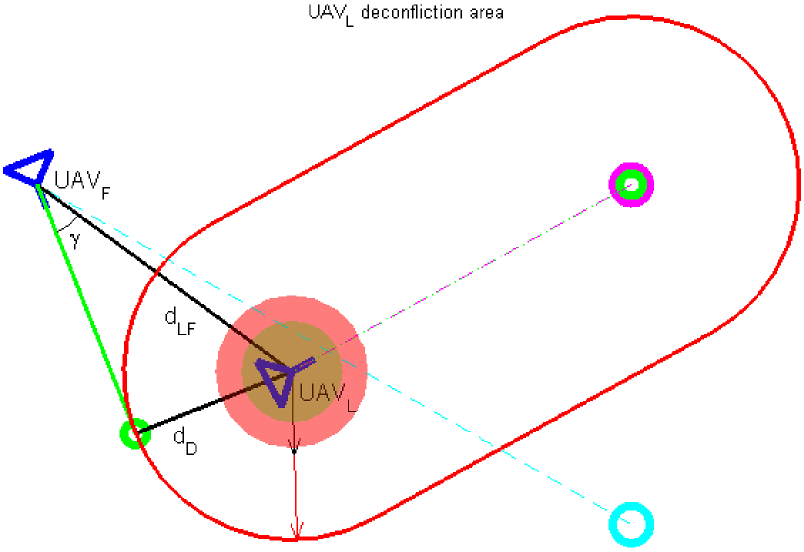


Figure 5.10: Basic deconfliction path

The follower will target the leader's rear tangent arc until the desired path interception ceases to exist, if it is in the leader's *deconfliction* or *Buffer Area*. If the follower is in the leader's *Frontal Area*, it will move away from the leader's path, perpendicularly to its course. If the follower UAV is in the leader's *deconfliction area*, the deconflicting velocity command is defined by:

$$\mathbf{v}_{ref_F} = \begin{bmatrix} \cos\gamma & \kappa\sin\gamma \\ -\kappa\sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F},\ \kappa = -1, 1. \tag{5.33}$$

where $\sin\gamma = \frac{d_D}{d_{ij}}$. An option, represented by κ, is selected as a function of the position of the tangency points ($\mathbf{x}_{tgWP}$):

$$\mathbf{x}_{tgWP} = \begin{bmatrix} -\sin\gamma & \kappa\cos\gamma \\ -\kappa\cos\gamma & -\sin\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot d_D. \tag{5.34}$$

These points have to be in the rear quadrant of the leader, and so

$$\begin{aligned} \langle \Delta\mathbf{x}_{L\_WP} | \mathbf{x}_{tgWP} \rangle \leq 0 \Leftrightarrow \left\langle \Delta\mathbf{x}_{L\_WP} \middle| \begin{bmatrix} -\sin\gamma & \kappa\cos\gamma \\ -\kappa\cos\gamma & -\sin\gamma \end{bmatrix} \mathbf{x}_{FL} \right\rangle \leq 0 \Leftrightarrow \\ \Leftrightarrow \left\langle \Delta\mathbf{x}_{L\_WP} \middle| \begin{bmatrix} -\frac{d_D}{d_{12}} & \kappa\sqrt{1-\left(\frac{d_D}{d_{12}}\right)^2} \\ -\kappa\sqrt{1-\left(\frac{d_D}{d_{12}}\right)^2} & -\frac{d_D}{d_{12}} \end{bmatrix} \mathbf{x}_{FL} \right\rangle \leq 0. \end{aligned} \tag{5.35}$$

If both options are valid, the one presenting the maximum $\left\langle \Delta\mathbf{x}_{F\_WP} \middle| \mathbf{v}_{ref_F} \right\rangle$ is chosen:

$$Opt = \underset{\kappa=-1,1}{\arg\max} \left\langle \Delta\mathbf{x}_{F\_WP} \middle| \mathbf{v}_{ref_F} \right\rangle . \tag{5.36}$$

Therefore, the option selection function is

$$CmdOpt = \begin{cases} 1 & \Leftarrow \left[\begin{array}{c} \left\langle \Delta\mathbf{x}_{L\_WP} \middle| \begin{bmatrix} -\sin\gamma & \cos\gamma \\ -\cos\gamma & -\sin\gamma \end{bmatrix} \mathbf{x}_{FL} \right\rangle \leq 0 \wedge Opt = 1 \\ \vee \\ \left\langle \Delta\mathbf{x}_{L\_WP} \middle| \begin{bmatrix} -\sin\gamma & -\cos\gamma \\ \cos\gamma & -\sin\gamma \end{bmatrix} \mathbf{x}_{FL} \right\rangle > 0 \end{array} \right] , \\ -1 & otherwise \end{cases} \tag{5.37}$$

resulting in:

$$\mathbf{v}_{cmd,F} = \begin{cases} \mathbf{v}_{ref_F}\big|_{\kappa=1} & \Leftarrow CmdOpt = 1 \\ \mathbf{v}_{ref_F}\big|_{\kappa=-1} & otherwise \end{cases} . \tag{5.38}$$

If the follower UAV is in the leader's *Buffer Area*, the deconflicting velocity command

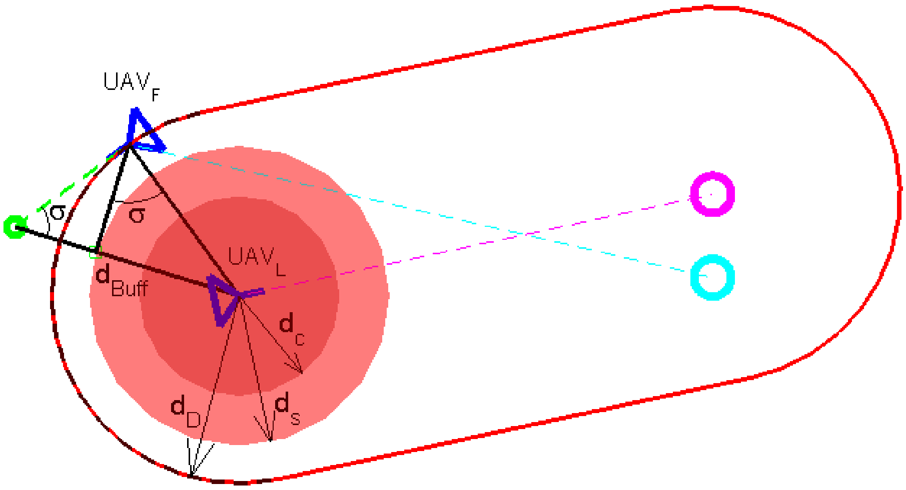


Figure 5.11: Buffer waypoint

will bring it to the *Deconfliction Area*. To achieve smoothness on the interface with all other operation areas, the commanded course has to match the commanded courses in those areas over the interface. This means that the commanded course has to be tangent to the *Deconfliction Area* if interfacing with this area, and it has to be outwards perpendicular to the *frontal* or *Safety Area* if interfacing with those areas. One solution is to aim to a *buffer waypoint* as depicted in figure 5.11, where

$$\sin\sigma = \frac{d_s}{d_{ij}}, \tag{5.39}$$

and the distance between the leader position and the *buffer waypoint* (d_{Buff}) is,

$$d_{Buff} = \frac{d_D^2}{d_s} = d_s \cdot b^2. \tag{5.40}$$

By definition,

$$\mathbf{x}_{BuffWP} := \mathbf{x}_L + \begin{bmatrix} -\sin\sigma & \kappa\cos\sigma \\ -\kappa\cos\sigma & -\sin\sigma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot d_{Buff},\ \kappa = -1, 1. \tag{5.41}$$

Therefore, the commanded velocity options are:

$$\begin{aligned} \mathbf{v}_{ref_F} &= \frac{\mathbf{x}_{BuffWP} - \mathbf{x}_F}{\|\mathbf{x}_{BuffWP} - \mathbf{x}_F\|} \cdot V_{c_F} = \dots \\ \dots &= \frac{\mathbf{x}_{FL} + \begin{bmatrix} -\sin\sigma & \kappa\cos\sigma \\ -\kappa\cos\sigma & -\sin\sigma \end{bmatrix} \frac{d_{Buff}}{\|\mathbf{x}_{FL}\|} \cdot \mathbf{x}_{FL}}{\sqrt{\left(\|\mathbf{x}_{FL}\| + d_{Buff}\right)^2 - 2\,\|\mathbf{x}_{FL}\|\, d_{Buff}\,(1 + \sin\sigma)}} \cdot V_{c_F}. \end{aligned} \tag{5.42}$$

To simplify the notation we can say that

$$\frac{\mathbf{x}_{FL} + \begin{bmatrix} -\sin\sigma & \kappa\cos\sigma \\ -\kappa\cos\sigma & -\sin\sigma \end{bmatrix} \frac{d_{Buff}}{\|\mathbf{x}_{FL}\|} \cdot \mathbf{x}_{FL}}{\sqrt{\left(\|\mathbf{x}_{FL}\| + d_{Buff}\right)^2 - 2\,\|\mathbf{x}_{FL}\|\, d_{Buff}\,(1 + \sin\sigma)}} = \begin{bmatrix} \cos\gamma & \kappa\sin\gamma \\ -\kappa\sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|}, \tag{5.43}$$

with $\gamma = \pi - \arctan\left(\frac{d_{Buff}\cos\sigma}{d_{Buff}\sin\sigma - d_{12}}\right)$, and the commanded velocity options,

$$\mathbf{v}_{ref_F} = \begin{bmatrix} \cos\gamma & \kappa\sin\gamma \\ -\kappa\sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F}. \tag{5.44}$$

In this case, the option selection function is

$$CmdOpt = \begin{cases} 1 & \Leftarrow \begin{bmatrix} \left\langle \Delta\mathbf{x}_{L\_WP} \middle| \begin{bmatrix} -\sin\sigma & \cos\sigma \\ -\cos\sigma & -\sin\sigma \end{bmatrix} \mathbf{x}_{FL} \right\rangle \leq 0 \wedge Optm = 1 \\ \vee \\ \left\langle \Delta\mathbf{x}_{L\_WP} \middle| \begin{bmatrix} -\sin\sigma & -\cos\sigma \\ \cos\sigma & -\sin\sigma \end{bmatrix} \mathbf{x}_{FL} \right\rangle > 0 \end{bmatrix} \\ -1 & otherwise \end{cases}, \tag{5.45}$$

and so,

$$\mathbf{v}_{cmd,F} = \begin{cases} \mathbf{v}_{ref_F}\big|_{\kappa=1} & \Leftarrow CmdOpt = 1 \\ \mathbf{v}_{ref_F}\big|_{\kappa=-1} & otherwise \end{cases}. \tag{5.46}$$

If the follower UAV is in the leader's *Frontal Area*, the deconflicting velocity command will bring it to the *Buffer Area*. In this case, the commanded velocity options are:

$$\mathbf{v}_{ref_F} = \begin{bmatrix} 0 & -\kappa \\ \kappa & 0 \end{bmatrix} \frac{\Delta\mathbf{x}_{L\_WP}}{\|\Delta\mathbf{x}_{L\_WP}\|} \cdot V_{c_F}, \tag{5.47}$$

Only one of these options is valid, the one pointing away from the leader's path. That is verified by:

$$\langle \kappa \Delta \mathbf{x}_{L\_WP} \perp \mathbf{x}_{FL} \rangle \geq 0, \tag{5.48}$$

and so,

$$\mathbf{v}_{cmd,F} = \begin{cases} \mathbf{v}_{ref_F}\big|_{\kappa=1} & \Leftarrow \langle \kappa \Delta \mathbf{x}_{L\_WP} \perp \mathbf{x}_{FL} \rangle \geq 0 \\ \mathbf{v}_{ref_F}\big|_{\kappa=-1} & otherwise \end{cases}. \tag{5.49}$$

The intrusion detection formal definition is:

$$Intrusion \Leftarrow \{\mathbf{x}_j \in ConflArea_{ij},\ i \neq j, i, j = 1, 2\} \tag{5.50}$$

When an intrusion of one UAV in the other's *conflict area* is detected, the intruding UAV is designated the leader and the other is the follower. The commanded actions are the same as for the path interception detection, ceasing with the end of the intrusion.

5.2.7 System Models

We now describe the system behavior under increasingly demanding and realistic assumptions.

5.2.7.1 Model 1 - Single Integrator

In Model 1 we assume the UAVs are perfect holonomic vehicles modeled by the simplest dynamics, namely single integrators:

$$\begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \mathbf{v}_i = \mathbf{v}_{cmd_i} \end{cases}, \tag{5.51}$$

where

$$\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{5.52a}$$

$$\mathbf{v}_{cmd_i} = \begin{bmatrix} u_i \\ v_i \end{bmatrix}. \tag{5.52b}$$

We used three hybrid automatons as a formal definition for the complete system dynamics, including control. They are defined next.

Definition V.21 (Main automaton)**.** (Fig. 5.12)

$$CAS = \left\{\mathcal{Q}, \mathbb{R}^6, f, \Phi\right\} \tag{5.53}$$

$$\mathcal{Q} = \{TrackWP, CPCAA, PDA\} \tag{5.54}$$

$$\mathcal{C} = \{\mathbf{x}_1, \mathbf{x}_2, L, F\} \in \mathbb{R}^6,\ \mathbf{x}_1 = \{x_1, y_1\}, \mathbf{x}_2 = \{x_2, y_2\} \tag{5.55}$$

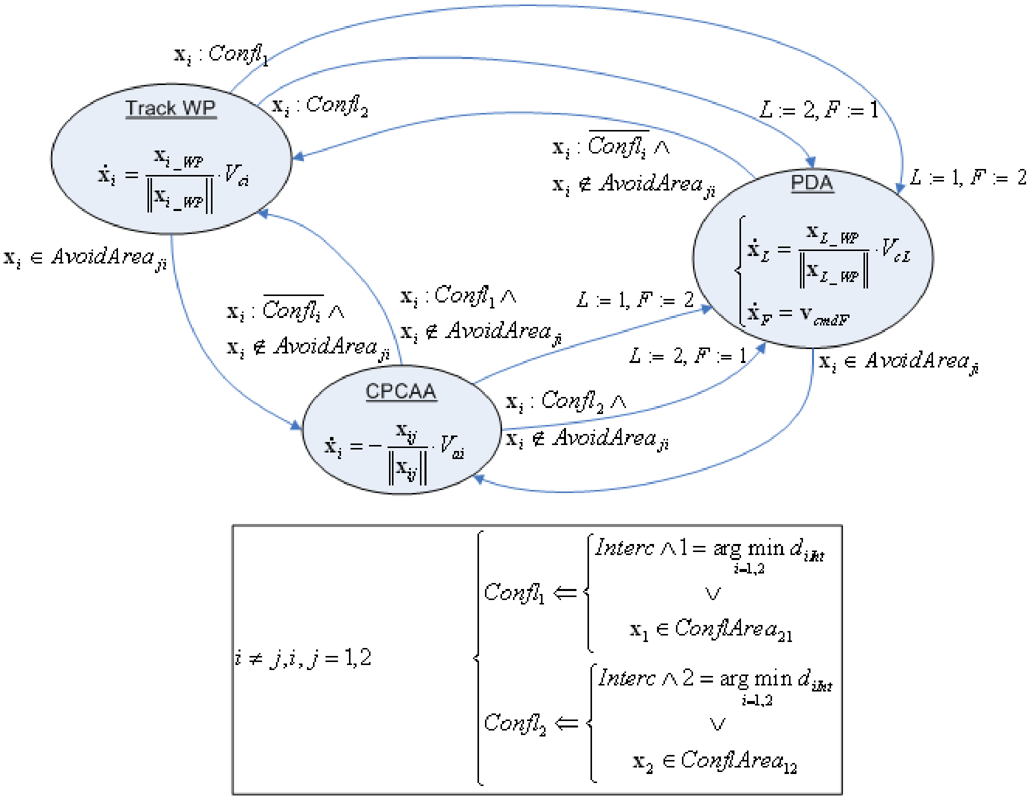
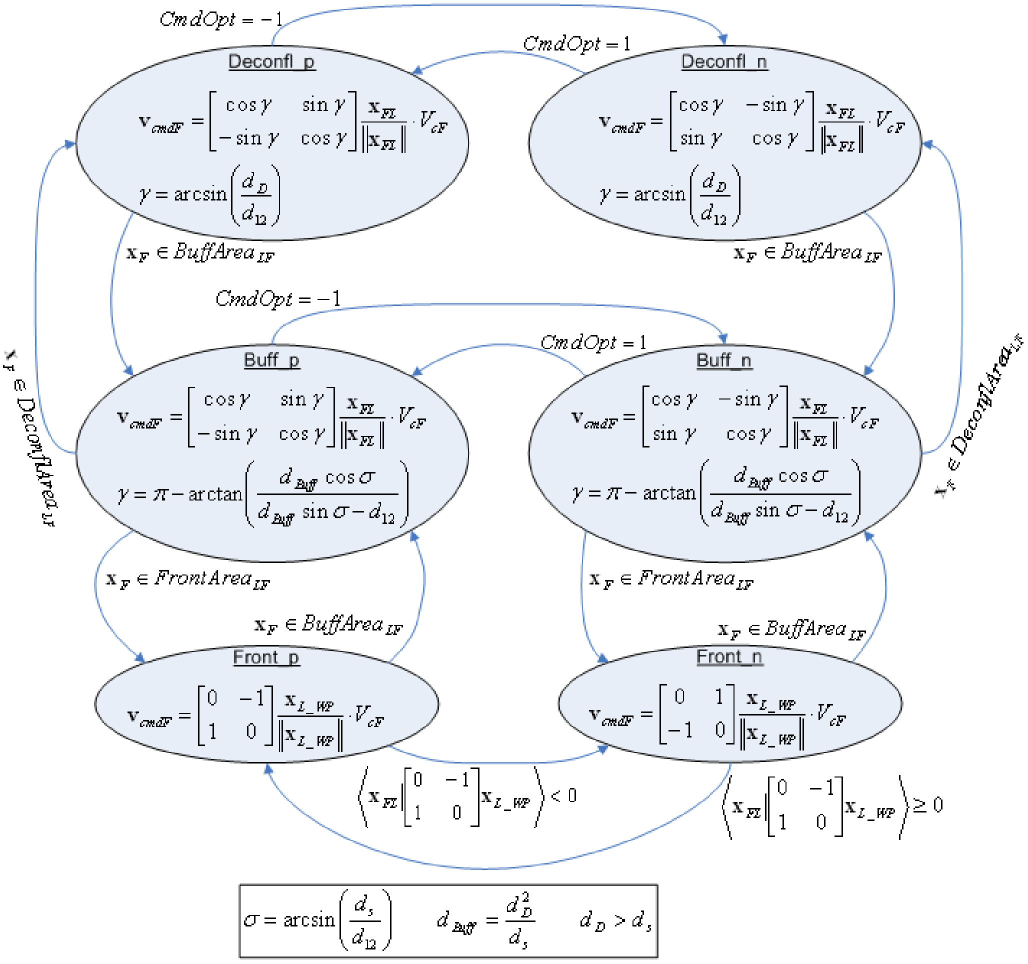


Figure 5.12: Main automaton - holonomic vehicles (single integrator)

$$\dot{\mathcal{C}} = \begin{cases} f\left(TrackWP,\mathcal{C}\right) & \Rightarrow \dot{\mathbf{x}}_i = \frac{\Delta\mathbf{x}_{i\_WP}}{\|\Delta\mathbf{x}_{i\_WP}\|}\cdot V_{c_i},\ i=1,2 \\ f\left(CPCAA,\mathcal{C}\right) & \Rightarrow \dot{\mathbf{x}}_i = -\frac{\Delta\mathbf{x}_{ij}}{\|\Delta\mathbf{x}_{ij}\|}\cdot V_{a_i},\ i \neq j, i, j = 1,2 \\ f\left(PDA,\mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_L = \frac{\Delta\mathbf{x}_{L\_WP}}{\|\Delta\mathbf{x}_{L\_WP}\|}\cdot V_{c_L} \\ \dot{\mathbf{x}}_F = \mathbf{v}_{cmd,F} \end{cases} \end{cases} \tag{5.56}$$

$$\Phi\left(q,\mathcal{X}\right) = \begin{cases} \left(CPCAA,\mathcal{C}\right) & \Leftarrow \begin{bmatrix} \mathbf{x}_1 \in AvoidArea_{21} \\ \vee \\ \mathbf{x}_2 \in AvoidArea_{12} \end{bmatrix} \\ \left(PDA,\mathcal{C} = \begin{cases} \mathbf{x}_1 = \mathbf{x}_1 \\ \mathbf{x}_2 = \mathbf{x}_2 \\ L = 1 \\ F = 2 \end{cases}\right) & \Leftarrow \mathbf{x}_1 \in ConflArea_{21} \vee \begin{bmatrix} Interception \\ \wedge \\ 1 = \arg\min_{i=1,2} d_{iInt} \end{bmatrix} \\ \left(PDA,\mathcal{C} = \begin{cases} \mathbf{x}_1 = \mathbf{x}_1 \\ \mathbf{x}_2 = \mathbf{x}_2 \\ L = 2 \\ F = 1 \end{cases}\right) & \Leftarrow \mathbf{x}_2 \in ConflArea_{12} \vee \begin{bmatrix} Interception \\ \wedge \\ 2 = \arg\min_{i=1,2} d_{iInt} \end{bmatrix} \\ \left(TrackWP,\mathcal{C}\right) & \Leftarrow otherwise \end{cases} \tag{5.57}$$



$$\mathbf{v}_{cmdF} = \begin{bmatrix} \cos\gamma & \sin\gamma \\ -\sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{cF} \qquad \gamma = \arcsin\left(\frac{d_D}{d_{12}}\right)$$

$$\mathbf{v}_{cmdF} = \begin{bmatrix} \cos\gamma & -\sin\gamma \\ \sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{cF} \qquad \gamma = \arcsin\left(\frac{d_D}{d_{12}}\right)$$

$$\gamma = \pi - \arctan\left(\frac{d_{Buff}\cos\sigma}{d_{Buff}\sin\sigma - d_{12}}\right)$$

$$\mathbf{v}_{cmdF} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \frac{\mathbf{x}_{L\_WP}}{\|\mathbf{x}_{L\_WP}\|} \cdot V_{cF} \qquad \mathbf{v}_{cmdF} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \frac{\mathbf{x}_{L\_WP}}{\|\mathbf{x}_{L\_WP}\|} \cdot V_{cF}$$

$$\left\langle \mathbf{x}_{FL} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{x}_{L\_WP} \right\rangle < 0 \qquad \left\langle \mathbf{x}_{FL} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{x}_{L\_WP} \right\rangle \geq 0$$

$$\sigma = \arcsin\left(\frac{d_s}{d_{12}}\right) \qquad d_{Buff} = \frac{d_D^2}{d_s} \qquad d_D > d_s$$

Figure 5.13: Velocity automaton

Definition V.22 (Velocity automaton)**.** This automaton is active only if the main automaton state is *PDA* - Fig. 5.13

$$Veloc = \{\mathcal{P}, \mathbb{R}^3, g, \Psi\} \tag{5.58}$$

$$\mathcal{P} = \{Deconfl\_p, Deconfl\_n, Buff\_p, Buff\_n, Front\_p, Front\_n\} \tag{5.59}$$

$$\mathcal{D} = \{\mathbf{v}_{cmd,F}, \gamma\} \in \mathbb{R}^3,\ \mathbf{v}_{cmd,F} = \{u_F, v_F\} \tag{5.60}$$

$$\dot{\mathcal{D}} = \begin{cases} g\left(Deconfl\_p, \mathcal{D}\right) & \Rightarrow \begin{cases} \mathbf{v}_{cmd,F} = \begin{bmatrix} \cos\gamma & \sin\gamma \\ -\sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F} \\ \gamma = \arcsin\left(\frac{d_D}{d_{12}}\right) \end{cases} \\ g\left(Deconfl\_n, \mathcal{D}\right) & \Rightarrow \begin{cases} \mathbf{v}_{cmd,F} = \begin{bmatrix} \cos\gamma & -\sin\gamma \\ \sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F} \\ \gamma = \arcsin\left(\frac{d_D}{d_{12}}\right) \end{cases} \\ g\left(Buff\_p, \mathcal{D}\right) & \Rightarrow \begin{cases} \mathbf{v}_{cmd,F} = \begin{bmatrix} \cos\gamma & \sin\gamma \\ -\sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F} \\ \gamma = \pi - \arctan\left(\frac{d_{Buff}\cos\sigma}{d_{Buff}\sin\sigma - d_{12}}\right) \end{cases} \\ g\left(Buff\_n, \mathcal{D}\right) & \Rightarrow \begin{cases} \mathbf{v}_{cmd,F} = \begin{bmatrix} \cos\gamma & -\sin\gamma \\ \sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F} \\ \gamma = \pi - \arctan\left(\frac{d_{Buff}\cos\sigma}{d_{Buff}\sin\sigma - d_{12}}\right) \end{cases} \\ g\left(Front\_p, \mathcal{D}\right) & \Rightarrow \mathbf{v}_{cmd,F} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \frac{\Delta\mathbf{x}_{L\_WP}}{\|\Delta\mathbf{x}_{L\_WP}\|} \cdot V_{c_F} \\ g\left(Front\_n, \mathcal{D}\right) & \Rightarrow \mathbf{v}_{cmd,F} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \frac{\Delta\mathbf{x}_{L\_WP}}{\|\Delta\mathbf{x}_{L\_WP}\|} \cdot V_{c_F} \end{cases} \tag{5.61}$$

$$\Psi\left(q, \mathcal{X}\right) = \begin{cases} \left(Deconfl\_p, \mathcal{D}\right) & \Leftarrow \mathbf{x}_F \in DeconflArea_{LF} \wedge CmdOpt = 1 \\ \left(Deconfl\_n, \mathcal{D}\right) & \Leftarrow \mathbf{x}_F \in DeconflArea_{LF} \wedge CmdOpt = -1 \\ \left(Buff\_p, \mathcal{D}\right) & \Leftarrow \mathbf{x}_F \in BuffArea_{LF} \wedge CmdOpt = 1 \\ \left(Buff\_n, \mathcal{D}\right) & \Leftarrow \mathbf{x}_F \in BuffArea_{LF} \wedge CmdOpt = -1 \\ \left(Front\_p, \mathcal{D}\right) & \Leftarrow \mathbf{x}_F \in FrontArea_{LF} \wedge \langle \Delta\mathbf{x}_{L\_WP} \perp \mathbf{x}_{FL} \rangle \geq 0 \\ \left(Front\_n, \mathcal{D}\right) & \Leftarrow \mathbf{x}_F \in FrontArea_{LF} \wedge \sim \langle \Delta\mathbf{x}_{L\_WP} \perp \mathbf{x}_{FL} \rangle \geq 0 \end{cases} \tag{5.62}$$

Definition V.23 (Waypoint automaton)**.** (Fig. 5.14)

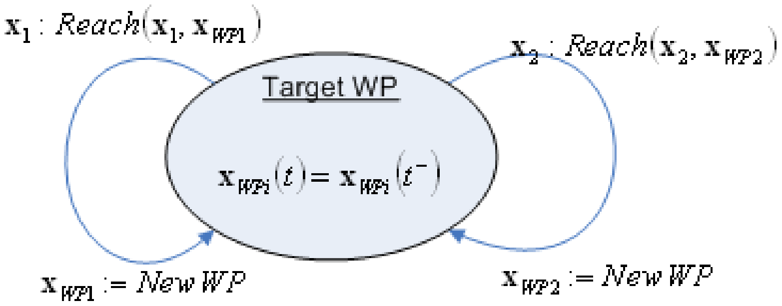


Figure 5.14: Waypoint automaton

$$GenWP = \left\{\mathcal{O}, \mathbb{R}^4, h, \Upsilon\right\} \tag{5.63}$$

$$\mathcal{O} = \{TargetWP\} \tag{5.64}$$

$$\mathcal{E} = \{\mathbf{x}_{WP1}, \mathbf{x}_{WP2}\} \in \mathbb{R}^4,\ \mathbf{x}_{WP1} = \{x_{WP1}, y_{WP1}\}, \mathbf{x}_{WP2} = \{x_{WP2}, y_{WP2}\} \tag{5.65}$$

$$\Upsilon(q, \mathcal{X}) = \left\{ \begin{array}{ll} \left(TargetWP, \mathcal{E} = \left\{ \begin{array}{l} \mathbf{x}_{WP1}(t) = NewWP \\ \mathbf{x}_{WP2}(t) = \mathbf{x}_{WP2}(t^-) \end{array} \right. \right) & \Leftarrow Reach(\mathbf{x}_1, \mathbf{x}_{WP1}) \\ \left(TargetWP, \mathcal{E} = \left\{ \begin{array}{l} \mathbf{x}_{WP1}(t) = \mathbf{x}_{WP1}(t^-) \\ \mathbf{x}_{WP2}(t) = NewWP \end{array} \right. \right) & \Leftarrow Reach(\mathbf{x}_2, \mathbf{x}_{WP2}) \end{array} \right. , \tag{5.66}$$

where *NewWP* is the waypoint generation function and $Reach(\mathbf{x}_i, \mathbf{x}_{WPi})$ is the boolean function containing the *reach* assertion logic for UAV i and the respective waypoint.

For this system we are able to prove that it presents both the *Safety* and *Liveliness* properties.

Theorem V.24. *The overall system satisfies the* Safety *property, if we assume single integrator holonomic vehicles, if the* CPCAA *is part of the control system, and if* $(\|\Delta\mathbf{x}_{12}\| \geq d_{s_{12}})(t_0)$ *holds true.*

Theorem V.25. *The overall system satisfies the* Liveness *property, if we assume single integrator holonomic vehicles, if the* PDA *and* TrackWP *states are part of the control system, and if* $(\|\Delta\mathbf{x}_{12}\| > d_{s_{12}} \wedge \mathbf{x}_F \notin CloseFArea_{LF})(t_0)$ *holds true:*

$$\begin{aligned} &(\mathbf{x}_F(t) \notin \{SafeArea_{LF} \vee CloseFArea_{LF}\} \Rightarrow ... \\ &... \Rightarrow \{\Diamond \|\Delta\mathbf{x}_{1\_WP}\| \leq d_{WP1} \wedge \Diamond \|\Delta\mathbf{x}_{2\_WP}\| \leq d_{WP2}\})(t). \end{aligned} \tag{5.67}$$

The proofs for these theorems are shown in the appendix B.1.

5.2.7.2 Model 2 - Double Integrator

In this second model we assume the UAVs are holonomic vehicles modeled by a double integrator:

$$\left\{ \begin{array}{l} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = \mathbf{a}_{cmd_i} \end{array} \right. , \tag{5.68}$$

where

$$\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{5.69a}$$

$$\mathbf{v}_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} \tag{5.69b}$$

$$\mathbf{a}_{cmd_i} = k_a(\mathbf{v}_{cmd_i} - \mathbf{v}_i). \tag{5.69c}$$

We use similar hybrid automatons to the ones in model 1. The only difference is in the main automaton.

Definition V.26 (Main automaton)**.** (Fig. 5.15)

$$CAS = \left\{\mathcal{Q}, \mathbb{R}^{10}, f, \Phi\right\} \tag{5.70}$$

$$\mathcal{Q} = \{TrackWP, CPCAA, PDA\} \tag{5.71}$$

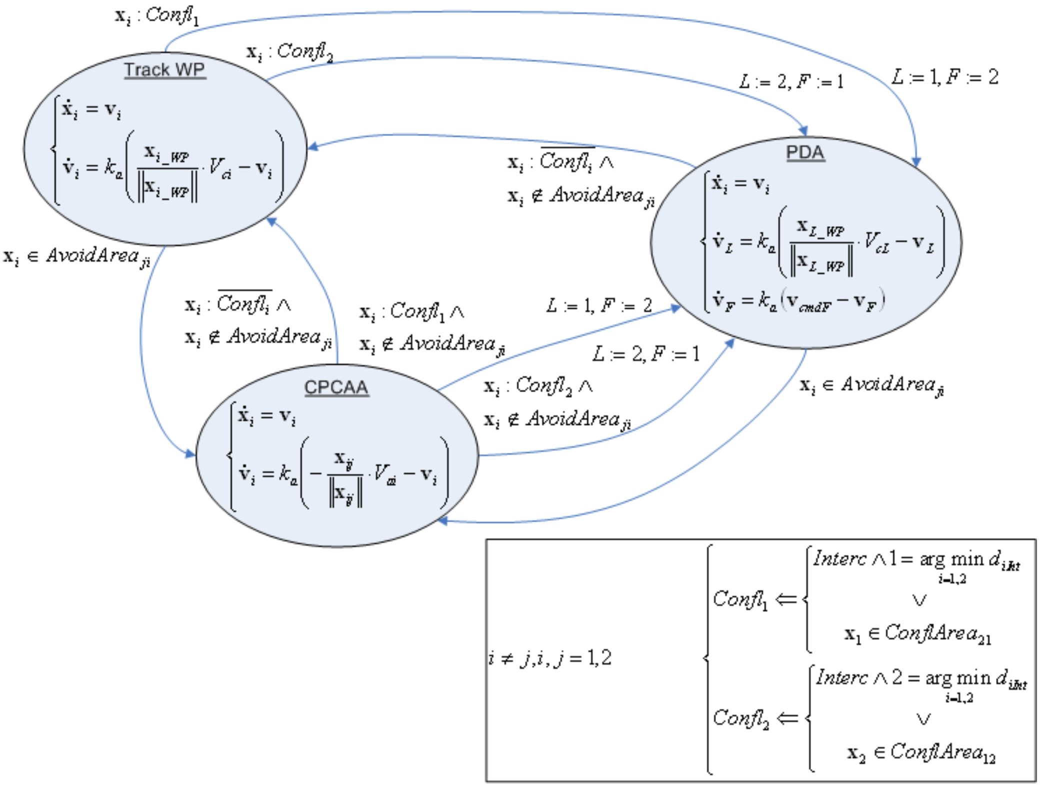


Figure 5.15: Main automaton - holonomic vehicles (double integrator)

$$\mathcal{C} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{v}_1, \mathbf{v}_2, L, F\} \in \mathbb{R}^{10}, \mathbf{x}_1 = \{x_1, y_1\}, \mathbf{x}_2 = \{x_2, y_2\}, \mathbf{v}_1 = \{u_1, v_1\}, \mathbf{v}_2 = \{u_2, v_2\} \tag{5.72}$$

$$\dot{\mathcal{C}} = \begin{cases} f(TrackWP, \mathcal{C}) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = k_a \left(\frac{\Delta \mathbf{x}_{i\_WP}}{\|\Delta \mathbf{x}_{i\_WP}\|} \cdot V_{c_i} - \mathbf{v}_i \right) \end{cases}, \; i = 1,2 \\ f(CPCAA, \mathcal{C}) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = k_a \left(-\frac{\Delta \mathbf{x}_{ij}}{\|\Delta \mathbf{x}_{ij}\|} \cdot V_{a_i} - \mathbf{v}_i \right) \end{cases}, \; i \neq j, i, j = 1,2 \\ f(PDA, \mathcal{C}) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i, \; i = 1,2 \\ \dot{\mathbf{v}}_L = k_a \left(\frac{\Delta \mathbf{x}_{L\_WP}}{\|\Delta \mathbf{x}_{L\_WP}\|} \cdot V_{c_L} - \mathbf{v}_L \right) \\ \dot{\mathbf{v}}_F = k_a \left(\mathbf{v}_{cmd,F} - \mathbf{v}_F \right) \end{cases} \end{cases} \tag{5.73}$$

$$\Phi\left(q,\mathcal{X}\right) = \begin{cases} \left(CPCAA, \mathcal{C}\right) & \Leftarrow \begin{bmatrix} \mathbf{x}_1 \in AvoidArea_{21} \\ \vee \\ \mathbf{x}_2 \in AvoidArea_{12} \end{bmatrix} \\ \left(PDA, \mathcal{C} = \begin{cases} \mathbf{x}_1 = \mathbf{x}_1 \\ \mathbf{x}_2 = \mathbf{x}_2 \\ L = 1 \\ F = 2 \end{cases} \right) & \Leftarrow \mathbf{x}_1 \in ConflArea_{21} \vee \begin{bmatrix} Interception \\ \wedge \\ 1 = \underset{i=1,2}{\arg\min} d_{iInt} \end{bmatrix} \\ \left(PDA, \mathcal{C} = \begin{cases} \mathbf{x}_1 = \mathbf{x}_1 \\ \mathbf{x}_2 = \mathbf{x}_2 \\ L = 2 \\ F = 1 \end{cases} \right) & \Leftarrow \mathbf{x}_2 \in ConflArea_{12} \vee \begin{bmatrix} Interception \\ \wedge \\ 2 = \underset{i=1,2}{\arg\min} d_{iInt} \end{bmatrix} \\ \left(TrackWP, \mathcal{C}\right) & \Leftarrow otherwise \end{cases} \tag{5.74}$$

For this system we prove that it presents the *Safety* property.

Theorem V.27. *The overall system satisfies the* Safety *property, if we assume double integrator holonomic vehicles, if the* CPCAA *is part of the control system, and if the following restrictions hold true,*

$$\left(\left\|\Delta \mathbf{x}_{12}\right\| \geq d_{s_{12}}\right)\left(t_0\right) \tag{5.75a}$$

$$d_{s_{12}} > d_{c_{12}} - \frac{V_{a_2} + V_{a_1}}{k_a} \ln\left(\frac{V_{c_1} + V_{c_2}}{V_{a_2} + V_{a_1}} + 1\right) + \frac{V_{c_2} + V_{c_1}}{k_a} \tag{5.75b}$$

The proof for this theorem is shown in the appendix B.2.

The *Liveness* property conditions should be similar to the ones defined for theorem V.25.

5.2.7.3 Model 3 - Control Delays

In this third model the control system is discrete and has delays. We use the same dynamics model as in model 2, but with $\mathbf{a}_{cmd_i} = ZOH\left(\mathbf{v}_{cmd_i}, t, \Delta t\right) - \mathbf{v}_i$. The $ZOH\left(\mathbf{v}, t, \Delta t\right)$ is piecewise-constant function holding the value of $\mathbf{v}\left(t_k\right)$ until $t = t_k + \Delta t$:

$$ZOH\left(\mathbf{v}, t, \Delta t\right) = \left\{\mathbf{v}\left(t_k\right) : t = \left[t_k,\ t_k + \Delta t\right), t \in \mathbb{R}\right\} \tag{5.76}$$

Both the velocity automaton and the waypoint automaton remain the same. The main automaton is described next.

Definition V.28 (Main automaton)**.** (Fig. 5.16)

$$CAS = \left\{\mathcal{Q}, \mathbb{R}^{10}, f, \Phi\right\} \tag{5.77}$$

$$\mathcal{Q} = \left\{TrackWP, CPCAA, PDA\right\} \tag{5.78}$$

$$\mathcal{C} = \left\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{v}_1, \mathbf{v}_2, L, F\right\} \in \mathbb{R}^{10}, \mathbf{x}_1 = \left\{x_1, y_1\right\}, \mathbf{x}_2 = \left\{x_2, y_2\right\}, \mathbf{v}_1 = \left\{u_1, v_1\right\}, \mathbf{v}_2 = \left\{u_2, v_2\right\} \tag{5.79}$$

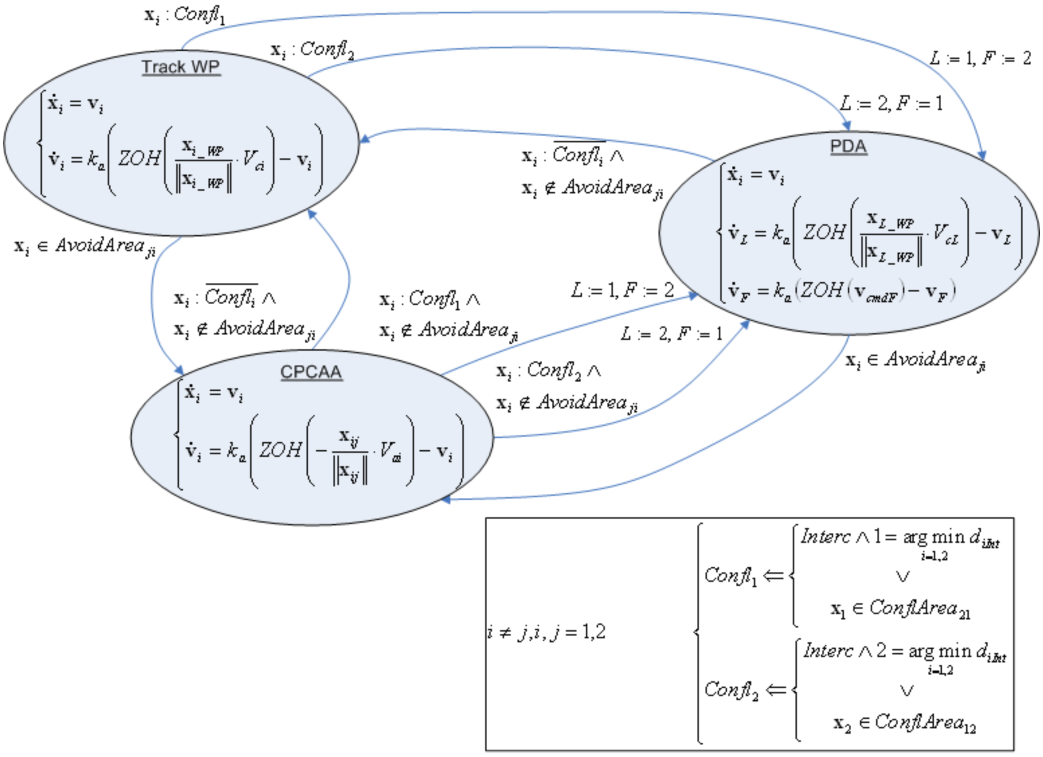


Figure 5.16: Main automaton - holonomic vehicles (control delays)

$$\dot{\mathcal{C}} = \begin{cases} f\left(TrackWP, \mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = k_a\left(ZOH\left(\frac{\Delta\mathbf{x}_{i\_WP}}{\|\Delta\mathbf{x}_{i\_WP}\|}\cdot V_{c_i}, t, \Delta t\right) - \mathbf{v}_i\right) \end{cases} , \ i = 1,2 \\ f\left(CPCAA, \mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = k_a\left(ZOH\left(-\frac{\Delta\mathbf{x}_{ij}}{\|\Delta\mathbf{x}_{ij}\|}\cdot V_{a_i}, t, \Delta t\right) - \mathbf{v}_i\right) \end{cases} , \ i \neq j, i, j = 1,2 \\ f\left(PDA, \mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i, \ i = 1,2 \\ \dot{\mathbf{v}}_L = k_a\left(ZOH\left(\frac{\Delta\mathbf{x}_{L\_WP}}{\|\Delta\mathbf{x}_{L\_WP}\|}\cdot V_{c_L}, t, \Delta t\right) - \mathbf{v}_L\right) \\ \dot{\mathbf{v}}_F = k_a\left(ZOH\left(\mathbf{v}_{cmd,F}, t, \Delta t\right) - \mathbf{v}_F\right) \end{cases} \end{cases} \tag{5.80}$$

$$\Phi\left(q,\mathcal{X}\right)=\begin{cases}
(CPCAA,\mathcal{C}) & \Leftarrow \begin{bmatrix} \mathbf{x}_1\in AvoidArea_{21}\\ \vee \\ \mathbf{x}_2\in AvoidArea_{12}\end{bmatrix}\\
\left(PDA,\mathcal{C}=\begin{cases}\mathbf{x}_1=\mathbf{x}_1\\ \mathbf{x}_2=\mathbf{x}_2\\ L=1\\ F=2\end{cases}\right) & \Leftarrow \mathbf{x}_1\in ConflArea_{21}\vee\begin{bmatrix} Interception\\ \wedge\\ 1=\arg\min\limits_{i=1,2} d_{iInt}\end{bmatrix}\\
\left(PDA,\mathcal{C}=\begin{cases}\mathbf{x}_1=\mathbf{x}_1\\ \mathbf{x}_2=\mathbf{x}_2\\ L=2\\ F=1\end{cases}\right) & \Leftarrow \mathbf{x}_2\in ConflArea_{12}\vee\begin{bmatrix} Interception\\ \wedge\\ 2=\arg\min\limits_{i=1,2} d_{iInt}\end{bmatrix}\\
(TrackWP,\mathcal{C}) & \Leftarrow otherwise
\end{cases} \tag{5.81}$$

For this system we prove that it presents the *Safety* property.

Theorem V.29. *The overall system satisfies the* Safety *property, even with control noise, if we assume double integrator holonomic vehicles, if the* CPCAA *is part of the control system, and if the following restrictions hold true,*

$$\left(\|\Delta\mathbf{x}_{12}\|\geq d_{s_{12}}\right)(t_0) \tag{5.82a}$$

$$d_{s_{12}}\geq d_{c_{12}}-\frac{V_{a_2}+V_{a_1}}{k_a}\ln\left(\frac{V_{c_1}+V_{c_2}}{V_{a_2}+V_{a_1}}+1\right)+\left(V_{c_2}+V_{c_1}\right)\left(\Delta t+\frac{1}{k_a}\right) \tag{5.82b}$$

The proof for this theorem is shown in appendix B.3.

As for the previous model, the *Liveness* property conditions should be similar to the ones defined for theorem V.25.

5.2.7.4 Model 4 - Control Disturbances

Model 4 extends model 3 introducing control disturbances. The difference to model 3 is that the velocity is no longer certain, but rather bounded on a set of velocities:

$$\mathbf{v}_i=\mathbf{v}_{iRef}+\overline{B_{\delta v}}\left(0,0\right)=\overline{B_{\delta v}}\left(\mathbf{v}_{iRef}\right) \tag{5.83}$$

where δv is the norm of the maximum velocity error,

$$\delta v=\|\mathbf{v}_{iRef}-\mathbf{v}_i\| \tag{5.84}$$

and $\overline{B_{\delta v}}\left(\mathbf{v}\right)$ is the closed ball centred on $\mathbf{v}$ and with radius δv. Both the velocity automaton and the waypoint automaton remain the same. The main automaton is described next.

Definition V.30 (Main automaton)**.** (Fig. 5.17)

$$CAS=\left\{\mathcal{Q},\mathbb{R}^{10},f,\Phi\right\} \tag{5.85}$$

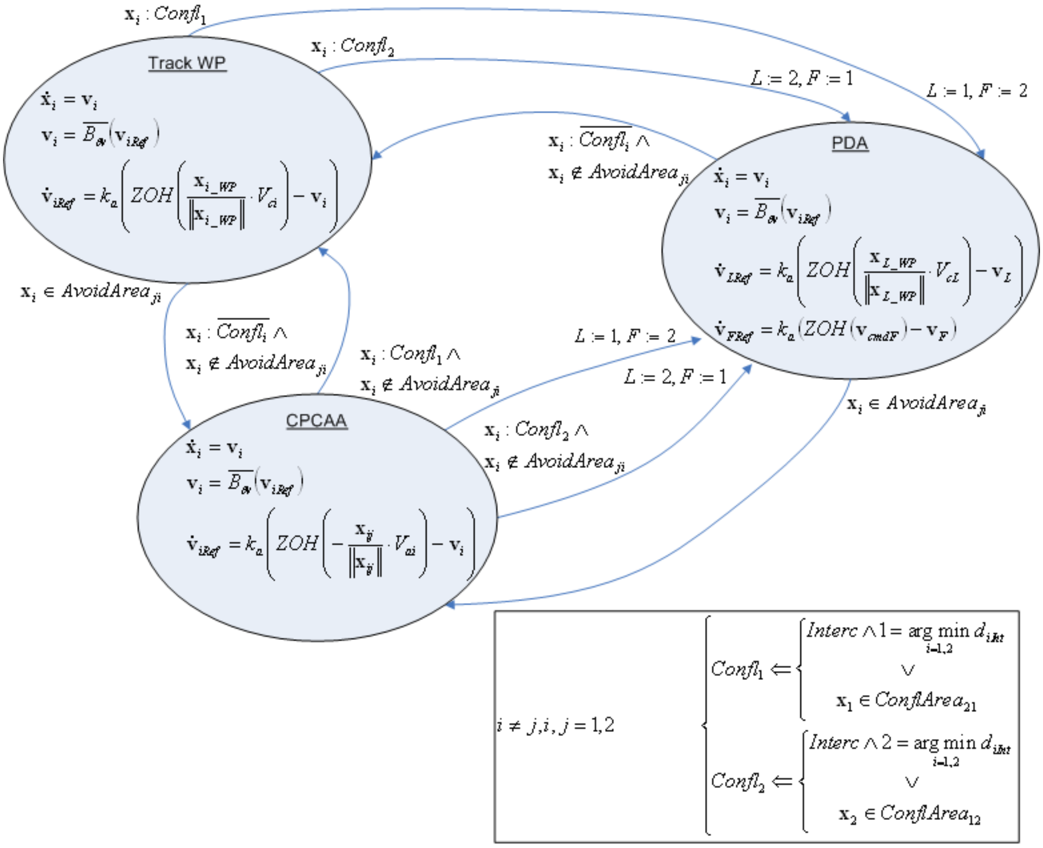


Figure 5.17: Main automaton - holonomic vehicles (control disturbances)

$$\mathcal{Q} = \{TrackWP, CPCAA, PDA\} \tag{5.86}$$

$$\mathcal{C} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{v}_1, \mathbf{v}_2, L, F\} \in \mathbb{R}^{10}, \mathbf{x}_1 = \{x_1, y_1\}, \mathbf{x}_2 = \{x_2, y_2\}, \mathbf{v}_1 = \{u_1, v_1\}, \mathbf{v}_2 = \{u_2, v_2\} \tag{5.87}$$

$$\dot{\mathcal{C}} = \begin{cases} f\left(TrackWP, \mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \mathbf{v}_i = \overline{B_{\delta v}}\left(\mathbf{v}_{iRef}\right) \\ \dot{\mathbf{v}}_{iRef} = k_a\left(ZOH\left(\frac{\Delta\mathbf{x}_{i\_WP}}{\|\Delta\mathbf{x}_{i\_WP}\|} \cdot V_{c_i}, t, \Delta t\right) - \mathbf{v}_i\right) \end{cases} & , i = 1, 2 \\ f\left(CPCAA, \mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \mathbf{v}_i = \overline{B_{\delta v}}\left(\mathbf{v}_{iRef}\right) \\ \dot{\mathbf{v}}_{iRef} = k_a\left(ZOH\left(-\frac{\Delta\mathbf{x}_{ij}}{\|\Delta\mathbf{x}_{ij}\|} \cdot V_{a_i}, t, \Delta t\right) - \mathbf{v}_i\right) \end{cases} & , i \neq j, i, j = 1, 2 \\ f\left(PDA, \mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \mathbf{v}_i \\ \mathbf{v}_i = \overline{B_{\delta v}}\left(\mathbf{v}_{iRef}\right) \\ \dot{\mathbf{v}}_{LRef} = k_a\left(ZOH\left(\frac{\Delta\mathbf{x}_{L\_WP}}{\|\Delta\mathbf{x}_{L\_WP}\|} \cdot V_{c_L}, t, \Delta t\right) - \mathbf{v}_L\right) \\ \dot{\mathbf{v}}_{FRef} = k_a\left(ZOH\left(\mathbf{v}_{cmd,F}, t, \Delta t\right) - \mathbf{v}_F\right) \end{cases} & , i = L, F \end{cases} \tag{5.88}$$

$$\Phi\left(q, \mathcal{X}\right) = \begin{cases} \left(CPCAA, \mathcal{C}\right) & \Leftarrow \begin{bmatrix} \mathbf{x}_1 \in AvoidArea_{21} \\ \vee \\ \mathbf{x}_2 \in AvoidArea_{12} \end{bmatrix} \\ \left(PDA, \mathcal{C} = \begin{cases} \mathbf{x}_1 = \mathbf{x}_1 \\ \mathbf{x}_2 = \mathbf{x}_2 \\ L = 1 \\ F = 2 \end{cases}\right) & \Leftarrow \mathbf{x}_1 \in ConflArea_{21} \vee \begin{bmatrix} Interception \\ \wedge \\ 1 = \arg\min\limits_{i=1,2} d_{iInt} \end{bmatrix} \\ \left(PDA, \mathcal{C} = \begin{cases} \mathbf{x}_1 = \mathbf{x}_1 \\ \mathbf{x}_2 = \mathbf{x}_2 \\ L = 2 \\ F = 1 \end{cases}\right) & \Leftarrow \mathbf{x}_2 \in ConflArea_{12} \vee \begin{bmatrix} Interception \\ \wedge \\ 2 = \arg\min\limits_{i=1,2} d_{iInt} \end{bmatrix} \\ \left(TrackWP, \mathcal{C}\right) & \Leftarrow otherwise \end{cases} \tag{5.89}$$

5.2.7.5 Model 5 - Unicycle

Model 5 extends model 1 to non-holonomic vehicles. The aircraft dynamics are now modeled by a simple 3 Degrees-of-Freedom (DOF) kinematic model:

$$\begin{cases} \dot{x} = V \cdot \cos\left(\chi\right) \\ \dot{y} = V \cdot \sin\left(\chi\right) \\ \dot{\chi} = \omega = \frac{g \cdot \tan(\phi)}{V} \end{cases} \tag{5.90}$$

where $\phi = k_\omega\left(\dot{x} \cdot v_{Ref} - \dot{y} \cdot u_{Ref}\right)$ and $V = \|\mathbf{v}_{Ref}\|$. As in all previous problems, both the velocity and the waypoint automaton remain the same. The main automaton is described next.

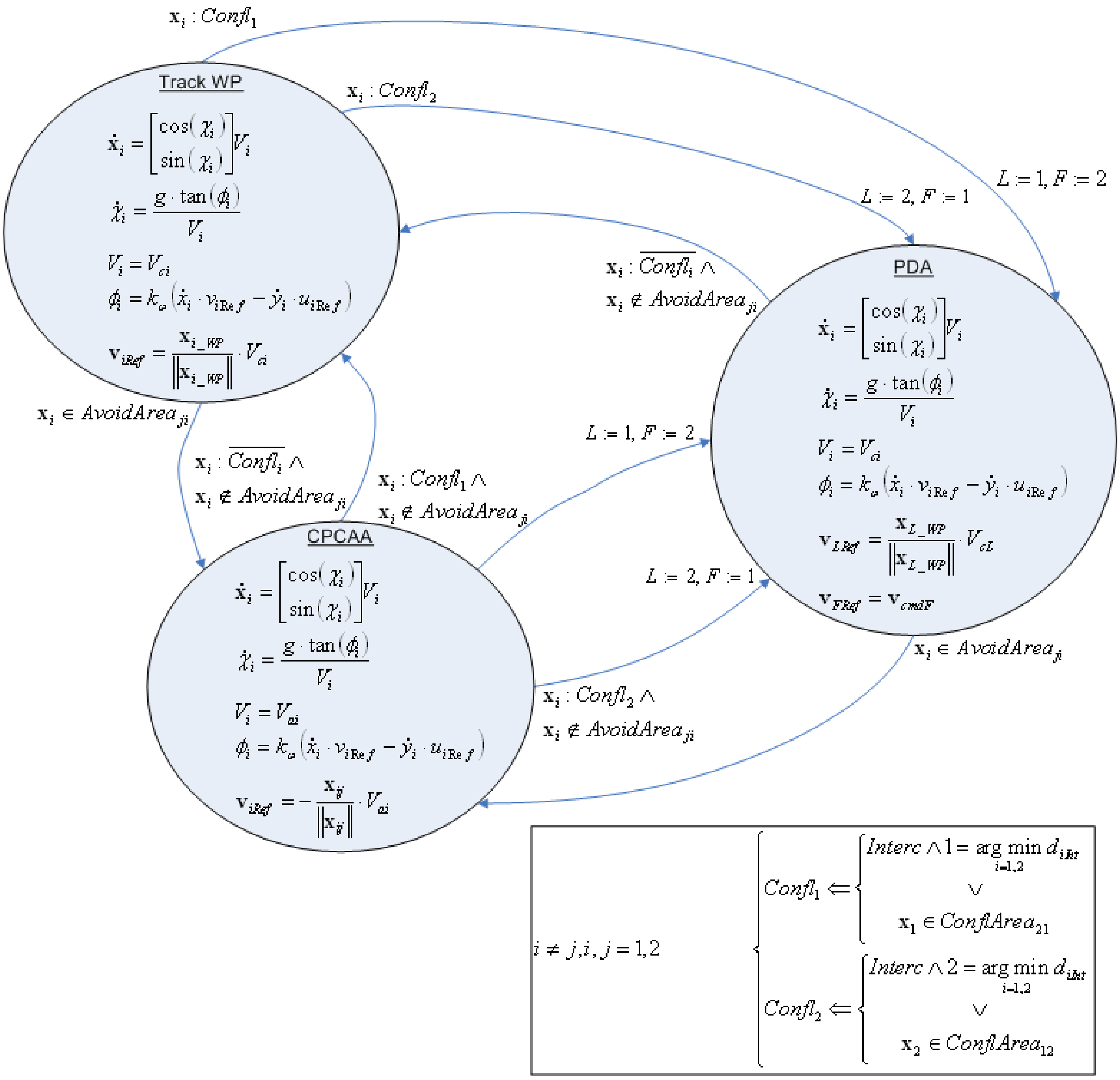


Figure 5.18: Main automaton - non-holonomic vehicles

Definition V.31 (Main automaton)**.** (Fig. 5.18)

$$CAS = \left\{\mathcal{Q}, \mathbb{R}^8, f, \Phi\right\} \tag{5.91}$$

$$\mathcal{Q} = \{TrackWP, CPCAA, PDA\} \tag{5.92}$$

$$\mathcal{C} = \{\mathbf{x}_1, \mathbf{x}_2, \chi_1, \chi_2, L, F\} \in \mathbb{R}^8,\ \mathbf{x}_1 = \{x_1, y_1\}, \mathbf{x}_2 = \{x_2, y_2\} \tag{5.93}$$

$$\dot{\mathcal{C}} = \begin{cases} f\left(TrackWP, \mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \begin{bmatrix} \cos\left(\chi_i\right) \\ \sin\left(\chi_i\right) \end{bmatrix} \cdot V_i \\ \dot{\chi}_i = \frac{g \cdot \tan(\phi_i)}{V_i} \\ V_i = V_{c_i} \\ \phi_i = k_\omega\left(\dot{x}_i \cdot v_{iRef} - \dot{y}_i \cdot u_{iRef}\right) \\ \mathbf{v}_{iRef} = \frac{\Delta\mathbf{x}_{i\_WP}}{\|\Delta\mathbf{x}_{i\_WP}\|} \cdot V_{c_i} \end{cases} \quad, i = 1, 2 \\ f\left(CPCAA, \mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \begin{bmatrix} \cos\left(\chi_i\right) \\ \sin\left(\chi_i\right) \end{bmatrix} \cdot V_i \\ \dot{\chi}_i = \frac{g \cdot \tan(\phi_i)}{V_i} \\ V_i = V_{a_i} \\ \phi_i = k_\omega\left(\dot{x}_i \cdot v_{iRef} - \dot{y}_i \cdot u_{iRef}\right) \\ \mathbf{v}_{iRef} = -\frac{\Delta\mathbf{x}_{ij}}{\|\Delta\mathbf{x}_{ij}\|} \cdot V_{a_i} \end{cases} \quad, i \neq j, i, j = 1, 2 \\ f\left(PDA, \mathcal{C}\right) & \Rightarrow \begin{cases} \dot{\mathbf{x}}_i = \begin{bmatrix} \cos\left(\chi_i\right) \\ \sin\left(\chi_i\right) \end{bmatrix} \cdot V_i \\ \dot{\chi}_i = \frac{g \cdot \tan(\phi_i)}{V_i} \\ V_i = V_{c_i} \\ \phi_i = k_\omega\left(\dot{x}_i \cdot v_{iRef} - \dot{y}_i \cdot u_{iRef}\right) \\ \mathbf{v}_{LRef} = \frac{\Delta\mathbf{x}_{L\_WP}}{\|\Delta\mathbf{x}_{L\_WP}\|} \cdot V_{c_L} \\ \mathbf{v}_{FRef} = \mathbf{v}_{cmd,F} \end{cases} \quad, i = L, F \end{cases} \tag{5.94}$$

$$\Phi\left(q, \mathcal{X}\right) = \begin{cases} \left(CPCAA, \mathcal{C}\right) & \Leftarrow \begin{bmatrix} \mathbf{x}_1 \in AvoidArea_{21} \\ \vee \\ \mathbf{x}_2 \in AvoidArea_{12} \end{bmatrix} \\ \left(PDA, \mathcal{C} = \begin{cases} \mathbf{x}_1 = \mathbf{x}_1 \\ \mathbf{x}_2 = \mathbf{x}_2 \\ L = 1 \\ F = 2 \end{cases}\right) & \Leftarrow \mathbf{x}_1 \in ConflArea_{21} \vee \begin{bmatrix} Interception \\ \wedge \\ 1 = \underset{i=1,2}{\arg\min} d_{iInt} \end{bmatrix} \\ \left(PDA, \mathcal{C} = \begin{cases} \mathbf{x}_1 = \mathbf{x}_1 \\ \mathbf{x}_2 = \mathbf{x}_2 \\ L = 2 \\ F = 1 \end{cases}\right) & \Leftarrow \mathbf{x}_2 \in ConflArea_{12} \vee \begin{bmatrix} Interception \\ \wedge \\ 2 = \underset{i=1,2}{\arg\min} d_{iInt} \end{bmatrix} \\ \left(TrackWP, \mathcal{C}\right) & \Leftarrow otherwise \end{cases} \tag{5.95}$$

5.2.8 Flight Results

This Collision Avoidance System was implemented on the University of Michigan Aerospace & Robotic Controls Laboratory indoor testbed. Figure 5.19 depicts the data flow of the whole control system. The operator interacts with a desktop computer where all the control automatons run (waypoint generator, CAS, and FCS). The vehicles are model helicopter with infrared reflectors. The Vicon system is a motion capture system which provides position and attitude data to the control computer. It locates the helicopter through cameras which detect the infrared reflection dots on each vehicle.

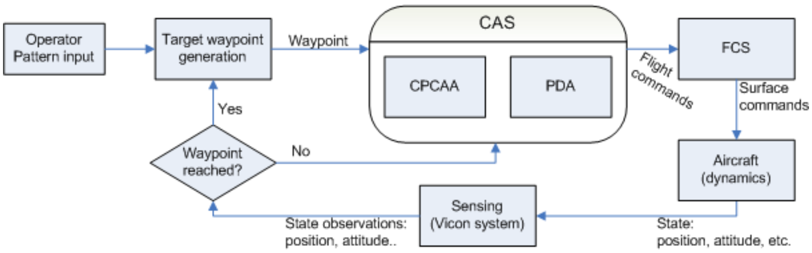


Figure 5.19: Data flow

During flight tests, although the helicopters sometimes got quite close, the system was able to avoid any collision. The proximity reached by the helicopter is probably due to the delay in the data exchange system and the controller running on MatLab.

5.3 Formation Control

We now present a formation flight control strategy featuring a collision avoidance scheme. The control algorithm is based on a Sliding Mode controller. The controller sliding surfaces account for aircraft maneuvering limitations, restricting the required velocities to a feasible set. Further, the relative position between vehicles affects the sliding surfaces shape, enabling the collision avoidance property. The control method derivation is based on an extended unicycle model, resulting on a controller adequate for fixed wing aircraft. Further, the control strategy handles static formations as well as dynamic. This makes it a very flexible tool which may be used by the UAV operators for several flight stages and tasks.

Formation flight is an interesting topic in aircraft control as it presents the challenge of interconnected systems [89]. Some applications include air refueling, forest fire mapping [90], radar deception, distributed sensing [16], fuel saving [91, 92], transportation [93, 94], communication networks [95], and more. For atmospheric data sampling [16] in particular, formation flight allows us to gather spatially distributed measurements simultaneously and control the geometric distribution of the samples. This is even more important when we need close samples at the same altitude.

NASA has been involved in a series of flight trials with automatic formation control. Vachon et al. [91] described in 2002 the performance benefits obtained in a formation of two F/A-18B. It was demonstrated that the formation flight allows performance improvements, reducing up to 20% of the follower's drag and saving up to

18% in fuel. Similar tests were extended to Unmanned Air Vehicles (UAVs) in February 21st 2003, demonstrating a formation flight with an X-45A and an F/A-18B, and in in August 1st 2004 with two X-45A vehicles. These flights were part of the X-45A vehicles flight trials in the Joint Unmanned Combat Air System (J-UCAS) program [96].

5.3.1 General Problem

The formation flight control problem is dominated by the issue of forcing a relative position among aircraft. Further, there are two other important issues in formation control: bringing the formation together and avoiding any collision.

5.3.2 Literature Review

Zheng et al. [97] derive a Sliding Mode controller for a leaderless formation and applied it to a group of holonomic vehicles. This method is tested with ground robots with good results. Sharifi et al. [98] present a decentralized sliding mode control method for a leader formation with communication delays. This study shows how the decentralized controller on one vehicle predicts other vehicles' dynamics and how this improves formation stability. Fowler and D'Andrea [92] describe the benefits of distributed control synthesis when compared to a centralized controller. The synthesized controllers are demonstrated on a special wind tunnel apparatus. Gu et al. [99] present a leader formation control method.Two YF-22 models were used to test the method in real flight. The control scheme is divided into an innerloop and an outerloop controller. The innerloop and the altitude outerloop are linear controllers and the horizontal position controller is based on Nonlinear Dynamic Inversion. Tests show that the follower is able to track its formation position safely. The main requirements are that the flight condition remains near the one used for the controller parameters estimation and that the system starts from the same quadrant as the desired position relative to the leader, achieved by a pre-positioning maneuver executed by a human pilot. Bayraktar et al. [100] describes an UAV system with an Hybrid model. Under this framework they develop a control system to guide UAVs in mapping and surveillance flights. In this work the authors also describe a formation control scheme and the flight trial results with two UAVs. The SMAVNET project proved the feasibility of safe formation flight up to 10 aircraft. The collision avoidance system is described by Leven et al. in [101].

Borrelli et al. [76] describe the distributed Receding Horizon Control (RHC) as a feasible solution even for a large number of aircrafts. It uses Mixed Integer Linear Programming (MILP) to ensure near optimum controllers [102, 103]. Waydo et al. [104] implemented and tested a formation control system based on receding horizon control (RHC) supervised by a high level controller. The control scheme was simulated and flight tested with the X-35A avionics mounted on an adapted T-33 jet trainer, following an F-15 fighter jet. The whole control protocol was provably safe. This was achieved as the aircraft dynamics, RHC dynamics, and the supervisor controller were described and specified with temporal logic, which allowed the authors to prove the desired properties. Stipanović et al. [70] describe the application of overlapping control to formation with constrained communications. Each vehicle only has access to the state of leader and vehicle in front of it. The method is decentralized taking advantage of the loose coupling induced by the communications. To compute the control variables, this method expands the interconnected system state dynamic equations using the inclusion principle, and defines the feedback control through the application of convex optimization.

Aircraft collision avoidance is a complex problem in normal flight as well as in formation flight. Most authors address each problem separately, aiming to regulate the formation aircraft relative position or to enforce a minimum distance among aircraft. Furthermore, most collision avoidance algorithms are not scalable, dealing only with the problem of a single pair of vehicles.

The formation stability problem was addressed by Swaroop et al. in [105]. It was proved that interconnected systems are stable in the vicinity of the equilibrium point if they are loosely coupled and the individual systems are exponentially stable. Parallel to the determination of performance benefits [91] NASA has been also studying the problem of string stability of their F/A-18 formation control system. Allen et al. [106] concluded that although the control system was string unstable it was applicable to a bounded number of aircraft as the position oscillation was limited by an acceptable range.

5.3.3 Current Approach

This work extends the formation Sliding Mode Controller (SMC) presented by Zheng et al. [97]. This study is constrained to the horizontal plane for simplicity. However, it is applicable to 3D formations with few adjustments. The main contributions are the inclusion of a collision avoidance scheme in the control system definition, the use of an extended unicycle dynamic model for the SMC derivation, and the incorporation of maneuvering restrictions on the sliding surfaces. The SMC derivation based on the extended unicycle model (section 2.4.2) creates a controller with dynamics adequate for fixed wing aircraft control. The SMC is defined for each aircraft and includes a set of two sliding surfaces for each of the other interacting aircraft (section 5.3.4). The sliding surfaces shapes and magnitudes restrict the maximum commanded velocities to the aircraft limits. Further on each set of sliding surfaces, one of those has a shape which drives the controller to command a maximum velocity in the opposite direction to other aircraft if they get to close.

In this work we distinguish static from dynamic formations. By static formations we mean formations where the vehicles' relative positions remain unchanged. The example of a static formation is one which maintains its frame constantly aligned with the Ground frame as described in section 5.3.9.1. In dynamic formations the vehicles may present relative velocities with respect to each other and the reference frame. A formation with its frame fixed to the leader or the path is dynamic, as the formation rotates with the leader or the path, inducing non-null inter-vehicle velocities (section 5.3.9.2).

In the solution presented by Zheng et al., there was a gain which controlled relative importance between the formation shape and each vehicle position error in the formation. The present solution has a similar gain which defines the relative importance between the formation shape and leader-relative position error, i.e., the deviation from each vehicle desired position relative to the leader (section 5.3.8).

We implemented some logic to force the collision avoidance dynamics, which is described by sections 5.3.5.2, 5.3.6.1, and 5.3.7.

This method was developed to be implemented on fixed wing Unmanned Air Vehicles (UAVs). The formation controller may be divided in two main modes: *Leader formation* and *Leaderless formation*. In the first method one of the team's aircraft is designated as the leader. This aircraft follows a predefined path or the flight commands defined by an operator. The follower aircraft are commanded by the formation controller to track a desired position relative to the leader. As the leader is not controlled by the formation controller, it does not try to avoid any of the followers. In the *Leaderless formation* method all aircraft are commanded by the formation controller to track a desired position relative to a virtual leader. This virtual leader is commanded to follow a predefined flight plan. Its dynamics are simulated by the 4 DOF model described in section 2.4.3. Because all real aircraft are controlled by the formation controller, they all act to avoid collisions. The leader, be it real or virtual, flies as if it were alone, defining the formation path through its speed and turn rate. Its speed and turn rate are limited to a range conducive to feasible paths by every aircraft in the formation. Currently, the method does not account for fixed obstacles or adversaries.

In terms of communications (section 5.3.10.1), this method requires that all interacting aircraft broadcast their horizontal position and velocity vectors, as well as their airspeed, roll, and yaw.

5.3.4 Sliding Mode Control

In this approach to formation flight we extend the formation Sliding Mode Controller presented by Zheng et al. [97]. In a formation with N UAVs, each UAV has $N - 1$ inter-UAV reference frames (fig. 5.20). Each inter-UAV reference frame is centered on UAV i CM, with the $\hat{\mathbf{x}}_{ij}$ axis pointing to UAV j's CM, and the $\hat{\mathbf{y}}_{ij}$ axis is rotated clockwise in the horizontal plane. ψ_{ij} is the deviation of $\hat{\mathbf{x}}_{ij}$ from the fixed frame $\hat{\mathbf{x}}_f$ axis. Two sets of sliding surfaces are implemented. Each sliding surface implements a different strategy on each inter-UAV frame axis (fig. 5.20). A *MaxVel* strategy is applied on the $\hat{\mathbf{y}}_{ij}$ axis to regulate the position error on that same axis. This strategy gets its name from the full speed command with large position errors. A CAS strategy is applied on the $\hat{\mathbf{x}}_{ij}$ axis for collision avoidance and to regulate the position error on that axis.

$$\hat{\mathbf{x}}_{ij} = [\cos(\psi_{ij})\ \sin(\psi_{ij})]^\intercal \tag{5.96a}$$

$$\hat{\mathbf{y}}_{ij} = [-\sin(\psi_{ij})\ \cos(\psi_{ij})]^\intercal \tag{5.96b}$$

$$\psi_{ij} = \tan^{-1}\left(\frac{y_j - y_i}{x_j - x_i}\right) \tag{5.96c}$$

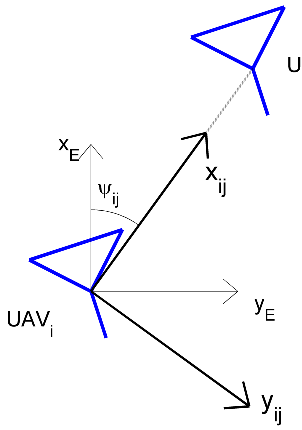


Figure 5.20: Inter-UAV relative frame

5.3.5 Formation Controller - Kinematic Derivation

5.3.5.1 MaxVel Sliding Surface

The *MaxVel* surface equation is presented here,

$$S = v + c_1 \cdot \operatorname{sgn}(x) - \frac{c_2}{x + c_3 \cdot \operatorname{sgn}(x)} \tag{5.97a}$$

$$c_1 = V_{max}$$

$$\dot{S} = \dot{v} + \frac{c_2 \cdot \dot{x}}{(x + c_3 \cdot \operatorname{sgn}(x))^2}. \tag{5.97b}$$

c_1 is set such that the commanded velocity doesn't exceed the maximum allowed velocity. The constants c_2 and c_3 are calibration variables and will be defined below.

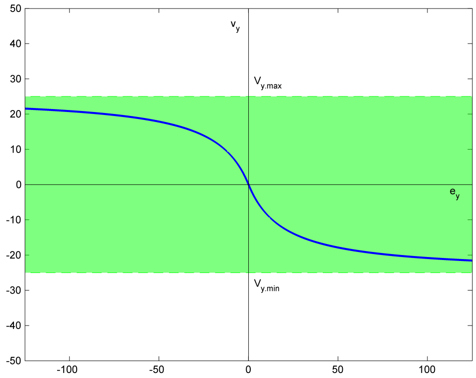


Figure 5.21: Standard position tracking sliding surface

5.3.5.2 Collision Avoidance Sliding Surface

On the $\hat{\mathbf{x}}_{ij}$ axis, which "connects" both UAVs, a repelling behavior is desired:

- If the UAVs are at a distance $\Delta x \leq r_{min}$, the velocity of each UAV should be maximum in the opposite direction relative to the other UAV, where r_{min} is the *safety distance*.
- If the UAVs are at a distance $\Delta x \leq r_d \Rightarrow v_x \in [-V_{max}, 0]$, where r_d is the relative desired distance. This means that the UAVs' relative distance should still increase.
- If the UAVs are at a distance $\Delta x \geq r_d \Rightarrow v_x \in [0, V_{max}]$, which means that the UAVs' relative distance should decrease. This also implies that the commanded velocity will not exceed V_{max} when the UAVs are far away.

We defined the sliding surface as an offset inverse function of velocity error *vs* position error. The desired behavior is implemented by the following sliding surface:

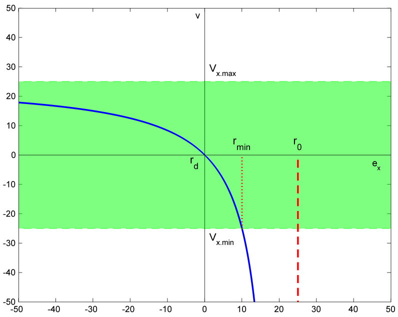


Figure 5.22: Collision avoidance sliding surface

$$S = v + c_1 - \frac{c_2}{x - c_3} \tag{5.98a}$$

$$c_1 = V_{max} \wedge c_2 = 2V_{max}\left(r_d - r_{min}\right) \wedge c_3 = 2\left(r_d - r_{min}\right)$$

$$\dot{S} = \dot{v} + \frac{c_2 \cdot \dot{x}}{\left(x - c_3\right)^2} \tag{5.98b}$$

5.3.5.3 Inter-vehicle Position Controller

The complete system behavior can be obtained by a weighted sum of the individual sliding surfaces projected on the Ground frame. This is defined by an $\hat{\mathbf{x}}_f$ axis pointing North and a $\hat{\mathbf{y}}_f$ axis pointing East, centered on a fixed point over the ground surface.

$$\mathbf{s}_{i,F} = \sum_{j \neq i}^{N} \mathbf{s}_{ij} \tag{5.99}$$

$$\begin{aligned} \mathbf{s}_{ij} = & \left[\dot{e}_{x,ij} + c_1 - \frac{c_2}{e_{x,ij} + c_3}\right] \hat{\mathbf{x}}_{ij} + \\ & + \left[\dot{e}_{y,ij} + c_1 \cdot \operatorname{sgn}\left(e_{y,ij}\right) - \frac{c_2}{e_{y,ij} + c_3 \cdot \operatorname{sgn}\left(e_{y,ij}\right)}\right] \hat{\mathbf{y}}_{ij} \end{aligned} \tag{5.100}$$

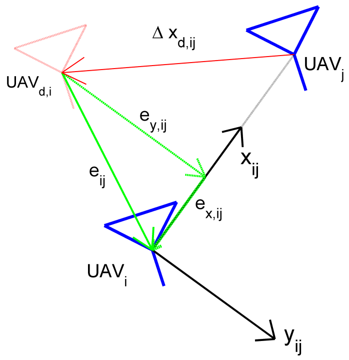


Figure 5.23: Position error vector with the projection on the inter-UAV frame.

where $e_{x,ij}$ and $e_{y,ij}$ are the position errors projected on the inter-UAV frame (fig. 5.23), as

$$\begin{aligned} \mathbf{e}_{ij} &= \mathbf{x}_i - \mathbf{x}_j - \Delta\mathbf{x}_{d,ij} && (5.101a)\\ e_{x,ij} &= \mathbf{e}_{ij} \cdot \hat{\mathbf{x}}_{ij} && (5.101b)\\ e_{y,ij} &= \mathbf{e}_{ij} \cdot \hat{\mathbf{y}}_{ij} && (5.101c)\\ \boldsymbol{R}_{\psi_{ij}} &= \begin{bmatrix} \hat{\mathbf{x}}_{ij}^{\intercal} \\ \hat{\mathbf{y}}_{ij}^{\intercal} \end{bmatrix} = \begin{bmatrix} \cos\left(\psi_{ij}\right) & \sin\left(\psi_{ij}\right) \\ -\sin\left(\psi_{ij}\right) & \cos\left(\psi_{ij}\right) \end{bmatrix} && (5.101d) \end{aligned}$$

$\boldsymbol{R}_{\psi_{ij}}$ is the rotation matrix from the Ground frame to the inter-UAV frame, $\Delta\mathbf{x}_{d,ij} = \mathbf{x}_{d,i} - \mathbf{x}_{d,j}$, and $\|\Delta\mathbf{x}_{d,ij}\| = r_d$. The formation desired positions for each vehicle i are defined by the system operator through the vector $\mathbf{x}_{d,i}^{\mathcal{F}} = [x_{i,\mathcal{F}}, y_{i,\mathcal{F}}]^{\intercal}$ in the formation reference frame (defined in sections 5.3.9.1 and 5.3.9.2).

With the definitions in (5.101) we may simplify (5.100) by merging $\dot{e}_{x,ij}\hat{\mathbf{x}}_{ij}$ and $\dot{e}_{y,ij}\hat{\mathbf{y}}_{ij}$ into $\dot{\mathbf{e}}_{ij}$, resulting in:

$$\begin{aligned} \mathbf{s}_{ij} = \dot{\mathbf{e}}_{ij} &+ \left[c_1 - \frac{c_2}{e_{x,ij} + c_3}\right]\hat{\mathbf{x}}_{ij} + \\ &+ \left[c_1 \cdot \operatorname{sgn}\left(e_{y,ij}\right) - \frac{c_2}{e_{y,ij} + c_3 \cdot \operatorname{sgn}\left(e_{y,ij}\right)}\right]\hat{\mathbf{y}}_{ij} \end{aligned} \qquad (5.102)$$

As the control vector has no direct effect in (5.99) we derive it:

$$\begin{aligned} \mathbf{s}_{i,F} &= \sum_{j \neq i}^{N} \left[\dot{\mathbf{e}}_{ij} + \left(c_1 - \frac{c_2}{e_{x,ij} + c_3}\right)\hat{\mathbf{x}}_{ij} + \right. \\ &\quad \left. + \left(c_1 \cdot \operatorname{sgn}\left(e_{y,ij}\right) - \frac{c_2}{e_{y,ij} + c_3 \cdot \operatorname{sgn}\left(e_{y,ij}\right)}\right)\hat{\mathbf{y}}_{ij}\right] && (5.103)\\ \dot{\mathbf{s}}_{i,F} &= \sum_{j \neq i}^{N} \left(\ddot{\mathbf{e}}_{ij} + \varepsilon_{ij}\right), && (5.104) \end{aligned}$$

where ε_{ij} is the pseudo error rate, defined as:

$$\begin{aligned}\varepsilon_{ij} := &\left(\frac{c_2 \cdot \dot{e}_{x,ij}}{\left(e_{x,ij}+c_3\right)^2} - c_1 \cdot \dot{\psi}_{ij} \cdot \operatorname{sgn}\left(e_{y,ij}\right) + \right.\\ &\left. + \frac{c_2 \cdot \dot{\psi}_{ij}}{e_{y,ij} + c_3 \cdot \operatorname{sgn}\left(e_{y,ij}\right)}\right) \hat{\mathbf{x}}_{ij} + \\ &+ \left(\frac{c_2 \cdot \dot{e}_{y,ij}}{\left(e_{y,ij}+c_3 \cdot \operatorname{sgn}\left(e_{y,ij}\right)\right)^2} + \right.\\ &\left. + c_1 \cdot \dot{\psi}_{ij} - \frac{c_2 \cdot \dot{\psi}_{ij}}{e_{x,ij}+c_3}\right) \hat{\mathbf{y}}_{ij},\end{aligned} \tag{5.105}$$

From equation (5.101) we can derive,

$$\dot{\mathbf{e}}_{ij} = \mathbf{v}_i - \mathbf{v}_j - \Delta \mathbf{v}_{d,ij} \tag{5.106a}$$

$$\ddot{\mathbf{e}}_{ij} = \mathbf{a}_i - \mathbf{a}_j - \Delta \mathbf{a}_{d,ij} \tag{5.106b}$$

$$\dot{e}_{x,ij} = \dot{\mathbf{e}}_{ij} \cdot \hat{\mathbf{x}}_{ij} + \mathbf{e}_{ij} \cdot \hat{\mathbf{y}}_{ij} \dot{\psi}_{ij} \tag{5.106c}$$

$$\dot{e}_{y,ij} = \dot{\mathbf{e}}_{ij} \cdot \hat{\mathbf{y}}_{ij} - \mathbf{e}_{ij} \cdot \hat{\mathbf{x}}_{ij} \dot{\psi}_{ij} \tag{5.106d}$$

$$\dot{\psi}_{ij} = \frac{\left(\mathbf{v}_j - \mathbf{v}_i\right) \cdot \hat{\mathbf{y}}_{ij}}{\left\|\boldsymbol{\Delta}\mathbf{x}_{ij}\right\|} \tag{5.106e}$$

The desired acceleration, by derivation of (2.14), is

$$\begin{aligned}\mathbf{a}_i &= \begin{bmatrix} \dot{V}_a \cdot \cos(\psi) - V_a \cdot \dot{\psi} \cdot \sin(\psi) + \dot{W}_x \\ \dot{V}_a \cdot \sin(\psi) + V_a \cdot \dot{\psi} \cdot \cos(\psi) + \dot{W}_y \end{bmatrix}_i = \ldots \\ \ldots &= \begin{bmatrix} \cos(\psi_i) & -V_{a.i} \cdot \sin(\psi_i) \\ \sin(\psi_i) & V_{a.i} \cdot \cos(\psi_i) \end{bmatrix} \begin{bmatrix} \dot{V}_{a.i} \\ \dot{\psi}_i \end{bmatrix} + \ldots \\ \ldots &+ \begin{bmatrix} \dot{W}_x \\ \dot{W}_y \end{bmatrix}_i = \mathbf{A}_i \nu_i + \dot{\mathbf{w}}_i,\ i = 1, \ldots, N,\end{aligned} \tag{5.107}$$

where ν_i is the ith aircraft control vector and $\dot{\mathbf{w}}_i$ is the ith aircraft local flow variation.

Merging equations (5.104) and (5.107) and rearranging, we obtain:

$$\dot{\mathbf{s}}_{i,F} = \sum_{j \neq i}^{N} \left(\boldsymbol{A}_i \nu_i + \dot{\mathbf{w}}_i - \boldsymbol{A}_j \nu_j - \dot{\mathbf{w}}_j - \Delta \mathbf{a}_{d,ij} + \varepsilon_{ij}\right). \tag{5.108}$$

We can select ν_i such that the known terms are canceled,

$$\tilde{\nu}_i = \sum_{j \neq i}^{N} \left(\boldsymbol{A}_i \nu_i - \boldsymbol{A}_j \nu_j - \Delta \mathbf{a}_{d,ij} + \varepsilon_{ij}\right) \tag{5.109a}$$

$$\nu_i = \frac{\boldsymbol{A}_i^{-1}}{N-1} \left[\sum_{j \neq i}^{N} \left(\boldsymbol{A}_j \nu_j + \Delta \mathbf{a}_{d,ij} - \varepsilon_{ij}\right) + \tilde{\nu}_i\right], \tag{5.109b}$$

where $\tilde{\nu}_i$ is the ith aircraft pseudo control vector, and yielding:

$$\dot{\mathbf{s}}_{i,F} = \tilde{\nu}_i + \sum_{j \neq i}^{N} \left(\dot{\mathbf{w}}_i - \dot{\mathbf{w}}_j\right). \tag{5.110}$$

To force the system to converge to the sliding surface we define a Lyapunov function candidate of the form $V = S^2$, yielding $\dot{V} = 2S\dot{S}$. To ensure formation stability about the desired positions, $\dot{V}$ needs to be negative definite, and so $S\dot{S} < 0$:

$$\mathbf{s}_{i,F} \cdot \dot{\mathbf{s}}_{i,F} = \mathbf{s}_{i,F} \cdot \left[\tilde{\nu}_i + \sum_{j \neq i}^{N} \left(\dot{\mathbf{w}}_i - \dot{\mathbf{w}}_j\right)\right]. \tag{5.111}$$

If the uncertainty of $\dot{\mathbf{w}}$ is bounded we can assume a worst-case scenario:

$$\|\dot{\mathbf{w}}_i\| \leq \dot{w}_{max},\ i = 1, ..., N \tag{5.112a}$$

$$\dot{\mathbf{w}}_i - \dot{\mathbf{w}}_j \leq 2\dot{w}_{max}\mathrm{sgn}\left(\mathbf{s}_{i,F}\right) \tag{5.112b}$$

$$\mathrm{sgn}\left(\mathbf{s}_{i,F}\right) = \frac{\mathbf{s}_{i,F}}{\|\mathbf{s}_{i,F}\|} \tag{5.112c}$$

$$\mathbf{s}_{i,F} \cdot \dot{\mathbf{s}}_{i,F} \leq \mathbf{s}_{i,F} \cdot \left[\tilde{\nu}_i + 2\left(N-1\right)\dot{w}_{max}\mathrm{sgn}\left(\mathbf{s}_{i,F}\right)\right]. \tag{5.112d}$$

As we need to assure $\mathbf{s}_{i,F} \cdot \dot{\mathbf{s}}_{i,F} < 0$, we force $\mathbf{s}_{i,F} \cdot \left[\tilde{\nu}_i + 2\left(N-1\right)\dot{w}_{max}\mathrm{sgn}\left(\mathbf{s}_{i,F}\right)\right] < 0$ with:

$$\tilde{\nu}_i = -\left[\Lambda \mathrm{sat}\left(\frac{\mathbf{s}_{i,F}}{\Phi}\right) + 2\left(N-1\right)\dot{w}_{max}\mathrm{sgn}\left(\mathbf{s}_{i,F}\right)\right], \tag{5.113}$$

where

$$\mathrm{sat}\left(\frac{\mathbf{s}_{i,F}}{\Phi}\right) = \begin{cases} \frac{\mathbf{s}_{i,F}}{\Phi} & \|\mathbf{s}_{i,F}\| < \Phi \\ \frac{\mathbf{s}_{i,F}}{\|\mathbf{s}_{i,F}\|} & otherwise \end{cases}. \tag{5.114}$$

$\mathbf{s}_{i,F} \cdot \dot{\mathbf{s}}_{i,F} \leq -\mathbf{s}_{i,F} \cdot \Lambda \mathrm{sat}\left(\frac{\mathbf{s}_{i,F}}{\Phi}\right) < 0$ is guaranteed if Λ is a positive definite matrix. The control vector is then,

$$\begin{aligned} \nu_i = \frac{\boldsymbol{A}_i^{-1}}{N-1} & \left[\sum_{j \neq i}^{N} \left(\boldsymbol{A}_j \nu_j + \Delta\mathbf{a}_{d,ij} - \varepsilon_{ij}\right) - \Lambda \mathrm{sat}\left(\frac{\mathbf{s}_{i,F}}{\Phi}\right) - \right. \\ & \left. -2\left(N-1\right)\dot{w}_{max}\mathrm{sgn}\left(\mathbf{s}_{i,F}\right)\right], \end{aligned} \tag{5.115}$$

where

$$\boldsymbol{A}_i^{-1} = \begin{bmatrix} \cos\left(\psi_i\right) & \sin\left(\psi_i\right) \\ -\frac{\sin(\psi_i)}{V_{a.i}} & \frac{\cos(\psi_i)}{V_{a.i}} \end{bmatrix} \tag{5.116}$$

5.3.6 Formation Controller - Corrections for Aircraft Dynamics

During the implementation and test of the strategies described in the previous section (5.3.5), we came to the conclusion that the system could not follow the idealized sliding surfaces. The reason for that was that the maximum achievable velocity differential between the controlled UAV and any of the other team UAVs was usually a lot smaller than the V_{max} which limited the sliding surfaces. The velocity differential on the $\hat{\mathbf{x}}_{ij}$ axis is $\Delta V_{x.ij} = (\mathbf{v}_i - \mathbf{v}_j) \cdot \hat{\mathbf{x}}_{ij}$ and is similar for the $\hat{\mathbf{y}}_{ij}$ axis. As such, the limit differential velocities are:

$$
\begin{aligned}
\Delta V_{x.ij.max} &= V_{i.max} - \mathbf{v}_j \cdot \hat{\mathbf{x}}_{ij} && (5.117a)\\
\Delta V_{x.ij.min} &= -V_{i.max} - \mathbf{v}_j \cdot \hat{\mathbf{x}}_{ij} && (5.117b)\\
\Delta V_{y.ij.max} &= V_{i.max} - \mathbf{v}_j \cdot \hat{\mathbf{y}}_{ij} && (5.117c)\\
\Delta V_{y.ij.min} &= -V_{i.max} - \mathbf{v}_j \cdot \hat{\mathbf{y}}_{ij}, && (5.117d)
\end{aligned}
$$

where $V_{i.max}$ is the maximum velocity UAV i can achieve.

Next we present the redefined *CAS* and *MaxVel* sliding surfaces, including the relative velocities and the acceleration constrains.

5.3.6.1 Collision Avoidance Sliding Surface

The *CAS* strategy is implemented on the $\hat{\mathbf{x}}_{ij}$ axis, which "connects" both UAVs. The following behavior is desired:

- If the UAVs are at a distance $|\Delta x| \leq r_{min}$, the velocity of each UAV should be maximum in the opposite direction relative to the other UAV. r_{min} is the *safety distance.*

- If the UAVs are at a distance $|\Delta x| \leq r_d \Rightarrow v_x \in [\Delta V_{x,min}, 0]$, where r_d is the relative desired distance. This means that the UAVs' relative distance should still increase.

- If the UAVs are at a distance $|\Delta x| \geq r_d \Rightarrow v_x \in [0, \Delta V_{x,max}]$, which means that the UAVs' relative distance should decrease. This also implies that the commanded relative velocity should be bounded by ΔV_{max} when the UAVs are far away.

- The desired acceleration and relative velocity should be feasible.

The maximum and minimum relative velocities in the $\hat{\mathbf{x}}_{ij}$ axis are:

$$
\begin{aligned}
\Delta V_{x,max,ij} &= V_{max,i} - \mathbf{v}_j \cdot \hat{\mathbf{x}}_{ij} && (5.118a)\\
\Delta V_{x,min,ij} &= -V_{max,i} - \mathbf{v}_j \cdot \hat{\mathbf{x}}_{ij}, && (5.118b)
\end{aligned}
$$

where $V_{max,i}$ is the UAV i maximum speed, and $\mathbf{v}_j$ is the UAV j velocity vector. The maximum acceleration in the $\hat{\mathbf{x}}_{ij}$ axis is:

$$
a_{x,max,ij} = |\hat{\mathbf{x}}_{B,i} \cdot \hat{\mathbf{x}}_{ij}|\, a_{Bx,max} + |\hat{\mathbf{y}}_{B,i} \cdot \hat{\mathbf{x}}_{ij}|\, a_{By,max}, \qquad (5.119)
$$

where $a_{(.),max}$ are the maximum accelerations in both UAV i's longitudinal ($\hat{\mathbf{x}}_{B,i}$) and lateral body axes ($\hat{\mathbf{y}}_{B,i}$).

We defined the sliding surface as an offset inverse function of velocity error *vs* position error (fig. 5.24). The desired behavior is implemented by the following sliding

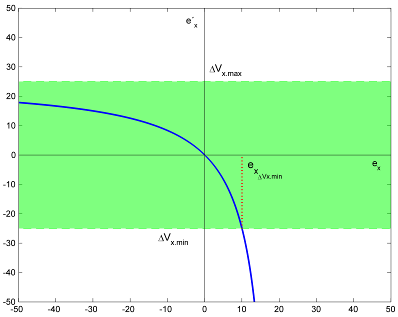


Figure 5.24: Collision avoidance sliding surface

surface:

$$S_{\hat{\mathbf{x}}} := \dot{e}_{\hat{\mathbf{x}}} - c_1 - \frac{c_1 c_2}{e_{\hat{\mathbf{x}}} - c_2} = \dot{e}_{\hat{\mathbf{x}}} - \frac{e_{\hat{\mathbf{x}}} c_1}{e_{\hat{\mathbf{x}}} - c_2} \tag{5.120a}$$

$$c_1 = \Delta V_{x,max,ij}$$

$$\dot{S}_{\hat{\mathbf{x}}} = \ddot{e}_{\hat{\mathbf{x}}} + \frac{c_1 c_2 \dot{e}_{\hat{\mathbf{x}}}}{\left(e_{\hat{\mathbf{x}}} - c_2\right)^2}. \tag{5.120b}$$

c_2 is defined by either an acceleration limit or a positional error limit:

$$c_2 = \max\left[\frac{4\left(1 + c_s\right) c_1^2}{27 a_{x,max,ij}}, -\frac{2 e_{x,\Delta V min,ij} V_{max,i}}{\Delta V_{x,min,ij}}\right], \tag{5.121}$$

where c_s is a safety factor and $e_{x,\Delta V min,ij}$ is the *proximity error margin*:

$$e_{x,\Delta V min,ij} := \left\{e_{x,ij} : \dot{e}_{x,ij} = V_{x,min,ij} \wedge S_{\hat{\mathbf{x}}} = 0\right\}, \tag{5.122}$$

5.3.6.2 MaxVel Sliding Surface

The *MaxVel* sliding surface is applied on the $\hat{\mathbf{x}}_{ij}$ axis. Its behavior should respect all the points defined above for the *CAS* strategy except for the first, i.e., the upper positional error limitation. We defined the sliding surface as a composition of two offset inverse functions of velocity error *vs* position error (fig. 5.25), one valid for positive positional errors and the other for negative positional errors.

The sliding surface equation is:

$$S_{\hat{\mathbf{y}}} := \dot{e}_{\hat{\mathbf{y}}} - \frac{e_{\hat{\mathbf{y}}} c_3}{e_{\hat{\mathbf{y}}} - c_4} \tag{5.123a}$$

$$c_3 = \begin{cases} V_{y,max,ij} & e_{\hat{\mathbf{y}}} < 0 \\ V_{y,min,ij} & otherwise \end{cases}$$

$$c_4 = \begin{cases} \frac{4(1+c_s)c_3^2}{27 a_{y,max,ij}} & e_{\hat{\mathbf{y}}} < 0 \\ -\frac{4(1+c_s)c_3^2}{27 a_{y,max,ij}} & otherwise \end{cases}$$

$$\dot{S}_{\hat{\mathbf{y}}} = \ddot{e}_{\hat{\mathbf{y}}} + \frac{c_3 c_4 \dot{e}_{\hat{\mathbf{y}}}}{\left(e_{\hat{\mathbf{y}}} - c_4\right)^2}. \tag{5.123b}$$

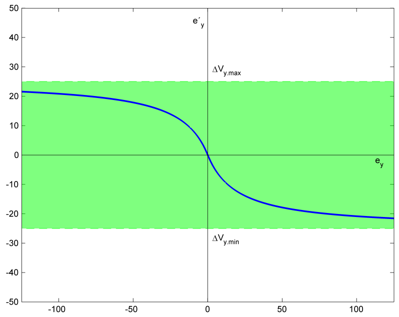


Figure 5.25: Standard position tracking sliding surface

The maximum and minimum relative velocities in the $\hat{\mathbf{y}}_{ij}$ axis are:

$$\Delta V_{y,max,ij} = V_{max,i} - \mathbf{v}_j \cdot \hat{\mathbf{y}}_{ij} \tag{5.124a}$$

$$\Delta V_{y,min,ij} = -V_{max,i} - \mathbf{v}_j \cdot \hat{\mathbf{y}}_{ij}. \tag{5.124b}$$

The maximum acceleration in the $\hat{\mathbf{y}}_{ij}$ axis is:

$$a_{y,max,ij} = \left|\hat{\mathbf{x}}_{B,i} \cdot \hat{\mathbf{y}}_{ij}\right| a_{Bx,max} + \left|\hat{\mathbf{y}}_{B,i} \cdot \hat{\mathbf{y}}_{ij}\right| a_{By,max}. \tag{5.125}$$

5.3.6.3 Inter-vehicle Position Controller

As in section 5.3.5.3 the inter-vehicle formation behavior is obtained by a weighted sum of the individual sliding surfaces projected on the Ground frame. With the adjusted sliding surface equations we now have:

$$\mathbf{s}_{i,F} = \sum_{j \neq i}^{N} \left[\dot{\mathbf{e}}_{ij} - \frac{e_{x,ij} c_1}{e_{x,ij} - c_2} \hat{\mathbf{x}}_{ij} - \frac{e_{y,ij} c_3}{e_{y,ij} - c_4} \hat{\mathbf{y}}_{ij} \right] \tag{5.126}$$

$$\dot{\mathbf{s}}_{i,F} = \sum_{j \neq i}^{N} \left(\ddot{\mathbf{e}}_{ij} + \varepsilon_{ij} \right), \tag{5.127}$$

with the pseudo error rate (ε_{ij}) defined as:

$$\varepsilon_{ij} := \left(\frac{c_1 c_2 \cdot \dot{e}_{x,ij}}{\left(e_{x,ij} - c_2\right)^2} + \frac{e_{y,ij} c_3}{e_{y,ij} - c_4} \dot{\psi}_{ij} \right) \hat{\mathbf{x}}_{ij} + \left(\frac{c_3 c_4 \cdot \dot{e}_{y,ij}}{\left(e_{y,ij} - c_4\right)^2} - \frac{e_{x,ij} c_1}{e_{x,ij} - c_2} \dot{\psi}_{ij} \right) \hat{\mathbf{y}}_{ij}. \tag{5.128}$$

5.3.7 Collision Avoidance Logic

The collision avoidance logic generates a deconflicting path, which drives conflicting UAVs to go around each other. This is activated if an UAV's path (to reach a

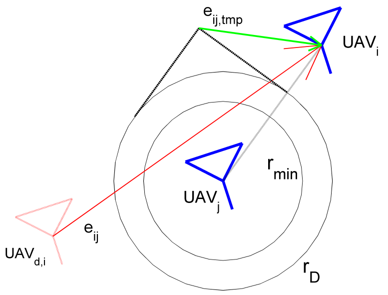


Figure 5.26: Collision avoidance logic. r_D and r_{min} are the deconfliction radius and *safety distance*, respectively. $e_{ij,temp}$ is the positional error to the temporary desired position, which changes with the UAVs' relative position.

desired position) intercepts the *Deconfliction Area* of another UAV. The *Deconfliction Area* is the area centered on an UAV position and bounded by the deconflition radius (r_D):

$$DeconfArea_j := \mathcal{B}_{r_D}(\mathbf{x}_j) = \left\{\mathbf{x} : \|\mathbf{x} - \mathbf{x}_j\| \leq r_D = r_{min} + c_2, \mathbf{x} \in \mathbb{R}^2\right\} \tag{5.129}$$

The collision avoidance logic is defined by:

$$e'_{x,ij} := \begin{cases} e_{x,ij} & e_{x,ij} \geq r_D - \|\Delta\mathbf{x}_{ij}\| \\ r_D - \|\Delta\mathbf{x}_{ij}\| & otherwise \end{cases} \tag{5.130a}$$

$$e'_{y,ij} := \begin{cases} e_{y,ij} & e_{x,ij} \geq r_D - \|\Delta\mathbf{x}_{ij}\| \vee \|e_{y,ij}\| > r_D \\ r_D \text{sgn}\left(e_{y,ij}\right) & otherwise \end{cases}, \tag{5.130b}$$

$e'_{x,ij}$ and $e'_{y,ij}$ are use instead of $e_{x,ij}$ and $e_{y,ij}$ in all calculations leading to (5.115).

5.3.8 Global Position Control

To drive the vehicles to a desired inertial position we implemented a *Leader formation* and a *Leaderless formation* scheme. The user may tune the shape-position balance, i.e., the balance between the formation shape stiffness and the regulation of the inertial position error.

The *Leader formation* scheme shape-position balance is controlled through an additional gain on the leader component, transforming (5.99) and (5.115):

$$\mathbf{s}_{i,F} := \left(1 + (N-1)\,k_L\right) \cdot \mathbf{s}_{il} + \sum_{j \neq i,l}^{N} \mathbf{s}_{ij} \tag{5.131a}$$

$$\begin{aligned} \nu_i = \frac{\boldsymbol{A}_i^{-1}}{1 + k_L} \Bigg[\left(\frac{1}{N-1} + k_L\right) \cdot \left(\boldsymbol{A}_l \nu_l + \Delta\mathbf{a}_{d,il} - \varepsilon_{il}\right) + \sum_{j \neq i,l}^{N} \frac{\left(\boldsymbol{A}_j \nu_j + \Delta\mathbf{a}_{d,ij} - \varepsilon_{ij}\right)}{N-1} - \\ - \Lambda \text{sat}\left(\frac{\mathbf{s}_{i,F}}{\Phi}\right) - 2\left(1 + k_L\right) \dot{w}_{max} \text{sgn}\left(\mathbf{s}_{i,F}\right) \Bigg], \end{aligned} \tag{5.131b}$$

where l is leader vehicle index and k_L the leader additional gain.

For the *Leaderless formation* scheme an additional pair of sliding surfaces needs to be added. These regulate directly the inertial position error. This error is defined with the desired inertial position computed from a virtual leader position. The sliding surfaces are both similar to the one described for the *MaxVel* strategy, as no collision avoidance with the virtual leader is needed. The main difference is that the projection axes are the inertial frame axes and $\mathbf{x}_{d,Lead}^{\mathcal{F}} = [0,0]^{\intercal}$, yielding:

$$\begin{aligned}
\mathbf{e}_i &:= \mathbf{x}_i - \mathbf{x}_{Lead} - \mathbf{x}_{d,i} && (5.132a)\\
\dot{\mathbf{e}}_i &:= \mathbf{v}_i - \mathbf{v}_{Lead} - \mathbf{v}_{d,i} && (5.132b)\\
\ddot{\mathbf{e}}_i &:= \mathbf{a}_i - \mathbf{a}_{Lead} - \mathbf{a}_{d,i} && (5.132c)\\
e_{x,i} &= \mathbf{e}_i \cdot \hat{\mathbf{x}}_{\mathcal{G}},\ e_{y,i} = \mathbf{e}_i \cdot \hat{\mathbf{y}}_{\mathcal{G}} && (5.132d)\\
\dot{e}_{x,i} &= \dot{\mathbf{e}}_i \hat{\mathbf{x}}_{\mathcal{G}},\ \dot{e}_{y,i} = \dot{\mathbf{e}}_i \hat{\mathbf{y}}_{\mathcal{G}} && (5.132e)
\end{aligned}$$

The sliding surface equation for inertial position regulator is:

$$\mathbf{s}_i := \left[\dot{e}_{x,i} - \frac{e_{x,i} c_5}{e_{x,i} - c_6}\right] \hat{\mathbf{x}}_{\mathcal{G}} + \left[\dot{e}_{y,i} - \frac{e_{y,i} c_7}{e_{y,i} - c_8}\right] \hat{\mathbf{y}}_{\mathcal{G}} \tag{5.133}$$

$$[c_5, c_6] = \begin{cases} \left[V_{x,max}, \frac{4(1+c_s)c_5^2}{27 a_{x,max}}\right] & e_{x,i} < 0 \\ \left[V_{x,min}, -\frac{4(1+c_s)c_5^2}{27 a_{x,max}}\right] & otherwise \end{cases}$$

$$[c_7, c_8] = \begin{cases} \left[V_{y,max}, \frac{4(1+c_s)c_7^2}{27 a_{y,max}}\right] & e_{y,i} < 0 \\ \left[V_{y,min}, -\frac{4(1+c_s)c_7^2}{27 a_{y,max}}\right] & otherwise \end{cases},$$

with the maximum and minimum relative velocities in the $\hat{\mathbf{x}}_{\mathcal{G}}$ and $\hat{\mathbf{y}}_{\mathcal{G}}$ axes:

$$\begin{aligned}
\Delta V_{x,max} &= V_{max,i} - V_{x,Lead} && (5.134a)\\
\Delta V_{x,min} &= -V_{max,i} - V_{x,Lead} && (5.134b)\\
\Delta V_{y,max} &= V_{max,i} - V_{y,Lead} && (5.134c)\\
\Delta V_{y,min} &= -V_{max,i} - V_{y,Lead}, && (5.134d)
\end{aligned}$$

where $V_{(.),Lead}$ are the virtual leader speeds on the inertial frame axes. The maximum feasible accelerations are:

$$\begin{aligned}
a_{x,max} &= \left|\hat{\mathbf{x}}_{B,i} \cdot \hat{\mathbf{x}}_{\mathcal{G}}\right| a_{Bx,max} + \left|\hat{\mathbf{y}}_{B,i} \cdot \hat{\mathbf{x}}_{\mathcal{G}}\right| a_{By,max} && (5.135a)\\
a_{y,max} &= \left|\hat{\mathbf{x}}_{B,i} \cdot \hat{\mathbf{y}}_{\mathcal{G}}\right| a_{Bx,max} + \left|\hat{\mathbf{y}}_{B,i} \cdot \hat{\mathbf{y}}_{\mathcal{G}}\right| a_{By,max}, && (5.135b)
\end{aligned}$$

Adding the virtual leader component (5.133) to (5.115), with a gain to tune the balance between the formation shape tracking and the desired inertial position tracking, yields:

$$\mathbf{s}_{i,F} := (N-1)\, k_L \mathbf{s}_i + \sum_{j \neq i}^{N} \mathbf{s}_{ij} \tag{5.136a}$$

$$\begin{aligned}
\nu_i = \frac{\boldsymbol{A}_i^{-1}}{1+k_L} \Bigg[& k_L \left(\mathbf{a}_{Lead} + \mathbf{a}_{d,i} - \varepsilon_i\right) + \sum_{j \neq i}^{N} \frac{\left(\boldsymbol{A}_j \nu_j + \Delta \mathbf{a}_{d,ij} - \varepsilon_{ij}\right)}{N-1} - \Lambda \text{sat}\left(\frac{\mathbf{s}_{i,F}}{\Phi}\right) - \\
& - (2 + k_L)\, \dot{w}_{max} \text{sgn}\left(\mathbf{s}_{i,F}\right) \Bigg], && (5.136b)
\end{aligned}$$

where k_L is the virtual leader gain and ε_i defined as in (5.128), but with the parameters of (5.133).

5.3.9 Formation Definition

Next we introduce several formation types. For each type we describe the definition of the desired positions, velocities, and accelerations of the formation vehicles.

5.3.9.1 Ground-aligned Formation Frame

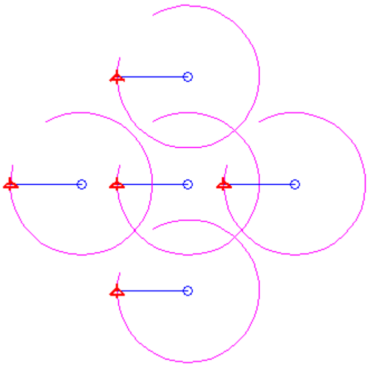


Figure 5.27: Ground-aligned formation frame. Cross shaped formation.

When the formation frame is permanently aligned to the Ground frame (fig. 5.27) the desired positions are defined by:

$$\mathbf{x}_{d,i}^{\mathcal{G}} = \mathbf{x}_{d,i}^{\mathcal{F}} - \mathbf{x}_{d,Lead}^{\mathcal{F}} + \mathbf{x}_{Lead}^{\mathcal{G}}, \tag{5.137}$$

where the superscripts $\mathcal{G}$ and $\mathcal{F}$ indicate if the vectors are described in the Ground or Formation frames, respectively. $\mathbf{x}_{Lead}$ and $\mathbf{x}_{d,Lead}$ are the leader current position and desired formation position. As the formation frame doesn't rotate, the desired relative positions remain constant and $\Delta\dot{\mathbf{x}}_{d,ij} = \Delta\ddot{\mathbf{x}}_{d,ij} = \dot{\psi}_{ij} = 0$. The equations in (5.106) can be simplified to

$$\dot{\mathbf{e}}_{ij} = \dot{\mathbf{x}}_i - \dot{\mathbf{x}}_j \tag{5.138a}$$

$$\ddot{\mathbf{e}}_{ij} = \ddot{\mathbf{x}}_i - \ddot{\mathbf{x}}_j \tag{5.138b}$$

$$\dot{e}_{x,ij} = \dot{\mathbf{e}}_{ij} \cdot \hat{\mathbf{x}}_{ij} \tag{5.138c}$$

$$\dot{e}_{y,ij} = \dot{\mathbf{e}}_{ij} \cdot \hat{\mathbf{y}}_{ij} \tag{5.138d}$$

This kind of formation produces the same path for every UAV with a constant positional offset. This may be useful to maintain a constant spatial sampling span over time.

5.3.9.2 Path-aligned Formation Frame

Most aircraft formations align their frame with the flight path. The most simple implementation maintains a constant formation shape, rotating it with the path direction (fig. 5.28). In these formations the shape coordinates defined by the operator

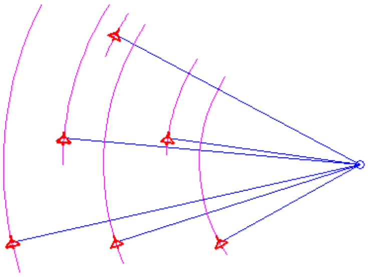


Figure 5.28: Path-aligned formation frame with fixed formation frame. Arrow shaped formation.

($\mathbf{x}_{d,i}^{\mathcal{F}}$) are rotated with the path heading and serve as inputs for the formation internal velocities and accelerations:

$$\mathbf{v}_{F.i}^{\mathcal{G}} = \boldsymbol{R}_{\psi_F}^{-1} \begin{bmatrix} -y_{i,\mathcal{F}} \\ x_{i,\mathcal{F}} \end{bmatrix} \dot{\psi}_F \tag{5.139a}$$

$$\mathbf{a}_{F.i}^{\mathcal{G}} = -\boldsymbol{R}_{\psi_F}^{-1} \mathbf{x}_{d,i}^{\mathcal{F}} \dot{\psi}_F^2 \tag{5.139b}$$

The formation may also adapt to the required kinematics instead of maintaining fixed formation positions, i.e., when the trajectory is curved the user defined Cartesian coordinates are transformed to polar coordinates (fig. 5.29). The alignments in x cease to be straight lines to become curves followed by the UAVs. In the same way, the y alignments are still straight lines, but not parallel to each other. All y alignment lines now cross the path's center of rotation. The user defined coordinates are transformed to new coordinates $\mathbf{x'}_{d,i}^{\mathcal{F}}$ in the formation reference frame. These are defined by three new polar coordinates: the formation radius of curvature r_F, defined in this case by the leader radius of curvature and its position on the formation, the UAV i radius of curvature r_i, and the UAV heading differential $\psi_{i.F}$,

$$r_L = \frac{V_{L.g}^2}{g \tan \phi_L} \tag{5.140a}$$

$$r_F = r_L + y_{L.F} \tag{5.140b}$$

$$r_i = r_F - y_{i,\mathcal{F}} \tag{5.140c}$$

$$\psi_{i.F} = \frac{x_{i,\mathcal{F}}}{r_F}. \tag{5.140d}$$

These result in the subsequent velocities and accelerations relative to the formation frame:

$$\mathbf{x'}_{d,i}^{\mathcal{F}} = \begin{bmatrix} r_i \sin \Delta\psi_{i.F} \\ y_{i,\mathcal{F}} + r_i \left(1 - \cos \Delta\psi_{i.F}\right) \end{bmatrix} \tag{5.141a}$$

$$\mathbf{v}_{Fi}^{\mathcal{G}} = \boldsymbol{R}_{\psi_F}^{-1} \begin{bmatrix} -y'_{i.F} \\ x'_{i.F} \end{bmatrix} \dot{\psi}_F \tag{5.141b}$$

$$\mathbf{a}_{Fi}^{\mathcal{G}} = -\boldsymbol{R}_{\psi_F}^{-1} \mathbf{x'}_{d,i}^{\mathcal{F}} \dot{\psi}_F^2. \tag{5.141c}$$

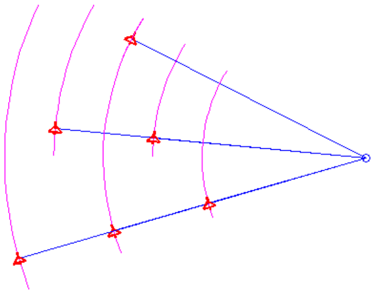


Figure 5.29: Path-aligned formation frame with adapting formation frame. Arrow shaped formation.

5.3.10 Distributed Controller

In order to have a reliable control system for close formation flight, every aircraft in the formation group should be able not only to compute its own controls, but also to know the leader state and to obtain the best state estimate of the other formation aircraft. If we had a perfect communication channel, the state update of any and every aircraft in the formation would be synchronized as soon as each aircraft had a state update. The reality of communications presents bandwidth constrains, latency, delays and information losses or corruption. If we have access to a reliable and fast enough communication channel, we can regard it as perfect. However, in our work we assume the communication channel is not perfect.

The communication issues led us to outline a system in which every aircraft is able to predict the group behavior with bounded and acceptable error margin, for a bounded communication latency or delay. To achieve this, we stated that every aircraft should be able to propagate a base reference, common to all other aircraft, and independently of the communications. We set this base reference to be the leader state. This is not possible with the *Leader formation* scheme, i.e., a real aircraft as the source for the leader state. That is because the base reference state may evolve arbitrarily/unpredictably, e.g., with wind disturbances, preventing the follower from predicting the leader state evolution, and obtaining a reliable leader state update only with the communication synchronizations. As such, we chose to use the *Leaderless formation* scheme, as the virtual leader state can be propagated in the same manner by all aircraft, maintaining a common reference.

5.3.10.1 Communications

The information needed for the formation controller may be divided in *State* updates and formation *Parameters*. The *Parameters* are set by the operator, and remain unchanged for long periods, when compared to the *State* updates. This means that their synchronization may be executed with low frequency. These parameters are:

$\mathbf{x}^{\mathcal{F}}_{d,i}$ - Desired positions in the formation (formation shape)

$Form_{type}$ - Formation type identifier (Ground-aligned, Path-aligned with fixed formation frame, or Path-aligned with adapting formation frame)

Λ - Convergence gains matrix

Φ - Sliding surface boundary layer parameter

k_L - Shape-Position balance gain

$\dot{w}_{max}$ - Wind maximum acceleration

$\phi_{max,L}$ - Leader maximum bank command (maneuvering constrain)

The *State* of UAV i is composed of:

$\mathbf{x}_i$ - Position

$\mathbf{v}_i$ - Velocity

ϕ_i, ψ_i - Roll and Yaw (heading) angles

$V_{a,i}$ - Airspeed

$\mathbf{w}_i$ - Average wind velocity

5.3.10.2 State Propagation

To cope with the communication latency and delays, each aircraft control computer propagates an estimated state of every formation aircraft. This propagation takes the last synchronized state (alg. 1, line 8) as the initial condition and applies the 4 DOF model (alg. 1, line 11) and the formation control method (alg. 1, line 15) to generate an estimate of the group state evolution. It is important that, between synchronizations, every aircraft state is propagated with the dynamic model (4 DOF), even the state of the aircraft where the computation happens. As such, the team state propagation will be similar on every aircraft. This also allows each aircraft to measure how far its state is from the estimate the other aircraft have, and implement safety/emergency procedures, if needed.

In the *Leaderless formation* mode, the virtual leader motion is simulated on each formation aircraft with the 4 DOF model. At the initial synchronization moment the formation flight plan is uploaded and activated on every aircraft, with a common initial state. From there on, each aircraft control computer propagates the virtual leader state, using the same 4 DOF model (alg. 1, line 3) and flight path control method (alg. 1, line 4), in order to generate an identical virtual leader path on every formation aircraft.

The formation controller computes the aircraft commands, taking as inputs the propagated virtual leader state and the most recent team state estimate (alg. 1, line 17).

5.3.11 Preliminary Simulation Results

5.3.11.1 Simulation Setup

To do the preliminary tests to the formation control method we used an extended unicycle aircraft model in a 3 DOF simulation (sec. 2.4.2). In these tests, the formations are composed of a leader aircraft and any number of followers. The simulated

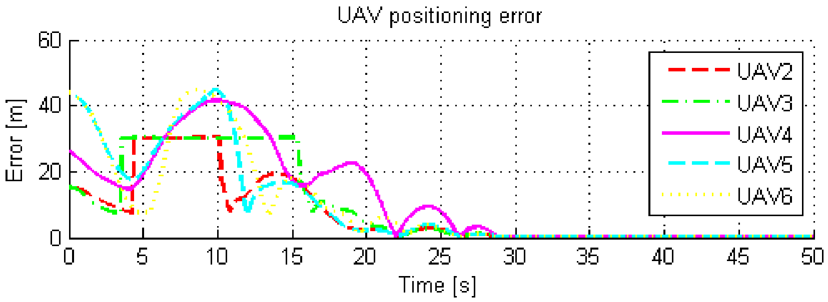
```
StateUpdate: $\left(\mathbf{X}_i(t), \mathbf{X}_L(t'), \mathbf{u}_L(t'), \hat{\mathbf{X}}_i(t'), \hat{\mathbf{u}}_i(t'), t, t', Sync, j, \forall i \in N\right)$
if $t - t' = \Delta t$ then
    //Virtual leader state propagation
    $\mathbf{X}_L(t) = \texttt{StateProp4DOF}(\mathbf{X}_L(t'), \mathbf{u}_L(t'), \Delta t)$;
    $\mathbf{u}_L(t) = \texttt{WPTracker}(\mathbf{u}_L(t))$;
    for $i = 1$ to $N$ do
        if $Sync$ then
            //State synchronization
            $\hat{\mathbf{X}}_i(t) = \mathbf{X}_i$;
        else
            //State propagation
            $\hat{\mathbf{X}}_i(t) = \texttt{StateProp4DOF}\left(\hat{\mathbf{X}}_i(t'), \hat{\mathbf{u}}_i(t'), \Delta t\right)$;
        end
    end
    for $i = 1$ to $N$ do
        $\hat{\mathbf{u}}_i(t) = \texttt{FormCtrl}\left(\hat{\mathbf{X}}_{i,\forall i \in N}(t), \mathbf{X}_L(t), \mathbf{u}_L(t)\right)$;
    end
    $\mathbf{u}_j(t) = \texttt{FormCtrl}\left(\mathbf{X}_j(t), \hat{\mathbf{X}}_{i,\forall i \in N, i \neq j}(t), \mathbf{X}_L(t), \mathbf{u}_L(t)\right)$;
end
return : $\mathbf{u}_j(t), \mathbf{X}_L(t), \hat{\mathbf{X}}_{i,\forall i \in N}(t), \mathbf{u}_L(t), \hat{\mathbf{u}}_{i,\forall i \in N}(t)$
```

Algorithm 1: Team state estimate update

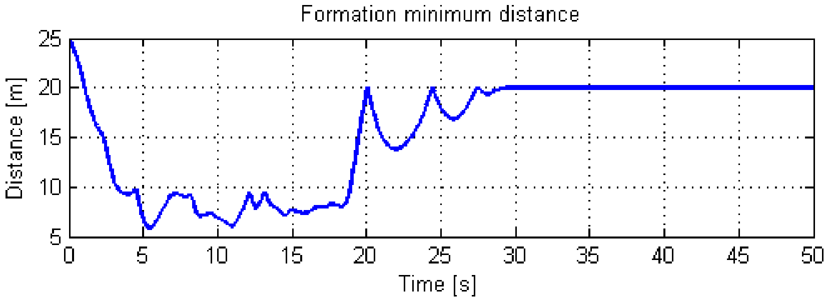
UAVs are restricted to a range of speeds between 18 and 25 meters per second, and to a maximum bank angle of 25 degrees. The controller parameters are set to control UAVs with a wingspan of 2.5 meters, with a *safety distance* of 5 meters. As the *safety distance* is set between the aircraft CMs, this would result in a distance of 2.5 meters between the aircraft wing tips.

We tested static and dynamic formations, with shapes like crosses, arrows, and in-line formations. We simulated several initial conditions. The simplest scenario featured every aircraft starting at their intended formation positions, to test the formation position tracking. In another scenario the aircraft started at positions symmetric relative to the desired ones. This tested the formation form-up feature and how well the system handled challenging initial aircraft positions.

The simulation incorporates perturbations as wind and positional errors. The wind is simulated with steady flow and gusts, i.e., air flow speed and direction variations about the steady flow. These variations are simulated by a Gauss-Markov Process (sec. 2.5.3). All the simulation results shown below include wind perturbations, with 16-knots of mean wind and gusts up to 8-knots. Every aircraft was subject to the same steady wind and completely independent gusts, to simulate a worst case scenario.



(a) Formation position error.



(b) Minimum distance among all aircraft in the formation.

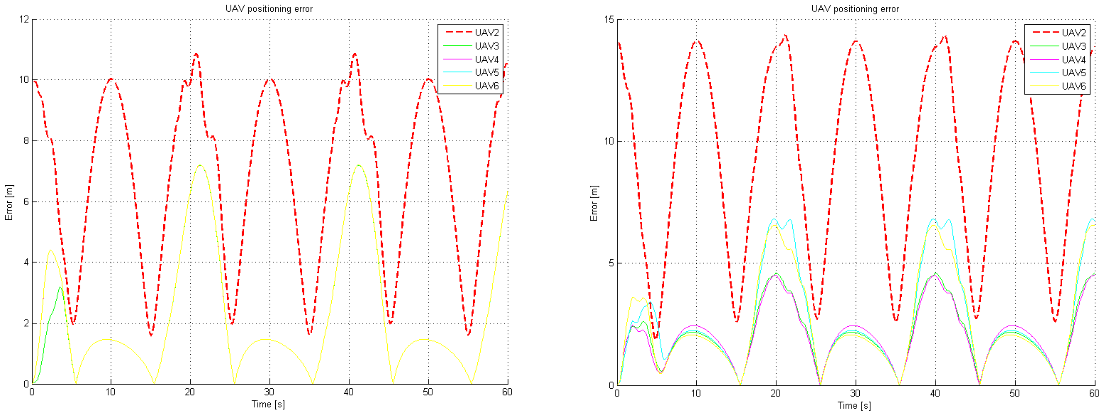
Figure 5.30: Simulation data where the follower aircraft start at a opposite positions to the desired ones relative to the leader

The positional perturbations were always applied to a single aircraft. We chose this to be the follower furthest ahead in the formation, to create a string perturbation on all the aircraft behind.

5.3.11.2 Results

The simulations show that the system performs well, even with wind and positional perturbations. The formations commanded had minimum distances among UAVs of 20 meters, which presented an additional challenge to the control system. The UAVs converge to the desired formation even if in the initial condition their positions are shuffled. Figure 5.30a shows the position convergence of a simulation where the follower aircraft started at symmetric positions relative to the ones desired around the leader, i.e., $\mathbf{x}_{Initial.Foll_i} - \mathbf{x}_{Initial.Lead} = \mathbf{x}_{d,Lead} - \mathbf{x}_{d,Foll_i}$. The illustration shows the aircraft settling in the desired formation positions after 30 seconds. Figure 5.30b illustrates the minimum distance evolution in the same simulation. For most of the time the minimum distance among UAVs is larger than 10 meters. It was driven below 10 meters only when the initial conditions had the aircraft shuffled. The minimum distance very rarely went below the *safety distance* of 5 meters, which only happened when the UAVs were started in very close positions and with conflicting headings.

After forming up the position error is kept low. It only grows slightly if the leader turns too tightly. As such, the minimum distance among the aircraft rarely decreases below one meter from the commanded. When the aircraft converge to their formation positions the average positional error is 0.76 meter with 0.41 meters of



(a) String stability - The UAVs were set on an *in line* formation.

(b) Mesh stability - The UAVs were set on an *arrow* shaped formation.

Figure 5.31: String and mesh stability - Attenuation of perturbations from UAV 2 to the other UAVs. UAV 2 had a sinusoidal motion perturbation. UAVs 3 to 6 show perturbations with reduced amplitude when compared to UAV 2.

standard deviation, even with independent air flow perturbations. If we take away the wind variation compensation (5.112), i.e., set $\dot{w}_{max} = 0m/s^2$, the positional error is still small, with an average of 1.27 meters and 1.40 meters of standard deviation.

The string and mesh stability about the desired formation positions was tested by the generation of step and sinusoidal positional perturbations. The results show that the motion perturbation is attenuated among the formation. Figure 5.31 shows the positional errors of all the follower UAVs with a sinusoidal perturbation of UAV 2's position. UAV 2 presents the largest positional errors, demonstrating good string perturbation attenuation. Figure 5.31a shows the results of an in line sinusoidal perturbation on an *in line* formation, demonstrating string stability. Figure 5.31b shows the results of a diagonal sinusoidal perturbation on an "arrow" formation, demonstrating mesh stability.

Figure 5.43 (at the end of the chapter) illustrates the positional evolution of an *arrow* formation during the *form-up* phase.

5.3.12 Simulation Results - Leaderless Formation

After the results of the preliminary tests we decided that the safest approach for the real world implementation would be to use the *Leaderless Formation* method. That way all UAVs would try to avoid each other and there would not be a special sensitivity to a single UAV state data.

The subsequent tests incorporated the corrections to account for the aircraft dynamics constrains, described in section 5.3.6. The simulation realism was also increased, starting with the 3 DOF simulation (sec. 2.4.2) and then testing the system with a 4 DOF simulation (sec. 2.4.3) and a 6 DOF simulation (sec. 2.4.7).

We now describe formation the controller tuning process and its results. To conclude the simulation results, we analyze the formation controller performance under several environment perturbations and initial conditions.

5.3.12.1 Controller Tuning

After the preliminary tuning of the upgraded controller with the 3 and 4 DOF simulations, we focused on fine tuning the formation control parameters with a more realistic 6 DOF simulation. This was executed in the same setup created for the initial tests with real UAVs (sec. 5.3.13). Due to some issues with the data dissemination latency we simulated one UAV with the Piccolo 6 DOF simulator [107, 108] and the other with the 4 DOF simulator. With this setup we have a more realistic data flow behavior, with Comunications delays and errors, as well as realistic dynamics for the one UAV. This UAV is the focus of the controller tuning analysis.

The first formation control parameters to be tuned were the tracking parameters, i.e., the tracking gain matrix Δ and the saturation boundary layer thickness Φ, in (5.136b). To have homogeneous control over the North and East axes we make

$$\Delta = k_P \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{5.142}$$

where k_P is the tracking gain. To tune k_P and Φ, we reduced the formation to a single UAV, tracking the virtual leader. We tested the tracking performance with weak wind (2 m/s), strong wind (8 m/s), and weak wind (2 m/s) with position reference disturbances. The formation is commanded to follow a circular trajectory with a 250m radius. The position reference disturbances were first order responses to a 100m impulse (fig. 5.32).

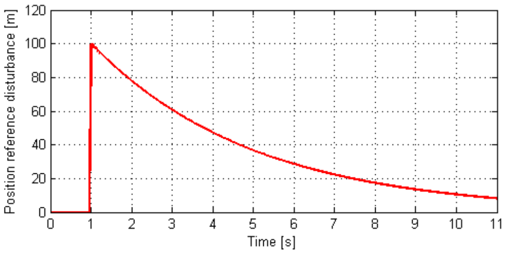
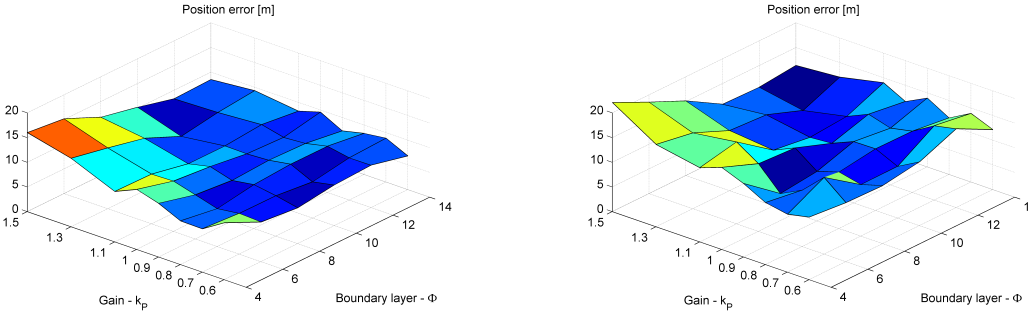


Figure 5.32: Position reference disturbance of 100 meters.

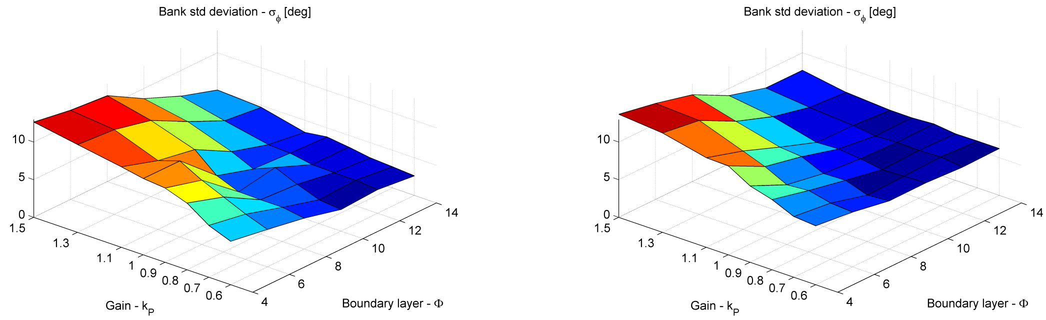


(a) Performance with weak wind (2 m/s).

(b) Performance with strong wind (8 m/s).

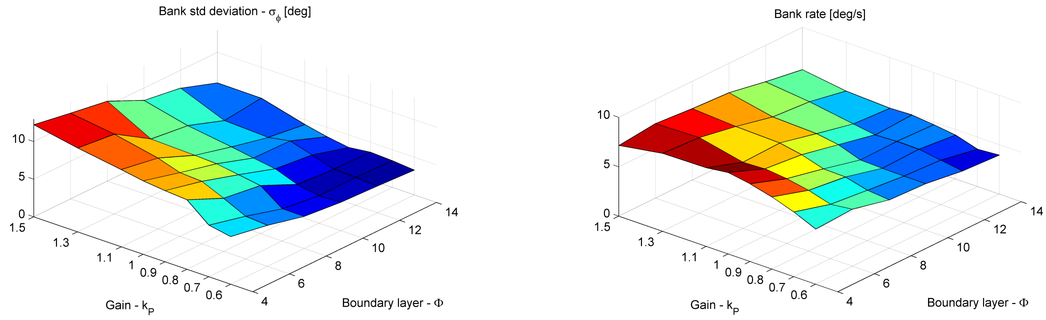
Figure 5.33: Tracking gain and boundary layer effect on the position tracking performance with different wind conditions.

To tune the tracking parameters we evaluated the performance through the position error and the control load. The control load was evaluated from the bank variation and the mean bank rate. Figure 5.33 shows that the controller tracks well in both wind conditions with $\Delta \in [0.8, 1.1]$ and $\Phi \in [6, 12]$. For stronger winds, lower tracking gains and larger boundary layers yield poorer results, due to the lack of control authority (fig. 5.33b). Higher tracking gains and tighter boundary layers lead to instability, no matter the wind conditions, which also reduce the tracking performance (fig. 5.33). Figure 5.34 shows that the control load increases with the tracking gain and with tighter boundary layers, as expected. The controller behaves similarly when it is subjected to position reference disturbances (fig. 5.34). From the illustrated performance (fig. 5.33 and 5.34) we may conclude that a good setting has:



(a) Bank variation with weak wind (2 m/s).

(b) Bank variation with strong wind (8 m/s).



(c) Bank variation with weak wind (2 m/s) and position reference disturbances (100 m).

(d) Mean bank rate with strong wind (8 m/s).

Figure 5.34: Tracking gain and boundary layer effect on the control load with different wind conditions and position reference disturbances.

$$\Delta \in [0.8, 1] \tag{5.143a}$$

$$\Phi \in [10, 12]\,. \tag{5.143b}$$

After the tracking parameters we tuned the leader gain k_L, in (5.136b). This gain defines the relative importance of the inertial position regulation in the control equation regarding the formation shape, i.e., the balance between the regulation of the global position error and the position error relative to the other formation UAVs.

To test the leader gain we used a 5 UAV ground-aligned formation with a cross shape (fig. 5.35). The formation is commanded to follow a circular trajectory with a

250m radius. The test formations set the closest UAVs at distances of 80 m and 50 m, testing the system behavior in loose and tight formations. As before, we tested the controller performance with weak wind (2 m/s), with strong wind (8 m/s) and with position reference disturbances.

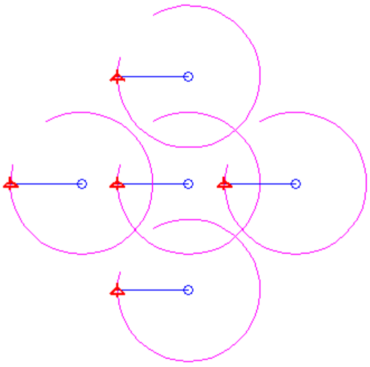
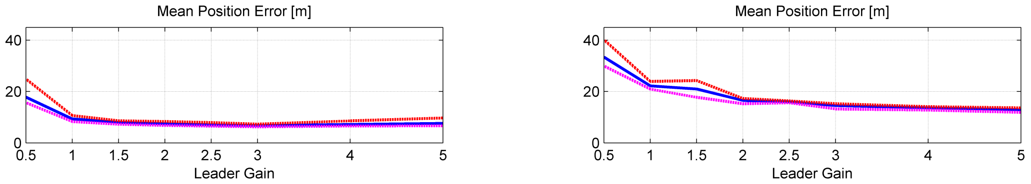


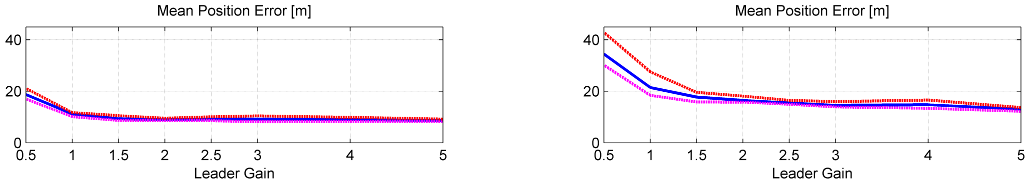
Figure 5.35: Ground-aligned formation with a cross shape.

The formation dynamics become more coupled with lower leader gains. This means that higher leader gains lead to formations that present better string or mesh stability [89]. That effect is apparent in figure 5.36, where lower leader gains yield poorer tracking performance. The control load, reflected in the bank variation (fig. 5.37), is only very mildly affected by this gain. The minimum distance reached by any UAV pair, over the simulation runs, suggests that a good k_L is around 2.5 (fig. 5.38)



(a) Tracking performance with weak wind (2 m/s). Minimum commanded distance among formation UAVs of 80 m.

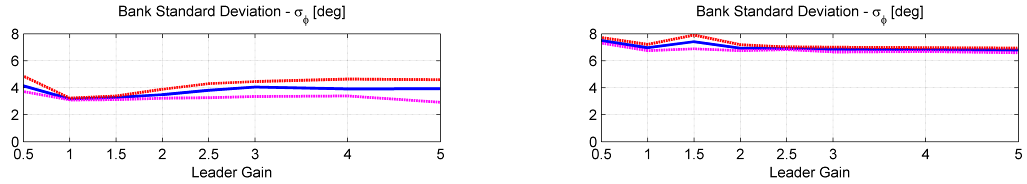
(b) Tracking performance with strong wind (8 m/s). Minimum commanded distance among formation UAVs of 80 m.



(c) Tracking performance with weak wind (2 m/s) and position reference disturbances (20 m). Minimum commanded distance among formation UAVs of 80 m.

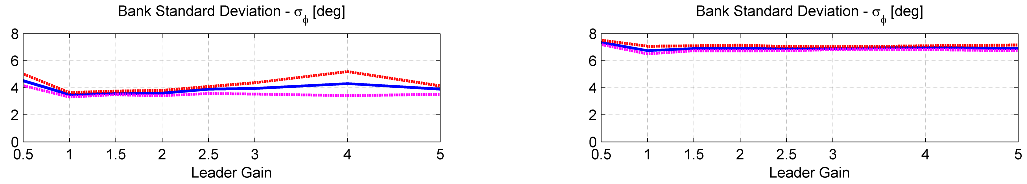
(d) Tracking performance with strong wind (8 m/s) in a tighter performance. Minimum commanded distance among formation UAVs of 50 m.

Figure 5.36: Leader gain effect on the position tracking performance with different wind conditions and position reference disturbances. The plots show the mean (blue), the maximum (red), and the minimum (pink) of several simulations average tracking error for each of the controller gain settings.



(a) Bank variation with weak wind (2 m/s). Minimum commanded distance among formation UAVs of 80 m.

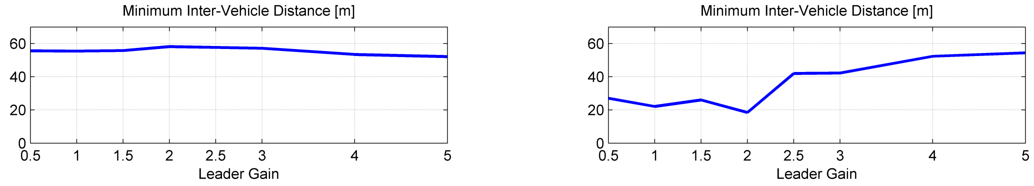
(b) Bank variation with strong wind (8 m/s). Minimum commanded distance among formation UAVs of 80 m.



(c) Bank variation with weak wind (2 m/s) and position reference disturbances (20 m). Minimum commanded distance among formation UAVs of 80 m.

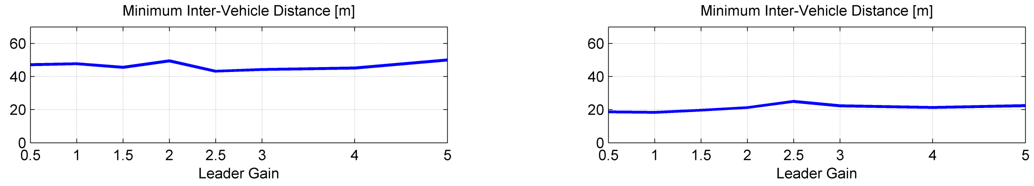
(d) Bank variation with strong wind (8 m/s) in a tighter performance. Minimum commanded distance among formation UAVs of 50 m.

Figure 5.37: Leader gain effect on the bank variation with different wind conditions and position reference disturbances. The plots show the mean (blue), the maximum (red), and the minimum (pink) of several simulations bank standard deviation for each of the controller gain settings.



(a) Overall minimum distance with weak wind (2 m/s). Minimum commanded distance among formation UAVs of 80 m.

(b) Overall minimum distance with strong wind (8 m/s). Minimum commanded distance among formation UAVs of 80 m.



(c) Overall minimum distance with weak wind (2 m/s) and position reference disturbances (20 m). Minimum commanded distance among formation UAVs of 80 m.

(d) Overall minimum distance with strong wind (8 m/s) in a tighter formation. Minimum commanded distance among formation UAVs of 50 m.

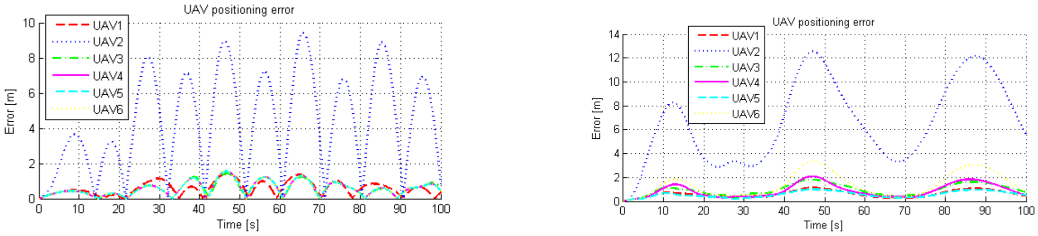
Figure 5.38: Leader gain effect on the overall minimum attained distance, with different wind conditions and position reference disturbances.

5.3.12.2 Performance Evaluation

The upgraded controller performance was tested with a 4 DOF simulation. To analyze the controller performance we needed a dataset that was representative of the distribution of the performance measurements. To generate that dataset we ran a very large number of simulations, automatically sweeping several environment conditions. The feasibility of the large number of simulation runs lead us to choose the 4 DOF simulation for the performance tests, instead of the 6 DOF simulation, that didn't allow us to automate the environment conditions sweep. Even so, the performance results are meaningful, as we used the same formation controller parameters and the formation behavior is similar in both simulations. The main difference between both simulations is the position error magnitude, which is about twice in the 6 DOF simulation when compared with the 4 DOF simulation results, in the same conditions.

As for the previous tests (sec. 5.3.11), the controller parameters are set to control UAVs with a wingspan of 2.5 meters, with a *safety distance* of 5 meters, resulting in a safety distance between the aircraft wing tips of 2.5 meters. The test conditions were the same as for the preliminary tests, yielding promising results.

The simplest scenario featured every aircraft starting at their intended formation positions. This tested the formation position tracking with perturbations such as wind and positional errors. The positional perturbations were intended to test the string and mesh stability of the formation control. We applied sinusoidal perturbations to a single aircraft and observed the reaction of the other formation aircraft. The formation aircraft show a positional perturbation strongly attenuated when compared with the one shown by the directly perturbed aircraft. Figure 5.39 shows an attenuation to less than *1/3* of the original perturbation for every formation aircraft. Further, the system shows no sign of perturbation build up over time. The formation aircraft perturbation remains proportional to the directly perturbed aircraft over time.

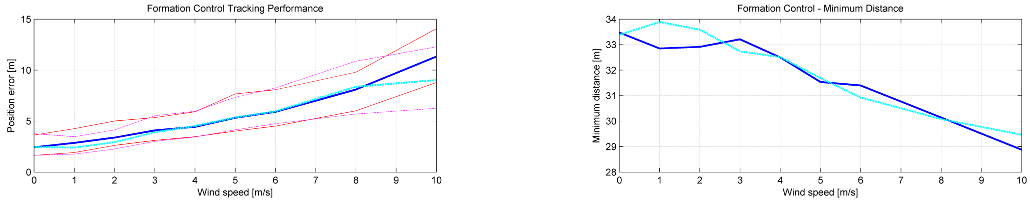


(a) String stability - The UAVs were set on an *in-line* formation.

(b) Mesh stability - The UAVs were set on an *arrow* shaped formation.

Figure 5.39: String and mesh stability - Attenuation of perturbations from UAV 2 to the other UAVs. UAV 2 presented a sinusoidal motion perturbation. UAVs 3 to 6 show perturbations with reduced amplitude when compared to UAV 2.

To test the system's accommodation of wind perturbations we simulated a range of wind conditions. This tests the system in steady state, i.e., with UAV positions starting and remaining close to the reference ones. In each simulation every aircraft was subject to the same steady wind and completely independent gusts, to simulate a worst case scenario. We simulated strong turbulence, generating several gusts with more than 50% of simulated steady wind speed. We tested the formation control over

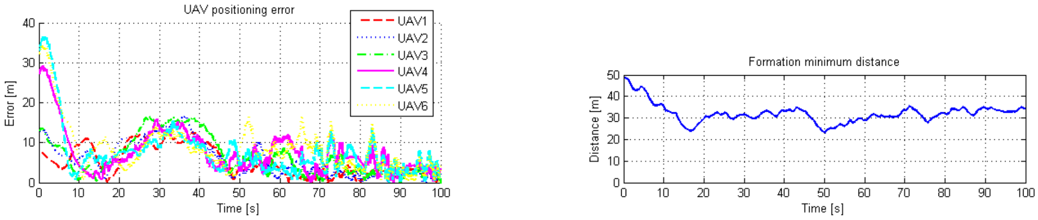


(a) Formation position mean tracking error. The mean tracking error was evaluated for each formation UAV from several simulation runs. The thicker curve represents the average among all UAVs. The top and bottom thinner curves are the mean tracking error of the worst and best performing UAVs.

(b) Minimum distance among UAVs. UAVs are kept safely apart even for strong winds.

Figure 5.40: Wind compensation by the formation controller. The solid curves represent the performance of the formation controlled without the wind variation compensation (5.112) and the dashed lines with it. The formation error increases with wind, however it is small even for strong winds.

several simulation runs with each test wind speed (fig. 5.40). The formation control was tested with and without the wind variation compensation (5.112), allowing us to evaluate its benefits. The results show that the wind variation compensation seems to yield better results only for the very strong winds. Moreover, it generates more variable control references, resulting in a wobbly flight. This leads to the conclusion that the system behaves better without the wind variation compensation. In general, the tests show that the system compensates for the wind perturbations well, maintaining the mean position error below 15 meters even for strong winds. Maybe even more important are the results in terms of minimum distance maintained among all UAVs. Figure 5.40b shows that, in steady state, the formation controller is able to maintain a distance among all aircraft quite close to the commanded and well above the *safety distance*.



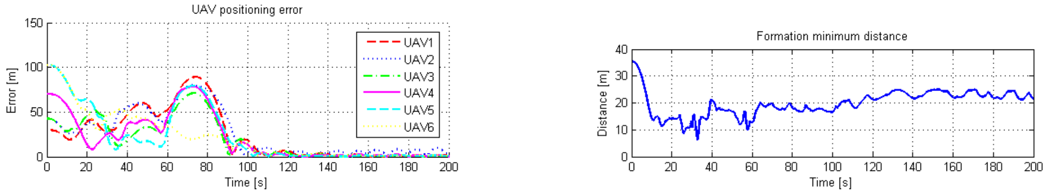
(a) UAVs' position error - The initial convergence and the final settling around the commanded formation positions is shown clearly.

(b) Minimum distance among all formation aircraft - The minimum distance does not go below 20 meters and converges to the intended one, 35 meters in this case.

Figure 5.41: Formation convergence - UAVs start in positions defined by an enlarged version of the commanded formation. The strong simulated wind (16 kts) leads to an overshoot after the initial convergence.

We tested the system convergence by initializing every aircraft in an enlarged version of the commanded formation. Figure 5.41 illustrates the correct convergence of the formation. After the initial convergence there is a momentary divergence due to the fact that the simulation featured a 16 kts wind, which lead to an initial overshoot. The strong wind speed is also the reason for the final jittery position error.

To test the form-up control, we also tested a scenario where the aircraft started on positions symmetric relative to the desired ones (fig. 5.43a). This tested how well the system handled challenging initial aircraft positions. The system behaves as expected, driving all aircraft to their intended formation positions while keeping a safety distance when the UAVs need to cross paths (fig. 5.42).



(a) UAVs' position error - Initially the system converges. When the UAVs get closer they start to interact in order to reposition themselves. The repositioning interaction leads some aircraft to diverge before the final convergence, when all are close to their intended relative directions.

(b) Minimum distance among all formation aircraft - At the end of the initial convergence the UAVs are quite close to each other, but never get closer than 5 meters. In fact they rarely get closer than 10 meters.

Figure 5.42: Formation convergence - UAVs start in positions defined by an enlarged version of the commanded formation with their positions symmetric to the intended ones on both formation axes (fig. 5.43a).

5.3.13 Flight Results

At the moment this thesis is being written, we initiated the formation controller real flight tests. For now we tested a formation with only one real aircraft and the rest simulated. These tests served to check that the formation followed the intended trajectory, observe the system's gust compensation, and reveal some points that could improve the controller performance.

The tests were executed with a testbed developed in the PITVANT project [109]. The testbed implements a ground control, i.e., the controller runs on MatLab on a laptop that receives telemetry and sends flight commands through a ground-station. The autopilot system is a Piccolo II [110], which receives airspeed and bank commands from Dune, a control software developed at the Underwater Systems and Technology Laboratory from University of Porto. Formation trajectory, i.e., the virtual leader trajectory, was controlled by path following controller developed by Oliveira and Encarnação [111, 112].

The best results yet presented a mean position error of 15.9 m, with an average wind speed of 3m/s. The gain and boundary layer thickness that yielded the best performance were:

$$\Delta = 0.9 \tag{5.144a}$$

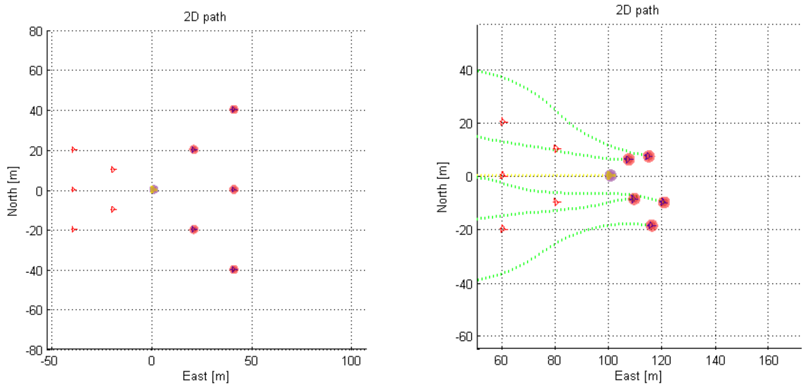
$$\Phi = 10, \tag{5.144b}$$

which fall in the tuning intervals predicted by the simulations (5.143). These results are very good if we take into account that the controller was running off-board and had to cope with communication delays that were worse than those present in the 6 DOF simulation. The transition of the controller to C++ and an on-board implementation should also improve the system performance.

5.4 Conclusions

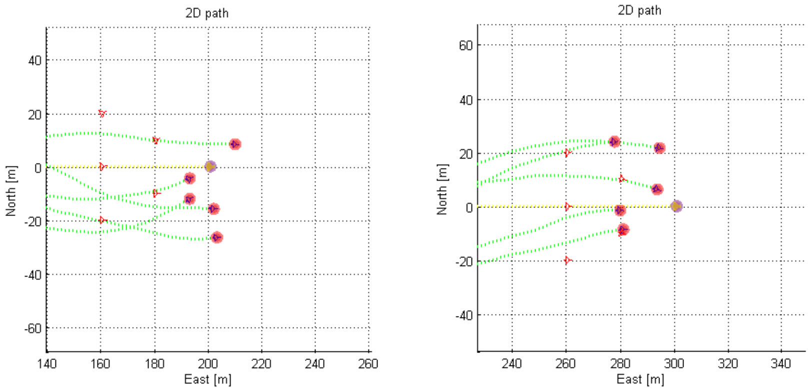
We developed a Collision Avoidance System for indoor helicopters and a formation flight controller with collision avoidance for fixed wing aircraft. The formation control system handles the air flow perturbations and the formation vehicles' positional perturbations well. The individual vehicle positional error is attenuated over the formation, showing good string and mesh stability. Furthermore, the controller handles every phase of the formation maneuver including the initial position convergence (form-up) and the formation shape tracking. The controller includes a collision avoidance feature which allows a safe form-up. Further, this feature allows a safe control of close formations, as shown by the results. During tests, the algorithm was setup to control UAVs with wingspan of approximately 2.5 meters. The results show that with such setup the UAVs very rarely come closer than 10-meters and never closer than the 5-meters *safety distance*. This means that the formation controller can support the acquisition of spatially distributed samples of airflow data simultaneously, by flying an array of UAVs. The controller seems to be capable of maintaining the UAVs close enough that several of them are inside a thermal simultaneously.

The Collision Avoidance System for helicopters was developed to gain experience with this kind of systems. We created a system featuring a Path Deconflicting Algorithm and Close Proximity Collision Avoidance Algorithm so that the system would have the benefits of both methods, i.e., path smoothness and better time efficiency provided by the PDA and safety enforcement provided by the CPCAA. We proved that the system is safe and able to reach its goals, with some of the presented dynamic models. One of the lessons learned was that a larger number of switching conditions in the controller introduces added complexity to prove the safety and liveliness properties.



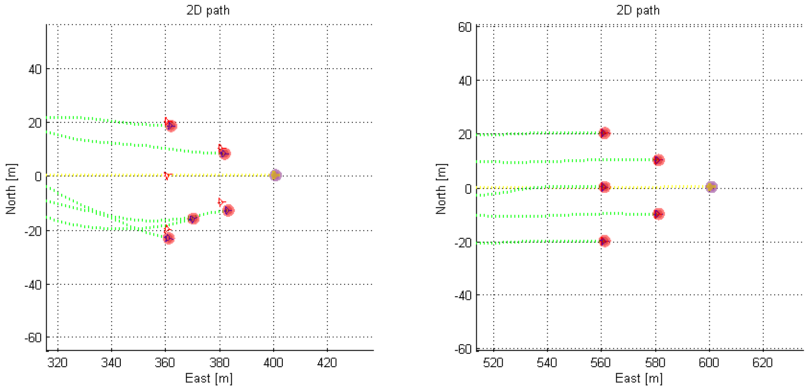
(a) Aircraft initial positions.

(b) Aircraft positions after 5 seconds.



(c) Aircraft positions after 10 seconds.

(d) Aircraft positions after 15 seconds.



(e) Aircraft positions after 20 seconds.

(f) Aircraft positions after 30 seconds.

Figure 5.43: Formation form-up illustration sequence. The follower aircraft start at symmetric positions to the desired ones relative to the leader, both in the x and y axes. The desired formation positions are shown as thin red aircraft. The real current aircraft positions are the blue arrows inside the circles. The circles represent the collision and safety zones of each vehicle. In this simulation the leader (clear aircraft) is always moving eastward at 20 m/s.

CHAPTER VI

Conclusions

6.1 Conclusions

This work studies the possibility to extend aircraft flight endurance through the exploitation of flow field energy. To exploit the flow field energy the aircraft needs to have enough maneuverability, and a suitable exploitation controller with an estimator for the phenomenon.

In chapter III we discussed the necessary conditions to perform perpetual flight by harvesting energy from the flow field phenomena: thermals, wind shear and gusts. We concluded that with every phenomena the aircraft efficiency and the exploited phenomenon strength are crucial to enable perpetual flight. For thermals the conditional parameters are the mean spacing among thermals, the mean thermal lifespan, the day maximum of mean updraft speed, the aircraft minimum sink rate, and the aircraft best lift over drag ratio. For wind shear the conditional parameters are the average wind shear gradient, the air density, the aircraft minimum drag ratio, the aircraft best lift over drag ratio, and the aircraft wing loading. For gusts the conditional parameters are the gusts' magnitude and wavelength, and the aircraft efficiency.

We also addressed the problem of estimating the flow field phenomena parameters, such that these can be used by an exploitation controller. We showed thermals and wind are observable, and that thermals and wind shear parameters can be estimated. The developed estimators provide independent estimation capabilities, i.e., without prior knowledge on the flight area characteristics or the phenomena's prior states, but may also incorporate these data. That is because the estimators are Particle Filter (PF) estimators, which are capable of global estimation, i.e., estimation with an unknown initial state. Further, the Particle Filters may use and represent any probability distribution, allowing the representation of non-Gaussian probability distributions, displayed by many of the phenomena representative parameters, the representation of multiple phenomena, or even the incorporation of prior information. The implemented estimators, a *Chimney Thermal* Regularized Adaptive Particle Filter (RAPF) locator [113] and a *Surface Wind Shear* PF estimator [114], perform well in simulation. The results are very good both in terms of estimation accuracy, and in terms of processing load. The processing load requirements are compatible with small computer capabilities for use in small UAVs, as intended.

To support the estimator development and the simulation of UAV flight through the studied flow field phenomena we studied several existing models. Further, we improved some of those models and created new ones to better capture some important features. The used models represent the phenomena's 3D shape, the interaction with the steady wind flow, and the dynamics. For thermals we presented 2D models, but more importantly models for *Chimney* and *Bubble Thermals*. The *Bubble Thermal* model is new and is based on the vortex shell hypothesis by Cone [45]. The extended *Chimney Thermal* model is based on the Allen model [4], and includes the thermal core movement and thermal interactions with the surrounding flow field, e.g., its leeward leaning when there is wind. In terms of wind shear we presented an existing model for the *Surface Wind Shear*, and new models for the *Layer* and the *Ridge Wind Shear*.

Most of the presented models lack the verification of how well they represent the reality. As such, we felt the need to get airflow data that was representative of the flow field phenomena and could be compared with the models' predictions, validating or rejecting them. As UAVs can be used to measure the airflow through which they are moving, they may be regarded as tools to collect the needed validation airflow data. Therefore, we developed a flight controller which enables the collection of airflow data at several distributed spatial points simultaneously. It is a formation flight controller which features a collision avoidance logic [115], allowing safe flight of several UAVs in and around the phenomenon of interest. Simulations and initial flight tests yield promising results in terms of performance and safety, providing good perspectives for the use of this method to validate the phenomena models.

6.2 Future Work

There is still plenty of work to do about flow field energy harvesting by aircraft. There is interest in using the techniques described in this work in the PITVANT project (Portuguese Research & Technology Project on UAVs [116, 117, 118, 119, 120]), that supported part of this work. It is envisioned that these techniques will enable the use of UAVs by the Portuguese Air Force to execute a permanent surveillance of the large Portuguese exclusive economic zone (fig. 6.1).

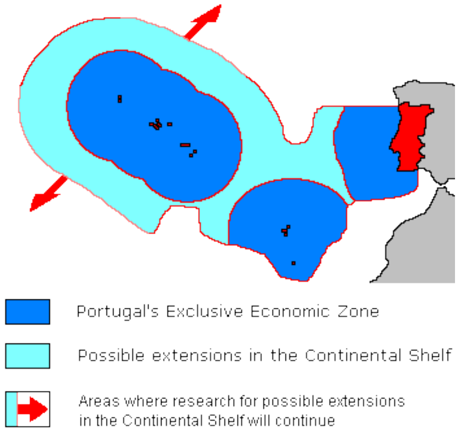


Figure 6.1: Portuguese exclusive economic zone

This thesis leaves several important questions unanswered. We outline some of those questions next. The current work also developed several algorithms that need to be validated in real world conditions, becoming useful tools. We describe what steps we envision for the verification of the developed methods and possible improvements for those. Further, we point to some other systems that would augment the current methods or that would be required for the functioning of the developed methods.

6.2.1 Open questions

Can a flow field exploitation controller be introduced in a UAV operation?

This is perhaps the most important question we came to face. We showed it is possible to largely extend the UAV flight endurance, but to do so the UAV needs to follow an exploitation trajectory. Most of the current operational scenarios have the UAVs flying to a target zone, executing its task, and coming back to the base airfield. In most of the operational situations, the activation of the exploration controller would require a more complex flight trajectory, sometimes increasing the time to reach the target zone, and always augmenting the path footprint, both in altitude and in horizontal area. Can the UAV missions accommodate such changes to the modus operandi? Which missions have the necessary flexibility? How can the exploration controller reduce its impact on the main mission goal execution?

How should the exploitation trajectories be combined?

If we want to be able to use all the flow field phenomena, the controller needs to handle several questions. When should it be exploiting thermals, wind shear, or gusts? When should it transition from dynamic wind shear soaring to static thermal soaring?

Can we harvest more energy if we have an electrical regeneration system?

We analyzed the problem of performing perpetual flight assuming the aircraft could only store the harvested energy in the form of altitude or velocity. There is also the possibility to have an electrical regeneration system, i.e., an electrical motor and a generator, to harvest the excess energy. The system would harvest energy through a propeller-generator, storing it on batteries. When required, an electrical motor would use the battery energy to propel the aircraft. Could such a system improve the potential for energy exploitation? Would the overhead and the losses of this system be compensated by the increase in harvested energy? If this system also provided the necessary energy for the aircraft electronics, what would be the change in the necessary conditions for perpetual flight?

Are the presented thermal and wind shear models valid?

As discussed before, most of the studied flow field phenomena models were not validated against real observation data. This important, because the phenomena models are the premises under which control algorithms and estimators are developed. If the models are incorrect, the estimators that use them should perform poorly. The controllers created to search for the phenomenon or to exploit it also depend on the characteristics of the modeled phenomenon.

Are *Bubble Thermals* more or less frequent than *Chimney Thermals*?

Or, which are the factors that affect the appearance rates of *Chimney Thermals* and *Bubble Thermals*? This is an example of two thermal models which affect the premises for the estimators and the controllers. There is no consensus about which of the thermal types is more common, or if both exist. A controller searching for a *Chimney Thermal* would not have to take into account the flight altitude, as long as the UAV is in the atmosphere mixing-layer. On the contrary, a controller searching for a *Bubble Thermal* needs to take into account the 3 dimensional space domain. The same happens with the estimators and the dependence of their performance on the estimation path and the observed thermal type.

6.2.2 Future developments

We now present some suggestions for future developments. Most of the suggestions envision the practical implementation and use of the tools developed in this thesis.

6.2.2.1 Flow Field Observer

Some of the estimation methods presented, in particular for wind shear and gusts, require the existence of a flow field observer. This observer would be required to output the airflow vector 3D components and, if possible, the rate of change of these components. The observer time constant should be small, as that is important to allow accurate estimates of the wind shear, and even more to detect gusts and enable a timely control response.

Such an observer needs to take into account the aircraft dynamics. Therefore, it requires the knowledge of the aircraft full 6 DOF model parameters. From the tests we performed we concluded that a time domain parameter observer yields poor results. A promising alternative is the method presented by Morelli [66], for aircraft parameter estimation in the frequency domain. This method was also specifically developed for aircraft not equipped with airflow angle sensors, as Angle-of-Attack (AOA) vanes. That is specially important for standard small UAVs, as these are not usually equipped with airflow angle sensors.

6.2.2.2 Thermal Estimation and Exploitation

The thermal locators presented may also be improved. The estimation evaluation function used is a function of the thermal Regularized Adaptive Particle Filter (RAPF) likelihood. We noticed that the likelihood is not always well correlated with the estimation performance. This makes us believe that there should be a better estimate evaluation function. With such a function the sensitivity to the number of particles and to the update rate may be better studied, allowing a better tunning of the estimator. Further, during the estimation process an estimate evaluation function may be considered as an online trust measure, which is most useful for user information and as input for soaring controllers.

As the thermal RAPF locator was developed to be integrated in UAVs, the method will only be fully proved when applied to real flight data, providing feedback to a soaring control system. After this system is tested in reality, a logical extension is the tracking of multiple thermals, allowing the UAV to optimize a soaring flight plan. It should also be extended to allow cooperative estimation by multiple UAVs, augmenting the team estimation performance and maybe more importantly the capacity to exploit the thermal field energy. As there are different types of thermals, the system should also be extended to be able to cope with multiple thermal types and distinguish them.

This work also reveals that a statistical study about regional distribution of thermals and their characteristics would provide very important inputs to flight path planners. That study could characterize the thermal appearance probability over an area, including the localization of *hot spots*, i.e., points over which there is a higher probability of formation of a thermal. This would allow a global enhancement of the trajectory energy efficiency. Another important statistical characterization is the distribution of the thermal updraft strength and size over different regions. This characterization is very important to assert or refuse some of the perpetual flight analysis assumptions. Also, if the regional thermal characteristics are well characterized, the estimation methods may use prior regional data, instead of relying solely on the aircraft sensor data.

The thermal localization may be enhanced if the flight controller is designed to improve the estimation process. The flight path may be defined by taking information theory into account, generating a more informative flight path.

6.2.2.3 Wind Shear Estimation

From the three wind shear phenomena we described we presented an estimator only for the *Surface Wind Shear*. The extension of the estimator for *Layer Wind Shear* should be straightforward, and will expand largely the estimator applicability. Further, as for the thermal locator, the extension for wind shear collaborative localization and exploitation should improve the team endurance and the wind shear estimate quality greatly.

During the present study the only wind shear soaring controllers we found were off-line flight path planners. As such, a very important development for wind shear soaring would be the development of a feedback-based flight controller, which would enable UAVs to use the wind shear in practice.

6.2.2.4 Control in Flow Fields

Most of the information generated by the flow field estimators we discussed in this work is important only if there are exploitation flight controllers capable of using it. Such a controller needs to be able to cope with the dynamic and stochastic nature of the flow field phenomena. The flight controller may define an initial optimized flight plan, but it should also be able to adapt it with on-route feedback. An example with thermal soaring is the definition of the limit altitude at which the aircraft should leave a thermal and the airspeed it should fly while searching for new thermals. Three of the competing goals in an optimal flight path definition are the: localization of the phenomenon, its exploitation, and the global mission goal of the aircraft.

APPENDICES

APPENDIX A

Thermal RAPF Estimator

```
RAPF: $\left(\chi_{t-1}, \dot{E}_{a,t}, M, w_{slow}, w_{fast}, N_{RSplMin}, N_{RandMin}, g\right)$
$\overline{\chi}_t = \chi_t = \emptyset$
for $m \leftarrow 1$ to $M$ do do
    $x_t^{[m]} =$ SampleMotionModel $\left(x_{t-1}^{[m]}\right)$;
    $y_t^{[m]} =$ SampleEnvironmentModel $\left(y_{t-1}^{[m]}\right)$;
    $w_t^{[m]} =$ MeasurementModel $\left(\dot{E}_{a,t}, x_t^{[m]}, y_t^{[m]}\right)$;
    $\overline{\chi}_t = \overline{\chi}_t + \left\langle x_t^{[m]}, y_t^{[m]}, w_t^{[m]} \right\rangle$;
    $w_{avg} = w_{avg} + \frac{w_t^{[m]}}{M}$;
end
//Likelihood evaluation for adaptive part
$w_{slow} = w_{slow} + \alpha_{slow} \left(w_{avg} - w_{slow}\right)$;
$w_{fast} = w_{fast} + \alpha_{fast} \left(w_{avg} - w_{fast}\right)$;
$N_{Rand} = N_{RandMin} + \left\lceil \left(M - N_{RSplMin} - N_{RandMin}\right) \max\left(1 - \frac{w_{fast}}{w_{slow}}, 0\right) \right\rceil$;
$N_{RSpl} = M - N_{RandMin}$
//Parameters for regularization
Compute the empirical covariance matrix $\mathbf{S}_t$ of $\overline{\chi}_t$;
Calculate to bandwidth $h = g \cdot \left[\frac{1}{2N_{RSpl}}\right]^{0.1}$;
//Resampling
for $m \leftarrow 1$ to $N_{RSpl}$ do do
    draw $i \in [1, ..., M]$ with probability $\propto w_t^{[i]}$;
    draw $\left[x_t^{[i]*}; y_t^{[i]*}\right] \sim \mathcal{N}\left(\left[x_t^{[i]}; y_t^{[i]}\right], h^2 \mathbf{S}_t\right)$;
    $\chi_t = \chi_t + \left\langle x_t^{[i]*}, y_t^{[i]*}, \frac{1}{M} \right\rangle$;
end
$\chi_t^{\left[N_{RSpl}+1:M\right]} =$ GenerateRandParticle $\left(N_{Rand}\right)$;
return: $\chi_t$
```

Algorithm 2: Regularized Adaptive Particle Filter (*RAPF*) pseudo-code

APPENDIX B

Safety and Liveness Proofs - Collision Avoidance System for Close Proximity Operations

B.1 Properties proof: Model 1 - Single integrator

Theorem B.1. *The overall system satisfies the* Safety *property, if we assume single integrator holonomic vehicles, if the* Close Proximity Collision Avoidance Algorithm (CPCAA) *is part of the control system, and if* $(\|\Delta\mathbf{x}_{12}\| \geq d_{s_{12}})(t_0)$ *holds true.*

Proof. If the desired trajectories for UAV 1 and UAV 2 are such that $(\Box \|\Delta\mathbf{x}_{12}\| > d_{s_{12}})(t)$, no collision will occur, because

$$d_{s_{12}} = r_{c_1} \cdot a_1 + r_{c_2} \cdot a_2 > d_{c_{12}}, \tag{B.1}$$

proving

$$(\Box \|\Delta\mathbf{x}_{12}\| > d_{c_{12}})(t). \tag{B.2}$$

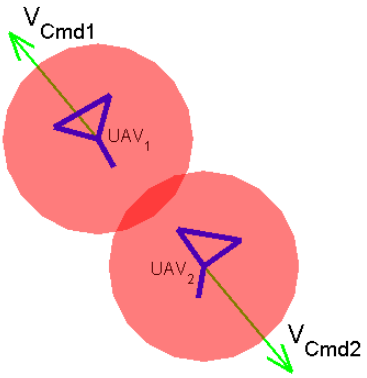


Figure B.1: Collision avoidance commands

If on any part of the trajectories $\|\Delta\mathbf{x}_{12}\| \leq d_{s_{12}}$, then the *CPCAA* state is activated, forcing $\mathbf{v}_1 = -\frac{\Delta\mathbf{x}_{12}}{\|\Delta\mathbf{x}_{12}\|} \cdot V_{a_1}$ and $\mathbf{v}_2 = \frac{\Delta\mathbf{x}_{12}}{\|\Delta\mathbf{x}_{12}\|} \cdot V_{a_2}$ (fig. B.1), which results in,

$$\begin{aligned}
&\frac{d\left\|\Delta\mathbf{x}_{12}\right\|}{dt} = \frac{\left\langle \Delta\mathbf{x}_{12}\left|\Delta\mathbf{v}_{12}\right.\right\rangle}{\left\|\Delta\mathbf{x}_{12}\right\|} = \frac{\left\langle \Delta\mathbf{x}_{12}\left|\mathbf{v}_{2}-\mathbf{v}_{1}\right.\right\rangle}{\left\|\Delta\mathbf{x}_{12}\right\|} = ...\\
&... = \left\langle \frac{\Delta\mathbf{x}_{12}}{\left\|\Delta\mathbf{x}_{12}\right\|}\left|\frac{\Delta\mathbf{x}_{12}}{\left\|\Delta\mathbf{x}_{12}\right\|}\cdot V_{a_2} - \left(-\frac{\Delta\mathbf{x}_{12}}{\left\|\Delta\mathbf{x}_{12}\right\|}\cdot V_{a_1}\right)\right.\right\rangle = ...\\
&... = \left\langle \frac{\Delta\mathbf{x}_{12}}{\left\|\Delta\mathbf{x}_{12}\right\|}\left|\frac{\Delta\mathbf{x}_{12}}{\left\|\Delta\mathbf{x}_{12}\right\|}\cdot \left(V_{a_2}+V_{a_1}\right)\right.\right\rangle = \left\langle \Delta\mathbf{x}_{12}\left|\Delta\mathbf{x}_{12}\right.\right\rangle \frac{V_{a_2}+V_{a_1}}{\left\|\Delta\mathbf{x}_{12}\right\|^2} = ...\\
&... = V_{a_2}+V_{a_1} \geq 0 \Rightarrow \frac{d\left\|\Delta\mathbf{x}_{12}\right\|}{dt} \geq 0,
\end{aligned} \tag{B.3}$$

meaning that:

$$\begin{aligned}
&\left(\left\|\Delta\mathbf{x}_{12}\right\| \leq d_{s_{12}}\right)(t_1) \Rightarrow \left(\frac{d\left\|\Delta\mathbf{x}_{12}\right\|}{dt} \geq 0\right)(t)\,, t\in[t_1,t_2] \Rightarrow ...\\
&...\exists t_2 > t_1 : \left(\circ\left\|\Delta\mathbf{x}_{12}\right\| > d_{s_{12}}\right)(t_2) \therefore \left(\Box\left\|\Delta\mathbf{x}_{12}\right\| \geq d_{s_{12}}\right)(t)\,, t > t_2.
\end{aligned} \tag{B.4}$$

And again, because $d_{s_{12}} > d_{c_{12}}$

$$CPCAA \models \left(\Box\left\|\Delta\mathbf{x}_{12}\right\| > d_{c_{12}}\right)(t)\,, \tag{B.5}$$

proving the safety property. ☐

Lemma B.2. *If the* CPCAA *is activated the system loses the liveness property, with a single integrator dynamic model.*

Proof. If the *CPCAA* is activated it means that the initially commanded trajectories result in $\frac{d\|\Delta\mathbf{x}_{12}\|}{dt} < 0$. The *CPCAA* activation enforces $\frac{d\|\Delta\mathbf{x}_{12}\|}{dt} \geq 0$. With the single integrator dynamic vehicle model, the system reacts instantaneously to the commands of the original state and the *CPCAA* state. This results in a "livelock" between both states, which keeps the vehicles from reaching their target waypoints. ☐

Remark B.3. If the initially commanded trajectories result exactly in $\mathbf{v}_{cmd,1} = -\frac{\Delta\mathbf{x}_{12}}{\|\Delta\mathbf{x}_{12}\|}\cdot V_{a_1}$ and $\mathbf{v}_{cmd,2} = -\mathbf{v}_{cmd,1}$ a "live-lock" would also occur, independently from the instantaneous reactions of the vehicles. However the probability of such a situation is null, and real vehicles and control systems don't react instantaneously. That being, a "live-lock" would never occur, but the resulting saw teeth path would be most inefficient and undesirable.

Lemma B.4. *The overall system satisfies the liveness property, if it stays enough time in the* TrackWP *state.*

$$\exists t_{f_i} : \left(\left\|\Delta\mathbf{x}_{i\_WP}\right\| \leq d_{wp_i}\right)(t_{f_i}) \therefore \left(\Diamond\left\|\Delta\mathbf{x}_{i\_WP}\right\| \leq d_{wp_i}\right)(t_0) \tag{B.6}$$

Proof. In the *TrackWP* state all vehicles are commanded to head directly to their current waypoints, with:

$$\mathbf{v}_i = \frac{\Delta\mathbf{x}_{i\_WP}}{\left\|\Delta\mathbf{x}_{i\_WP}\right\|}\cdot V_{c_i} \tag{B.7}$$

Therefore, each vehicle i will reach a position $\Delta\mathbf{x}_{i\_WP}$, where$\|\Delta\mathbf{x}_{i_WP}\| \leq d_{wp_i}$, in $t_{f_i} = \frac{\|\Delta\mathbf{x}_{i\_WP}\|-d_{wp_i}}{\|\mathbf{v}_i\|} = \frac{\|\Delta\mathbf{x}_{i\_WP}\|-d_{wp_i}}{V_{c_i}}$ time units, proving the lemma. ☐

We can further extend liveness to situations with less restrictive constraints. To achieve that we need to prove that the system can always reach the *TrackWP* state and never enters the *CPCAA* state.

Lemma B.5. *The system never enters the* CPCAA *state, if the follower is in the leader's* Frontal Area *and out of the* Close Frontal Area*.*

$$(\square (\mathbf{x}_F \in \{FrontArea_{LF} \backslash CloseFArea_{LF}\} \Rightarrow \circ (\mathbf{x}_F \notin SafeArea_{LF}))) (t) \tag{B.8}$$

Proof. Suppose the follower start position $\mathbf{x}_F (t_0)$ is in the leader's *Frontal Area*, out of the *Close Frontal Area* (fig. B.2), and on the right side of the line connecting the leader to its waypoint. We need to know the minimum possible separation between both aircraft before the follower reaches the *Buffer Area*,

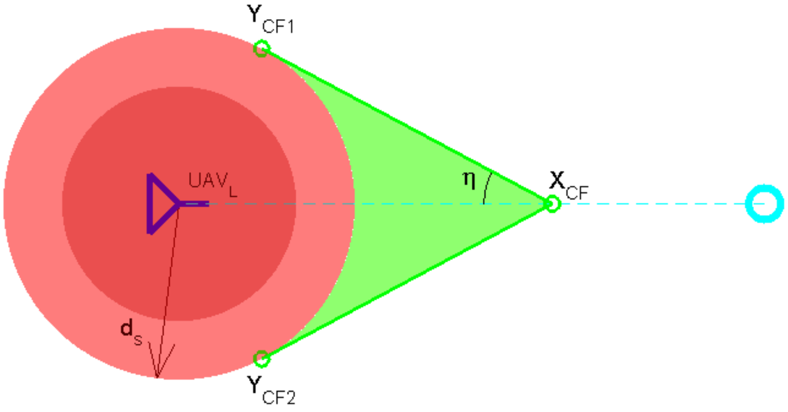


Figure B.2: Close Frontal Area

$$d_{GMin} = \min_{t \geq t_0} \{ \|\Delta \mathbf{x}_{LF}\| (t) : \mathbf{x}_F (t) \notin BuffArea_{LF}, \ ... \\ ... \ \mathbf{x}_F (t_0) \in \{FrontArea_{LF} \backslash CloseFArea_{LF}\}\} . \tag{B.9}$$

The derivative of $\|\Delta \mathbf{x}_{LF}\|$ is,

$$\frac{d \|\Delta \mathbf{x}_{LF}\|}{dt} = \frac{\langle \Delta \mathbf{x}_{LF} | \Delta \mathbf{v}_{LF} \rangle}{\|\Delta \mathbf{x}_{LF}\|}, \tag{B.10}$$

where

$$\Delta \mathbf{v}_{LF} = \mathbf{v}_{cmd,F} - \mathbf{v}_L = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \frac{\Delta \mathbf{x}_{L\_WP}}{\|\Delta \mathbf{x}_{L\_WP}\|} \cdot V_{c_F} - \frac{\Delta \mathbf{x}_{L\_WP}}{\|\Delta \mathbf{x}_{L\_WP}\|} \cdot V_{c_L} = ... \\ ... = \begin{bmatrix} -V_{c_L} & -V_{c_F} \\ V_{c_F} & -V_{c_L} \end{bmatrix} \frac{\Delta \mathbf{x}_{L\_WP}}{\|\Delta \mathbf{x}_{L\_WP}\|}, \tag{B.11}$$

resulting in:

$$\begin{aligned}\frac{d\left\|\Delta\mathbf{x}_{LF}\right\|}{dt} &= \left\langle \frac{\Delta\mathbf{x}_{LF}}{\left\|\Delta\mathbf{x}_{LF}\right\|} \middle| \begin{bmatrix} -V_{c_L} & -V_{c_F} \\ V_{c_F} & -V_{c_L} \end{bmatrix} \frac{\Delta\mathbf{x}_{L\_WP}}{\left\|\Delta\mathbf{x}_{L\_WP}\right\|} \right\rangle = \ldots \\ \ldots &= \left\langle \mathbf{R}_{L\_WP} \cdot \frac{\Delta\mathbf{x}_{LF}}{\left\|\Delta\mathbf{x}_{LF}\right\|} \middle| \mathbf{R}_{L\_WP} \begin{bmatrix} -V_{c_L} & -V_{c_F} \\ V_{c_F} & -V_{c_L} \end{bmatrix} \frac{\Delta\mathbf{x}_{L\_WP}}{\left\|\Delta\mathbf{x}_{L\_WP}\right\|} \right\rangle = \ldots \\ \ldots &= \left\langle \frac{\Delta\mathbf{x}_{LF}^{L\_WP}}{\left\|\Delta\mathbf{x}_{LF}\right\|} \middle| \begin{bmatrix} -V_{c_L} \\ V_{c_F} \end{bmatrix} \right\rangle = \frac{V_{c_F} \cdot \Delta y_{LF}^{L\_WP} - V_{c_L} \cdot \Delta x_{LF}^{L\_WP}}{\left\|\Delta\mathbf{x}_{LF}\right\|}, \end{aligned} \tag{B.12}$$

and $\left\|\Delta\mathbf{x}_{LF}\right\|(t) = d_{GMin}$ only if

$$\frac{d\left\|\Delta\mathbf{x}_{LF}\right\|}{dt} = 0 = V_{c_F} \cdot \Delta y_{LF}^{L\_WP} - V_{c_L} \cdot \Delta x_{LF}^{L\_WP} \Rightarrow \Delta y_{LF}^{L\_WP} = \pm\Delta x_{LF}^{L\_WP} \frac{V_{c_L}}{V_{c_F}}. \tag{B.13}$$

This means that the global minimum is on the lines drawn from the leader position with function $\Delta y_{LF}^{L\_WP} = \pm\Delta x_{LF}^{L\_WP} \frac{V_{c_L}}{V_{c_F}}$, as illustrated in figure B.3 by the black lines. Moreover, any start position outside the global minimum lines leads the vehicles to an ever increasing separation. We can call this area the *Frontal ejection area*,

$$FEjectArea_{ij} = \left\{ \mathbf{x}_j : \mathbf{x}_j \in FrontArea_{ij} \wedge \left|\Delta y_{ij}^{i\_WP}\right| \geq \Delta x_{ij}^{i\_WP} \frac{V_{c_i}}{V_{c_j}} \right\}. \tag{B.14}$$

This results in

$$\min_{t \geq t_0} \left\{ \left\|\Delta\mathbf{x}_{LF}\right\|(t) : \mathbf{x}_F(t) \in \{FEjectArea_{LF}\} \right\} = \left\|\Delta\mathbf{x}_{LF}\right\|(t_0) > d_{s_{LF}}. \tag{B.15}$$

Now we must prove a similar result for the rest of the *Frontal Area*. Let us define a minimum reachable distance d_{min} as

$$d_{min} = \sqrt{\left(\Delta y_{LF}^{L\_WP}\right)^2 + \left(\Delta x_{LF}^{L\_WP}\right)^2} = \Delta x_{LF}^{L\_WP} \sqrt{\left(\frac{V_{c_L}}{V_{c_F}}\right)^2 + 1}. \tag{B.16}$$

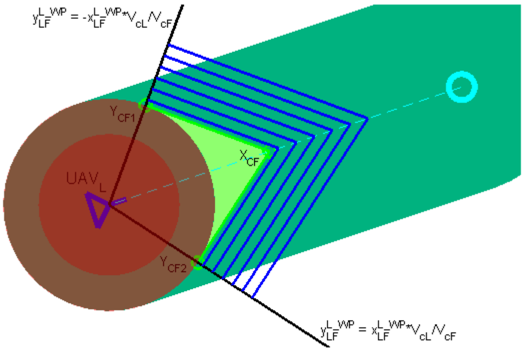


Figure B.3: Global minimum lines

To achieve that minimum distance, because the relative velocity in the leader's reference frame is always $\begin{bmatrix} -V_{c_L} \\ V_{c_F} \end{bmatrix}$, while the follower is in the *Frontal Area*, the follower would have to depart from a position in the lines illustrated in figure B.3 in blue, defined by:

$$\Delta y_{LF}^{L\_WP} = \mp \Delta x_{LF}^{L\_WP} \frac{V_{c_F}}{V_{c_L}} \pm d_{min} \sqrt{\left(\frac{V_{c_F}}{V_{c_L}}\right)^2 + 1},\ \Delta x_{LF}^{L\_WP} \geq \frac{d_{min}}{\sqrt{\left(\frac{V_{c_L}}{V_{c_F}}\right)^2 + 1}}, \tag{B.17}$$

Let us now exclude the points in the *Frontal ejection area.* If no start position is in the *Close Frontal Area*, by definition, all points $\Delta \mathbf{x}_{LF}(t_0)$ on the right side of the *Frontal Area* respect

$$\Delta x_{LF}^{L\_WP} > \frac{d_{s_{ij}}}{\sin \eta_{ij}} - \Delta y_{LF}^{L\_WP} \frac{V_{c_L}}{V_{c_F}} \Leftrightarrow \Delta y_{LF}^{L\_WP} > \left(\frac{d_{s_{ij}}}{\sin \eta_{ij}} - \Delta x_{LF}^{L\_WP}\right) \frac{V_{c_F}}{V_{c_L}}, \tag{B.18}$$

and the ones on the left side of the *Frontal Area* respect

$$\Delta y_{LF}^{L\_WP} < -\left(\frac{d_{s_{ij}}}{\sin \eta_{ij}} - \Delta x_{LF}^{L\_WP}\right) \frac{V_{c_F}}{V_{c_L}}. \tag{B.19}$$

If we mix equations B.17 and B.18, for the right side, we end up with,

$$\begin{gathered} \left(\frac{d_{s_{ij}}}{\sin \eta_{ij}} - \Delta x_{LF}^{L\_WP}\right) \frac{V_{c_F}}{V_{c_L}} < -\Delta x_{LF}^{L\_WP} \frac{V_{c_F}}{V_{c_L}} + d_{min} \sqrt{\left(\frac{V_{c_F}}{V_{c_L}}\right)^2 + 1} \Leftrightarrow \ldots \\ \ldots \Leftrightarrow \frac{d_{s_{ij}}}{\sin \eta_{ij}} \frac{V_{c_F}}{V_{c_L} \sqrt{\left(\frac{V_{c_F}}{V_{c_L}}\right)^2 + 1}} < d_{min} \Leftrightarrow \frac{d_{s_{ij}}}{\sin \eta_{ij}} \frac{\tan \eta_{ij}}{\sqrt{(\tan \eta_{ij})^2 + 1}} = d_{s_{LF}} < d_{min}. \end{gathered} \tag{B.20}$$

The same procedure can be used for the left side. The equations B.15 and B.20 result in,

$$d_{GMin} = \min_{\substack{\mathbf{x}_F(t_0) \in \{FrontArea_{LF} \setminus CloseFArea_{LF}\} \\ t \geq t_0}} \{\|\Delta \mathbf{x}_{LF}\|(t) : \mathbf{x}_F(t) \notin BuffArea_{LF}\} > d_{s_{LF}}, \tag{B.21}$$

proving that if the follower is in the *Frontal Area* and out of the *Close Frontal Area*, it will never enter the *Safety Area*, as intended. □

If the follower is in the *Frontal Area*, the system can only pass to the *TrackWP* state if:

1. The follower movement area is near enough to its waypoint, triggering the *reach* assertion, and the new trajectory doesn't conflict with the leader's one.
2. The leader reaches its waypoint and the new trajectory doesn't conflict with the follower's one.

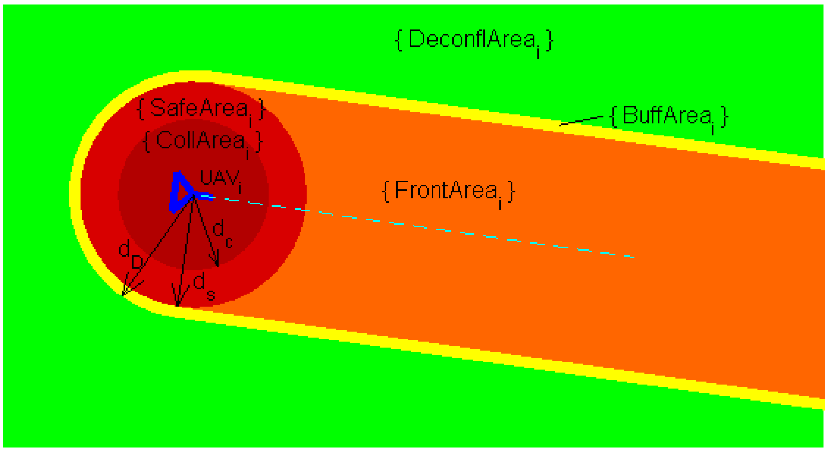


Figure B.4: Operating areas

3. The follower can reach the *Buffer Area*, then the *Deconfliction Area* and finally the *TrackWP* state (fig. B.4).

Lemma B.6. *The follower can reach the leader's* Buffer Area*, if the follower is in the leader's* Frontal Area *and out of the* Close Frontal Area.

$$\left(\mathbf{x}_F \in \{FrontArea_{LF} \backslash CloseFArea_{LF}\} \rightarrow \circ \left(\mathbf{x}_F \in BuffArea_{LF}\right)\right)(t) \tag{B.22}$$

Proof. From the *Frontal Area*, the follower can only reach the *Safety* and the *Buffer Area*. By lemma's B.5 conclusions, the follower can only reach the *Buffer Area*. The follower's exit velocity vector is always perpendicular to the boundary between the *Frontal Area* and the *Buffer Area*. As such, the maximum time needed for the follower to reach the *Buffer Area* is,

$$t_{FtoB\_max} = \frac{d_{s_{LF}}}{V_{c_F}}, \tag{B.23}$$

proving that the follower will reach the *Buffer Area*. □

Lemma B.7. *The system never enters the* CPCAA *state, if the follower is on the leader's* Buffer Area.

$$\left(\Box \left(\mathbf{x}_F \in BuffArea_{LF} \Rightarrow \circ \left(\mathbf{x}_F \notin SafeArea_{LF}\right)\right)\right)(t)\,. \tag{B.24}$$

Proof. Suppose the follower start position $\mathbf{x}_F(t_0)$ is in the leader's *Buffer Area*. To prove that the system doesn't enter on *CPCAA* state, we need to prove that the distance global minimum is greater than the safety distance,

$$\begin{aligned} d_{GMin} = \min_{t \geq t_0} \{ \|\Delta \mathbf{x}_{LF}\|(t) : \mathbf{x}_F(t) \notin \{DeconflArea_{LF}, FrontArea_{LF}\}, \ldots \\ \ldots \mathbf{x}_F(t_0) \in \{BuffArea_{LF}\}\} > d_{s_{LF}}. \end{aligned} \tag{B.25}$$

When the follower is in the leader's *Buffer Area*, there are three possible situations for the valid command options: only the option with $\kappa = 1$ is valid, only the option with $\kappa = -1$ is valid, or both are valid, as illustrated in figure B.5.

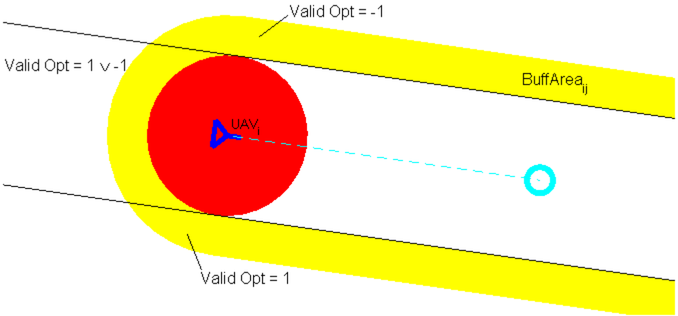


Figure B.5: Buffer Area - Valid command option regions

Let us assume the follower start position is such that $\Delta x_{LF}^{L\_WP} \leq 0$. Here, the commanded velocity for the follower will be:

$$\mathbf{v}_{cmd,F} = \begin{bmatrix} \cos\gamma & \pm\sin\gamma \\ \mp\sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F} \Rightarrow \mathbf{v}^{LF}_{cmd,F} = \begin{bmatrix} -\cos\gamma \\ \pm\sin\gamma \end{bmatrix} V_{c_F}, \tag{B.26}$$

where $\gamma = \pi - \arctan\left(\frac{d_{Buff}\cos\sigma}{d_{Buff}\sin\sigma - d_{12}}\right)$. The derivative of the distance between both aircraft is,

$$\begin{aligned} \frac{d\,\|\Delta\mathbf{x}_{LF}\|}{dt} &= \frac{\langle \Delta\mathbf{x}_{LF}\,|\Delta\mathbf{v}_{LF}\rangle}{\|\Delta\mathbf{x}_{LF}\|} = \left\langle \frac{\Delta\mathbf{x}_{LF}^{LF}}{\|\Delta\mathbf{x}_{LF}\|} \middle| \begin{bmatrix} -\cos\gamma \\ \pm\sin\gamma \end{bmatrix} V_{c_F} - \begin{bmatrix} \cos\psi_{LF} \\ -\sin\psi_{LF} \end{bmatrix} V_{c_L} \right\rangle = \ldots \\ \ldots &= -\left(V_{c_F}\cos\gamma + V_{c_L}\cos\psi_{LF}\right), \end{aligned} \tag{B.27}$$

where ψ_{LF} is the angle between leader's movement vector and the follower's relative

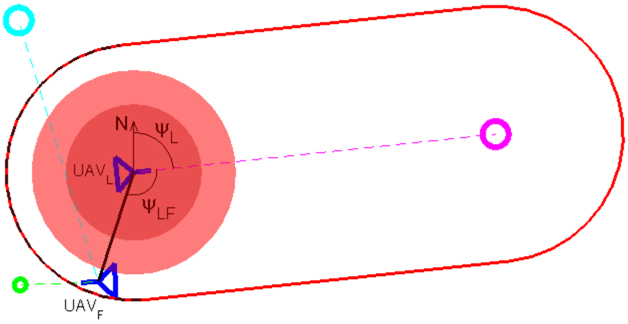


Figure B.6: Leader to follower angle

position (fig. B.6). If we now consider only the points in the leader's *Buffer Area*,

$$\gamma \in \left(\frac{\pi}{2}, \pi\right), \tag{B.28}$$

and if the follower is behind the leader,

$$\psi_{LF} \in \left[\frac{\pi}{2}, \frac{3\pi}{2}\right]. \tag{B.29}$$

This results in,

$$\frac{d\left\|\Delta\mathbf{x}_{LF}\right\|}{dt} > 0 : \psi_{LF} \in \left[\frac{\pi}{2}, \frac{3\pi}{2}\right] \wedge \gamma \in \left(\frac{\pi}{2}, \pi\right), \tag{B.30}$$

which means that the velocity command always points away from the leader, if the follower is behind it, and:

$$\begin{aligned}&\min_{t \geq t_0} \left\{ \left\|\Delta\mathbf{x}_{LF}\right\|(t) : \mathbf{x}_F(t) \notin \left\{DeconflArea_{LF}, FrontArea_{LF}\right\} \wedge \Delta x_{LF}^{L\_WP} \leq 0, \text{ ...}\right.\\ &\left.\text{... } \mathbf{x}_F(t_0) \in \left\{BuffArea_{LF}\right\}\right\} = \left\|\Delta\mathbf{x}_{LF}\right\|(t_0) > d_{s_{LF}}.\end{aligned} \tag{B.31}$$

Let us now assume that the follower start position is such that $\Delta x_{LF}^{L\_WP} > 0$ and it is on the right side of the *Buffer Area.* This implies that $d_{s_{LF}} < \Delta y_{LF}^{L\_WP} < d_{D_{LF}}$, that $\psi_{LF} \in \left[0, \frac{\pi}{2}\right)$, and also that

$$\gamma \in \left(\psi_{LF} + \arctan\left(\frac{d_{Buff} - d_s}{\Delta x_{LF}^{L\_WP}}\right), \pi\right). \tag{B.32}$$

In this region, only the option with $\kappa = 1$ is valid and the commanded velocity for the follower will be:

$$\mathbf{v}_{cmd,F}^{LF} = \begin{bmatrix} -\cos\gamma \\ \sin\gamma \end{bmatrix} V_{c_F} \tag{B.33}$$

If we now take the distance derivative between both aircraft, projected only on the Y axis of the leader:

$$\begin{aligned}\frac{d\Delta y_{LF}^{L\_WP}}{dt} &= \left\langle \Delta\mathbf{x}_{L\_WP} \perp \Delta\mathbf{v}_{LF} \right\rangle = \left\langle \begin{bmatrix} \cos\psi_{LF} \\ -\sin\psi_{LF} \end{bmatrix} \perp \mathbf{v}_{cmd,F}^{LF} \right\rangle = \text{...}\\ \text{...} &= \left\langle \begin{bmatrix} \sin\psi_{LF} \\ \cos\psi_{LF} \end{bmatrix} \middle\| \begin{bmatrix} -\cos\gamma \\ \sin\gamma \end{bmatrix} V_{c_F} \right\rangle = \text{...}\\ \text{...} &= V_{c_F}\left(\cos\psi_{LF}\sin\gamma - \sin\psi_{LF}\cos\gamma\right) = V_{c_F}\sin\left(\gamma - \psi_{LF}\right).\end{aligned} \tag{B.34}$$

We can minimize this derivative with argument ψ_{LF}:

$$\begin{aligned}\min_{\psi_{LF} \in \left[\frac{\pi}{2}, \pi\right]} \frac{d\Delta y_{LF}^{L\_WP}}{dt} &= \min_{\psi_{LF} \in \left[\frac{\pi}{2}, \pi\right]} V_{c_F}\sin\left(\gamma - \psi_{LF}\right) = \text{...}\\ \text{...} &= \min_{\psi_{LF} \in \left[\frac{\pi}{2}, \pi\right]} V_{c_F}\sin\left(\psi_{LF} + \arctan\left(\frac{d_{Buff} - d_s}{\Delta x_{LF}^{L\_WP}}\right) + w - \psi_{LF}\right)\end{aligned} \tag{B.35}$$

where w has an infinitesimal value, greater then 0. Thus, resulting in

$$\min_{\psi_{LF}\in\left[\frac{\pi}{2},\pi\right]} \frac{d\Delta y_{LF}^{L\_WP}}{dt} = \min_{\psi_{LF}\in\left[\frac{\pi}{2},\pi\right]} V_{c_F} \sin\left(\arctan\left(\frac{d_{Buff}-d_s}{\Delta x_{LF}^{L\_WP}}\right)+w\right) \tag{B.36}$$

and because $d_{Buff}-d_s > 0 \wedge \Delta x_{LF}^{L\_WP} > 0 \Rightarrow \arctan\left(\frac{d_{Buff}-d_s}{\Delta x_{LF}^{L\_WP}}\right)+w > 0$, then

$$\left(\Box\left(\left\{\mathbf{x}_F : \Delta x_{LF}^{L\_WP} > 0 \wedge \mathbf{x}_F \in \{BuffArea_{LF}\}\right\} \Rightarrow \min_{\psi_{LF}\in\left[\frac{\pi}{2},\pi\right]} \frac{d\Delta y_{LF}^{L\_WP}}{dt} > 0\right)\right)(t) \tag{B.37}$$

The same can be proved by symmetry for the left side of the *Buffer Area*, with $\Delta y_{LF}^{L\_WP} < -d_{s_{LF}}$ and the option $\kappa = -1$, which implies that

$$\min_{\substack{\mathbf{x}_F(t_0)\in\{BuffArea_{LF}\}\wedge\Delta x_{LF}^{L\_WP}>0 \\ t\geq t_0}} \{\|\Delta \mathbf{x}_{LF}\|(t) : \mathbf{x}_F(t) \notin \{DeconflArea_{LF}, FrontArea_{LF}\}\} = ...$$

$$... = \min_{\substack{\mathbf{x}_F(t_0)\in\{BuffArea_{LF}\}\wedge\Delta x_{LF}^{L\_WP}>0 \\ t\geq t_0}} \{\langle\Delta \mathbf{x}_{L\_WP} \perp \Delta \mathbf{x}_{LF}\rangle(t_0)\} > d_{s_{LF}} \tag{B.38}$$

Equations (B.31) and (B.38) prove the lemma. □

Lemma B.8. *The follower will never go back to the* Frontal Area*, once in the leader's* Buffer Area*.*

$$(\Box(\mathbf{x}_F \in BuffArea_{LF} \Rightarrow \circ(\mathbf{x}_F \notin FrontArea_{LF})))(t) \tag{B.39}$$

Proof. The equation B.37 proves that if the follower is at any point from the *Buffer Area* on the interface with the *Frontal Area* it will move away from the interface. □

Lemma B.9. *The follower will reach the* Deconfliction Area*, once in the leader's* Buffer Area*.*

$$(\mathbf{x}_F \in BuffArea_{LF} \rightarrow \circ(\mathbf{x}_F \in DeconflArea_{LF}))(t) \tag{B.40}$$

Proof. Equation B.37 also proves that if the follower is at any point from the *Buffer Area* on the interface with the *Deconfliction Area* and $\Delta x_{LF}^{L\_WP} > 0$, it will move away from the line between the leader and its waypoint, i.e., into the *Deconfliction Area.* Furthermore, equation B.30 proves that, if the follower is on the *Buffer Area* and $\Delta x_{LF}^{L\_WP} \leq 0$, the radial distance between both aircraft always increases. This means that if the follower is on the interface between the *Buffer Area* and the *Deconfliction Area*, it will move outwards to the *Deconfliction Area*, proving the lemma. □

Lemma B.10. *Once the follower is in the leader's* Deconfliction Area *it will not enter the* Buffer Area*.*

$$(\Box(\mathbf{x}_F \in DeconflArea_{LF} \Rightarrow \circ(\mathbf{x}_F \notin BuffArea_{LF})))(t)\,. \tag{B.41}$$

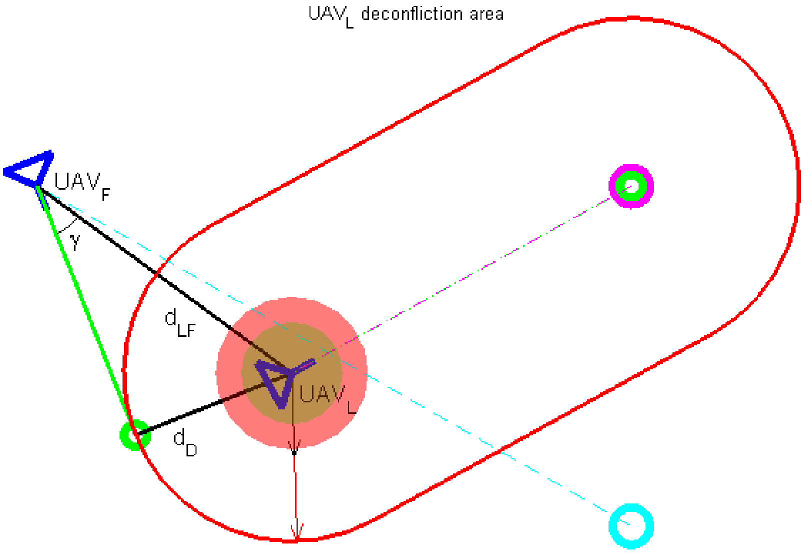


Figure B.7: Basic deconfliction path

Proof. Let us define the interface between the *Buffer* and *Deconfliction Area* (fig. B.7):

$$DeconflPath=\left\{\mathbf{x}_F:\begin{cases}\Delta y_{LF}^{L\_WP}=d_{D_{LF}} & \Leftarrow \Delta x_{LF}^{L\_WP}>0\wedge\Delta y_{LF}^{L\_WP}>0\\ \Delta y_{LF}^{L\_WP}=-d_{D_{LF}} & \Leftarrow \Delta x_{LF}^{L\_WP}>0\wedge\Delta y_{LF}^{L\_WP}<0\\ \left\|\mathbf{x}_{LF}\right\|=d_{D_{LF}} & \Leftarrow \Delta x_{LF}^{L\_WP}\leq 0\end{cases}\right\}. \tag{B.42}$$

Notice that $DeconflPath\in DeconflArea_{LF}$. If $\Delta x_{LF}^{L\_WP}\geq 0\wedge\Delta y_{LF}^{L\_WP}>0$,

$$\begin{aligned}&\mathbf{v}_{cmd,F}=\left[\begin{array}{cc}\cos\gamma & \sin\gamma\\ -\sin\gamma & \cos\gamma\end{array}\right]\frac{\mathbf{x}_{FL}}{\left\|\mathbf{x}_{FL}\right\|}\cdot V_{c_F}\Rightarrow\ldots\\ &\ldots\Rightarrow\Delta\mathbf{v}_{LF}^{L\_WP}=\mathbf{v}_{cmd,F}^{L\_WP}-\mathbf{v}_{L}^{L\_WP}=\left[\begin{array}{cc}\cos\gamma & \sin\gamma\\ -\sin\gamma & \cos\gamma\end{array}\right]\left[\begin{array}{c}-\cos\psi_{LF}\\ -\sin\psi_{LF}\end{array}\right]V_{c_F}-\left[\begin{array}{c}1\\ 0\end{array}\right]V_{c_L}=\ldots\\ &\ldots=\left[\begin{array}{c}-\cos\left(\gamma-\psi_{LF}\right)\\ \sin\left(\gamma-\psi_{LF}\right)\end{array}\right]V_{c_F}-\left[\begin{array}{c}1\\ 0\end{array}\right]V_{c_L}.\end{aligned} \tag{B.43}$$

At the interface ($\Delta y_{LF}^{L\_WP}=d_{D_{LF}}$):

$$\gamma=\psi_{LF}\Rightarrow\Delta\mathbf{v}_{LF}^{L\_WP}=-\left[\begin{array}{c}1\\ 0\end{array}\right]\left(V_{c_F}+V_{c_L}\right), \tag{B.44}$$

meaning that the follower would stay on the line $\Delta y_{LF}^{L\_WP}=d_{D_{LF}}$. Using the symmetry for $\Delta y_{LF}^{L\_WP}<0$, we may state that

$$\begin{aligned}&\left(\Box\left(\mathbf{x}_F\in DeconflArea_{LF}\wedge\Delta x_{LF}^{L\_WP}\geq 0\Rightarrow\left|\Delta y_{LF}^{L\_WP}\right|\geq d_{D_{LF}}\right)\right)(t)\Rightarrow\ldots\\ &\ldots\Rightarrow\left(\Box\left(\mathbf{x}_F\in DeconflArea_{LF}\wedge\Delta x_{LF}^{L\_WP}\geq 0\Rightarrow\circ\left(\mathbf{x}_F\notin BuffArea_{LF}\right)\right)\right)(t).\end{aligned} \tag{B.45}$$

Now, when $\Delta x_{LF}^{L\text{-}WP} \leq 0 \Rightarrow \psi_{LF} \in \left[\frac{\pi}{2}, \frac{3\pi}{2}\right]$, the relative velocity in the leader to follower reference frame is,

$$\Delta \mathbf{v}_{LF}^{LF} = \mathbf{v}_{cmd,F}^{LF} - \mathbf{v}_{L}^{LF} = \left[\begin{array}{c} -\cos\gamma \\ \pm\sin\gamma \end{array} \right] V_{c_F} - \left[\begin{array}{c} \cos\psi_{LF} \\ -\sin\psi_{LF} \end{array} \right] V_{c_L} \Rightarrow ...$$

$$... \Rightarrow \frac{d\left\|\Delta \mathbf{x}_{LF}\right\|}{dt} = u_{LF}^{LF} = -\left(V_{c_F}\cos\gamma + V_{c_L}\cos\psi_{LF}\right) \tag{B.46}$$

Limiting the positions to $\|\mathbf{x}_{LF}\| = d_{D_{LF}}$,

$$\gamma = \frac{\pi}{2} \Rightarrow \frac{d\left\|\Delta \mathbf{x}_{LF}\right\|}{dt} = -V_{c_L}\cos\psi_{LF} \Rightarrow \frac{d\left\|\Delta \mathbf{x}_{LF}\right\|}{dt} \in \left[0, V_{c_L}\right], \tag{B.47}$$

and so,

$$\begin{gathered} \left(\Box\left(\mathbf{x}_F \in DeconflArea_{LF} \wedge \Delta x_{LF}^{L\text{-}WP} \leq 0 \Rightarrow \|\Delta \mathbf{x}_{LF}\| \geq d_{D_{LF}}\right)\right)(t) \Rightarrow ... \\ ... \Rightarrow \left(\Box\left(\mathbf{x}_F \in DeconflArea_{LF} \wedge \Delta x_{LF}^{L\text{-}WP} \leq 0 \Rightarrow \circ\left(\mathbf{x}_F \notin BuffArea_{LF}\right)\right)\right)(t), \end{gathered} \tag{B.48}$$

proving the lemma. □

Lemma B.11. *The system will reach the* TrackWP*, if the follower is in the leader's* Deconfliction Area.

$$\left(\mathbf{x}_F \in DeconflArea_{LF} \rightarrow \{active\, TrackWP\}\right)(t). \tag{B.49}$$

Proof. There are only two reasons for the *Path Deconflicting Algorithm (PDA)* state to be active. There is an interception between both vehicles' line paths or the leader UAV is in the follower conflict area.
Let us take the interception case and assume the follower is on the right side of the leader, $\psi_{LF} \in [0, \pi]$. The interception can only exist if the follower and its waypoint are on different sides of leader. In this situation, the follower velocity command option is $CmdOpt = 1$, resulting in:

$$\mathbf{v}_{cmd,F} = \left[\begin{array}{cc} \cos\gamma & \sin\gamma \\ -\sin\gamma & \cos\gamma \end{array} \right] \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F} \Rightarrow \Delta \mathbf{v}_{LF}^{L\text{-}WP} = \left[\begin{array}{c} -\cos\left(\gamma - \psi_{LF}\right) \\ \sin\left(\gamma - \psi_{LF}\right) \end{array} \right] V_{c_F} - \left[\begin{array}{c} 1 \\ 0 \end{array} \right] V_{c_L}, \tag{B.50}$$

where $\gamma = \arcsin\left(\frac{d_D}{d_{12}}\right)$. When

$$\begin{gathered} \Delta x_{LF}^{L\text{-}WP} \geq 0 \Rightarrow \psi_{LF} \in \left[0, \frac{\pi}{2}\right] \wedge \Delta y_{LF}^{L\text{-}WP} \geq d_{D_{LF}} \therefore \gamma \in \left(0, \psi_{LF}\right] \Rightarrow ... \\ ... \Rightarrow \left\{ \begin{array}{l} u_{cmd_F}^{L\text{-}WP} \in \left[-\left(V_{c_F} + V_{c_L}\right), -V_{c_L}\right) \\ v_{cmd_F}^{L\text{-}WP} \in \left(-V_{c_F}, 0\right] \end{array} \right. , \end{gathered} \tag{B.51}$$

meaning that the follower moves to the back of the leader. And when it crosses $\Delta x_{LF}^{L\text{-}WP} = 0$:

$$\begin{gathered} \Delta x_{LF}^{L\text{-}WP} < 0 \Rightarrow \psi_{LF} \in \left(\frac{\pi}{2}, \pi\right] \wedge \|\mathbf{x}_{LF}\| \geq d_{D_{LF}} \therefore \gamma \in \left(0, \frac{\pi}{2}\right] \Rightarrow ... \\ ... \Rightarrow \left\{ \begin{array}{l} u_{cmd_F}^{L\text{-}WP} \in \left[-\left(V_{c_F} + V_{c_L}\right), V_{c_F} - V_{c_L}\right) \\ v_{cmd_F}^{L\text{-}WP} \in \left[-V_{c_F}, 0\right) \end{array} \right. , \end{gathered} \tag{B.52}$$

which enforces the monotonic movement towards $\Delta y_{LF}^{L\text{-}WP} = 0$, without conflict because of lemma B.10. So:

$$\left(\left\{\mathbf{x}_F : \mathbf{x}_F \in \mathit{DeconflArea}_{LF} \wedge \mathit{Interception} \wedge \Delta y_{LF}^{L\text{-}WP} \geq 0\right\} \rightarrow \circ \left(\Delta y_{LF}^{L\text{-}WP} < 0\right)\right)(t) \tag{B.53}$$

This proves that the follower will cross $\Delta y_{LF}^{L\text{-}WP} = 0$, ceasing the interception. The same can be proved for the left side, and so:

$$\left(\left\{\mathbf{x}_F : \mathbf{x}_F \in \mathit{DeconflArea}_{LF} \wedge \mathit{Interception}\right\} \rightarrow \circ \sim \mathit{Interception}\right)(t) \tag{B.54}$$

Now let us assume that leader is between the follower and its waypoint, and that the waypoint is again on the left side. This doesn't force the follower to be on the right or left side. However, equation B.54 just proved that the follower will eventually be on the left side ($\Delta y_{LF}^{L\text{-}WP} < 0$). On the left side the velocity commands are:

$$\mathbf{v}_{cmd,F} = \begin{cases} \mathbf{v}_{cmd,F} = \begin{bmatrix} \cos\gamma & \sin\gamma \\ -\sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F} & \Delta y_{LF}^{L\text{-}WP} \geq -d_{D_{LF}} \\ \mathbf{v}_{cmd,F} = \begin{bmatrix} \cos\gamma & -\sin\gamma \\ \sin\gamma & \cos\gamma \end{bmatrix} \frac{\mathbf{x}_{FL}}{\|\mathbf{x}_{FL}\|} \cdot V_{c_F} & \Delta y_{LF}^{L\text{-}WP} < -d_{D_{LF}} \end{cases}. \tag{B.55}$$

Taking $0 > \Delta y_{LF}^{L\text{-}WP} \geq -d_{D_{LF}}$,

$$\begin{aligned} &\Delta x_{LF}^{L\text{-}WP} \leq 0 \wedge \psi_{LF} \in \left(\pi, \frac{3\pi}{2}\right] \wedge \|\mathbf{x}_{LF}\| \geq d_{D_{LF}} \therefore \gamma \in \left[\psi_{LF} - \pi, \frac{\pi}{2}\right] \Rightarrow ... \\ &... \Rightarrow \left\{ \begin{array}{l} u_{cmd_F}^{L\text{-}WP} \in (-V_{c_L}, V_{c_F} - V_{c_L}] \\ v_{cmd_F}^{L\text{-}WP} \in (-V_{c_F}, 0] \end{array} \right. . \end{aligned} \tag{B.56}$$

Now with $\Delta y_{LF}^{L\text{-}WP} = -d_{D_{LF}}$,

$$\Delta x_{LF}^{L\text{-}WP} \leq 0 \wedge \psi_{LF} = \pi + \arctan\left(\frac{\Delta y_{LF}^{L\text{-}WP}}{\Delta x_{LF}^{L\text{-}WP}}\right) \therefore \gamma = \psi_{LF} - \pi \Rightarrow \mathbf{v}_{cmd,F}^{L\text{-}WP} = \begin{bmatrix} V_{c_F} - V_{c_L} \\ 0 \end{bmatrix}, \tag{B.57}$$

showing that the follower converges to the line defined by $\Delta y_{LF}^{L\text{-}WP} = -d_{D_{LF}}$. Because the follower can reach the left side of the leader,

$$\begin{aligned} &\left(\left(\Delta y_{LF}^{L\text{-}WP}(t_0) > 0 \wedge \mathbf{x}_{WPF} \in \left\{\mathbf{x} : \Delta x^{L\text{-}WP} \leq 0 \vee \Delta y^{L\text{-}WP} \notin (-d_{D_{LF}}, 0]\right\}\right) \rightarrow ... \right. \\ &\left. ... \rightarrow \mathbf{x}_L \notin \mathit{ConflArea}_{FL}\right)(t). \end{aligned} \tag{B.58}$$

To finish, we have to prove that even if the follower's waypoint position is such that $\mathbf{x}_{WPF} \in \left\{\mathbf{x} : x^{L\text{-}WP} > 0 \wedge y^{L\text{-}WP} \in (-d_{D_{LF}}, 0]\right\}$, the follower will eventually avoid the leader's intrusion. Because the leader is tracking its waypoint, it will take a finite time to reach it. The next waypoint or the subsequent ones will keep the leader tracking the same direction or a different one. If it is different it can "move" the relative position of follower waypoint such that $\mathbf{x}_{WPF} \in \left\{\mathbf{x} : x^{L\text{-}WP} \leq 0 \vee y^{L\text{-}WP} \notin (-d_{D_{LF}}, 0]\right\}$,where equation B.58 is applicable. If, with every subsequent leader waypoints, there is no direction change or it is such that $\mathbf{x}_{WPF} \in \left\{\mathbf{x} : y^{L\text{-}WP} \in (-d_{D_{LF}}, 0]\right\}$ is still true, the leader will at some point surpass the follower waypoint resulting in $\mathbf{x}_{WPF} \in \left\{\mathbf{x} : x^{L\text{-}WP} \leq 0\right\}$. Thus, equation B.58 can be extended to,

$$\left(\Delta y_{LF}^{L\text{-}WP}(t_0) > 0 \rightarrow \mathbf{x}_L \notin ConflArea_{FL}\right)(t)\,. \tag{B.59}$$

The same proof can be applied to the case where the follower's waypoint is on the leader's right side, proving the lemma. □

Theorem B.12. *The overall system satisfies the* Liveness *property, if we assume single integrator holonomic vehicles, if the* PDA *and* TrackWP *states are part of the control system, and if* $(\mathbf{x}_i \notin AvoidArea_{ij}, i \neq j, i, j = 1, 2)(t_0)$ *holds true:*

$$(\mathbf{x}_i \notin AvoidArea_{ij}, i \neq j, i, j = 1, 2 \rightarrow \{\|\Delta\mathbf{x}_{1\_WP}\| \leq d_{WP1} \wedge \|\Delta\mathbf{x}_{2\_WP}\| \leq d_{WP2}\})(t)\,. \tag{B.60}$$

Proof. By lemma B.2, the system doesn't present liveness if it reachs the *CPCAA* state. The system can only reach *CPCAA* state if any vehicle is in the other's *Frontal* or *Buffer Area.* Lemmas B.5 and B.7 together prove that,

$$\begin{aligned}&(\Box(\mathbf{x}_F \in \{BuffArea_{LF} \vee FrontArea_{LF} \backslash CloseFArea_{LF}\} \Rightarrow ...\\&... \Rightarrow \circ(\mathbf{x}_F \notin SafeArea_{LF})))(t)\,.\end{aligned} \tag{B.61}$$

Therefore, *CPCAA* can't be activated if $\mathbf{x}_F(t_0) \notin \{CloseFArea_{LF} \vee SafeArea_{LF}\}$. Now let's prove that the system can reach its goal. For that we just need to prove that it eventually reaches the *TrackWP* state, due to lemma B.4. From lemma B.6, B.8, and B.9 we can conclude that

$$\begin{aligned}&(\mathbf{x}_F \in \{BuffArea_{LF} \vee FrontArea_{LF} \backslash CloseFArea_{LF}\} \rightarrow ...\\&... \rightarrow \circ(\mathbf{x}_F \in DeconflArea_{LF}))(t)\,.\end{aligned} \tag{B.62}$$

Adding the conclusions of lemmas B.10 and B.11:

$$(\mathbf{x}_i \notin AvoidArea_{ij}, i \neq j, i, j = 1, 2 \rightarrow \{active\,TrackWP\})(t)\,. \tag{B.63}$$

Equation B.63 together with lemma's B.2 conclusions prove the theorem (eq. B.60). □

B.2 Properties proof: Model 2 - Double integrator

Theorem B.13. *The overall system satisfies the* Safety *property, if we assume double integrator holonomic vehicles, if the* CPCAA *is part of the control system, and if the next restrictions hold true,*

$$(\|\Delta\mathbf{x}_{12}\| \geq d_{s_{12}})(t_0) \tag{B.64a}$$

$$d_{s_{12}} > d_{c_{12}} - \frac{V_{a_2} + V_{a_1}}{k_a} \ln\left(\frac{V_{c_1} + V_{c_2}}{V_{a_2} + V_{a_1}} + 1\right) + \frac{V_{c_2} + V_{c_1}}{k_a} \tag{B.64b}$$

Proof. If the desired trajectories for UAV 1 and UAV 2 are such that $\left(\Box \left\|\Delta \mathbf{x}_{12}\right\| > d_{s_{12}}\right)(t)$, no collision will occur, because

$$d_{s_{12}} = r_{c_1} \cdot a_1 + r_{c_2} \cdot a_2 > d_{c_{12}}, \tag{B.65}$$

proving

$$\left(\Box \left\|\Delta \mathbf{x}_{12}\right\| > d_{c_{12}}\right)(t)\,. \tag{B.66}$$

If on any part of the trajectories $\left\|\Delta \mathbf{x}_{12}\right\| \leq d_{s_{12}}$, the *CPCAA* state is activated. The minimum distance achievable is defined by:

$$\begin{aligned} d_{min} = \min_{\substack{\psi_1,\psi_2 \\ t \geq t_0}} \Bigg\{ &\left\|\Delta \mathbf{x}_{12}\right\|(t) : \left\|\Delta \mathbf{x}_{12}\right\|(t_0) = d_{s_{12}}, \dot{\mathbf{v}}_i = k_a \left(-\frac{\Delta \mathbf{x}_{ij}}{\left\|\Delta \mathbf{x}_{ij}\right\|} \cdot V_{a_i} - \mathbf{v}_i\right), \ldots \\ &\ldots\, i \neq j, i, j = 1, 2 \Big\}\,. \end{aligned} \tag{B.67}$$

The minimum is achieved with $\frac{d\left\|\Delta \mathbf{x}_{12}\right\|}{dt} = \frac{\langle \Delta \mathbf{x}_{12} | \Delta \mathbf{v}_{12} \rangle}{\left\|\Delta \mathbf{x}_{12}\right\|} = \Delta \mathbf{v}_{12}^{12} = 0$. Lets assume $t_0 = 0$,

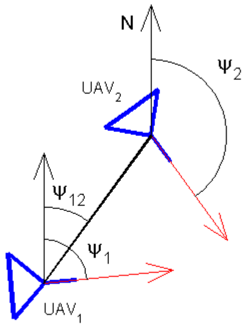


Figure B.8: Relative angle from UAV *2* to UAV *1*

$$\begin{aligned} \Delta \mathbf{v}_{12}(t) &= \mathbf{v}_2 + \left(\frac{\Delta \mathbf{x}_{12}}{\left\|\Delta \mathbf{x}_{12}\right\|} \cdot V_{a_2} - \mathbf{v}_2\right)\left(1 - e^{-t \cdot k_a}\right) - \ldots \\ &\ldots - \mathbf{v}_1 - \left(-\frac{\Delta \mathbf{x}_{12}}{\left\|\Delta \mathbf{x}_{12}\right\|} \cdot V_{a_1} - \mathbf{v}_1\right)\left(1 - e^{-t \cdot k_a}\right) = \ldots \\ &\ldots = \frac{\Delta \mathbf{x}_{12}}{\left\|\Delta \mathbf{x}_{12}\right\|}\left(V_{a_2} + V_{a_1}\right)\left(1 - e^{-t \cdot k_a}\right) + \left(\mathbf{v}_2 - \mathbf{v}_1\right) e^{-t \cdot k_a} \Leftrightarrow \ldots \\ &\ldots \Leftrightarrow \Delta \mathbf{v}_{12}^{12} = \left(V_{a_2} + V_{a_1}\right)\left(1 - e^{-t \cdot k_a}\right) + \ldots \\ &\ldots + \left(V_{c_2} \cdot \cos\left(\psi_2 - \psi_{12}\right) - V_{c_1} \cdot \cos\left(\psi_1 - \psi_{12}\right)\right) e^{-t \cdot k_a}, \end{aligned} \tag{B.68}$$

or

$$\Delta \mathbf{v}_{12}^{12} = V_{a_2} + V_{a_1} + \left[\left(V_{c_2} \cdot \cos\left(\psi_2 - \psi_{12}\right) - V_{a_2}\right) - \left(V_{c_1} \cdot \cos\left(\psi_1 - \psi_{12}\right) + V_{a_1}\right)\right] e^{-t \cdot k_a}, \tag{B.69}$$

where ψ_{12} is the relative angle from UAV *2* to UAV *1* (fig. B.8). And now,

$$\Delta \mathbf{v}_{12}^{12} = 0 \Rightarrow e^{-t\cdot k_a} = -\frac{V_{a_2}+V_{a_1}}{\left(V_{c_2}\cdot\cos\left(\psi_2-\psi_{12}\right)-V_{a_2}\right)-\left(V_{c_1}\cdot\cos\left(\psi_1-\psi_{12}\right)+V_{a_1}\right)} \Leftrightarrow ...$$

$$\Leftrightarrow t_k = \frac{1}{k_a}\ln\left(\frac{\left(V_{c_1}\cdot\cos\left(\psi_1-\psi_{12}\right)+V_{a_1}\right)-\left(V_{c_2}\cdot\cos\left(\psi_2-\psi_{12}\right)-V_{a_2}\right)}{V_{a_2}+V_{a_1}}\right). \tag{B.70}$$

To minimize $\left\|\Delta\mathbf{x}_{12}\right\|(t)$ we need to maximize t_k:

$$\max t_k \Rightarrow \max\left\{\frac{\left(V_{c_1}\cdot\cos\left(\psi_1-\psi_{12}\right)+V_{a_1}\right)-\left(V_{c_2}\cdot\cos\left(\psi_2-\psi_{12}\right)-V_{a_2}\right)}{V_{a_2}+V_{a_1}}\right\} \Rightarrow ...$$

$$\Rightarrow \max\left\{V_{c_1}\cdot\cos\left(\psi_1-\psi_{12}\right)-V_{c_2}\cdot\cos\left(\psi_2-\psi_{12}\right)\right\} \Rightarrow \left\{\begin{array}{l}\psi_2=\pi+\psi_{12}\\ \psi_1=\psi_{12}\end{array}\right. . \tag{B.71}$$

Agreeing with the intuition that the vehicles will get closer if they start in a heads-on configuration. Therefore,

$$d_{min} = \left\|\Delta\mathbf{x}_{12}\right\|(t_k) = \left\|\Delta\mathbf{x}_{12}\right\|(t_0) + \int_0^{t_k} \Delta\mathbf{v}_{12}^{12}\Big|_{\psi_2=\pi+\psi_{12},\psi_1=\psi_{12}} dt, \tag{B.72}$$

and

$$\int_0^{t_k}\left[V_{a_2}+V_{a_1}-\left[V_{c_2}+V_{a_2}+V_{c_1}+V_{a_1}\right]e^{-t\cdot k_a}\right]dt = ...$$

$$... = \left(V_{a_2}+V_{a_1}\right)t_k - \frac{V_{c_2}+V_{a_2}+V_{c_1}+V_{a_1}}{k_a}\left(1-e^{-t_k\cdot k_a}\right), \tag{B.73}$$

and from equation B.70:

$$\int_0^{t_k}\Delta\mathbf{v}_{12}^{12}dt = \frac{V_{a_2}+V_{a_1}}{k_a}\ln\left(\frac{V_{c_1}+V_{c_2}}{V_{a_2}+V_{a_1}}+1\right)-\frac{V_{c_2}+V_{c_1}}{k_a} \Rightarrow ...$$

$$d_{min} = d_{s_{12}} + \frac{V_{a_2}+V_{a_1}}{k_a}\ln\left(\frac{V_{c_1}+V_{c_2}}{V_{a_2}+V_{a_1}}+1\right)-\frac{V_{c_2}+V_{c_1}}{k_a}. \tag{B.74}$$

Therefore

$$d_{c_{12}} < d_{min} \Rightarrow d_{s_{12}} > d_{c_{12}} - \frac{V_{a_2}+V_{a_1}}{k_a}\ln\left(\frac{V_{c_1}+V_{c_2}}{V_{a_2}+V_{a_1}}+1\right)+\frac{V_{c_2}+V_{c_1}}{k_a}, \tag{B.75}$$

proving the theorem. □

B.3 Properties proof: Model 3 - Control delays

Theorem B.14. *The overall system satisfies the* Safety *property, even with control noise, if we assume double integrator holonomic vehicles, if the* CPCAA *is part of the control system, and if the next restrictions hold true,*

$$(\|\Delta \mathbf{x}_{12}\| \geq d_{s_{12}})(t_0) \tag{B.76a}$$

$$d_{s_{12}} \geq d_{c_{12}} - \frac{V_{a_2}+V_{a_1}}{k_a}\ln\left(\frac{V_{c_1}+V_{c_2}}{V_{a_2}+V_{a_1}}+1\right) + (V_{c_2}+V_{c_1})\left(\Delta t+\frac{1}{k_a}\right) \tag{B.76b}$$

Proof. This proof is similar to the one for theorem B.13. If the desired trajectories for UAV 1 and UAV 2 are such that $(\Box\,\|\Delta \mathbf{x}_{12}\| > d_{s_{12}})(t)$, no collision will occur, because

$$d_{s_{12}} = r_{c_1}\cdot a_1 + r_{c_2}\cdot a_2 > d_{c_{12}}, \tag{B.77}$$

proving

$$(\Box\,\|\Delta \mathbf{x}_{12}\| > d_{c_{12}})(t)\,. \tag{B.78}$$

If on any part of the trajectories $\|\Delta \mathbf{x}_{12}\| \leq d_{s_{12}}$, then the *CPCAA* state is activated. The minimum distance achievable is defined by:

$$d_{min} = \min_{\substack{\psi_1,\psi_2\\ t\geq t_0}}\left\{\|\Delta \mathbf{x}_{12}\|(t) : \|\Delta \mathbf{x}_{12}\|(t_0) = d_{s_{12}},\ \ldots\right.$$

$$\left.\ldots\ \dot{\mathbf{v}}_i = k_a\left(ZOH\left(-\frac{\Delta \mathbf{x}_{ij}}{\|\Delta \mathbf{x}_{ij}\|}\cdot V_{a_i}, t, \Delta t\right) - \mathbf{v}_i\right), i\neq j, i,j=1,2\right\}. \tag{B.79}$$

The minimum is achieved with $\frac{d\|\Delta \mathbf{x}_{12}\|}{dt} = \frac{\langle \Delta \mathbf{x}_{12} | \Delta \mathbf{v}_{12}\rangle}{\|\Delta \mathbf{x}_{12}\|} = \Delta \mathbf{v}_{12}^{12} = 0$. Lets assume $t_0 = 0$,

$$\Delta \mathbf{v}_{12}(t) = \mathbf{v}_2 + \left(ZOH\left(-\frac{\Delta \mathbf{x}_{12}}{\|\Delta \mathbf{x}_{12}\|}\cdot V_{a_2}, t, \Delta t\right) - \mathbf{v}_2\right)\left(1-e^{-t\cdot k_a}\right) - \mathbf{v}_1 - \ldots$$

$$\ldots - \left(ZOH\left(-\frac{\Delta \mathbf{x}_{12}}{\|\Delta \mathbf{x}_{12}\|}\cdot V_{a_1}, t, \Delta t\right) - \frac{\Delta \mathbf{x}_{12}}{\|\Delta \mathbf{x}_{12}\|}\cdot V_{a_1} - \mathbf{v}_1\right)\left(1-e^{-t\cdot k_a}\right). \tag{B.80}$$

Let us assume the worst case scenario, where the next update of the *Zero Order Hold (ZOH)* function happens Δt time after the system entered in the *CPCAA* state, and the unupdated commanded velocities directed the aircraft on a heads-on trajectory,

$$\Delta \mathbf{v}_{12}(t) = \begin{cases} \mathbf{v}_2 - \mathbf{v}_1 & 0\leq t\leq \Delta t\\ \frac{\Delta \mathbf{x}_{12}}{\|\Delta \mathbf{x}_{12}\|}\left(V_{a_2}+V_{a_1}\right)\left(1-e^{\Delta t - t\cdot k_a}\right) + (\mathbf{v}_2-\mathbf{v}_1)\,e^{\Delta t - t\cdot k_a} & t>\Delta t \end{cases} \Leftrightarrow \ldots$$

$$\ldots \Leftrightarrow \Delta \mathbf{v}_{12}^{12} = \begin{cases} V_{c_2}\cdot\cos(\psi_2-\psi_{12}) - V_{c_1}\cdot\cos(\psi_1-\psi_{12}) & 0\leq t\leq\Delta t\\ V_{a_2}+V_{a_1}+\left[\left(V_{c_2}\cdot\cos(\psi_2-\psi_{12})-V_{a_2}\right)-\ldots\right. & \\ \left.\ldots\left(V_{c_1}\cdot\cos(\psi_1-\psi_{12})+V_{a_1}\right)\right]e^{\Delta t - t\cdot k_a} & t>\Delta t \end{cases}, \tag{B.81}$$

where ψ_{12} is the relative angle from UAV *2* to UAV *1* (fig. B.8). And now,

$$\Delta \mathbf{v}_{12}^{12} = 0 \Rightarrow e^{\Delta t - t\cdot k_a} = -\frac{V_{a_2}+V_{a_1}}{\left(V_{c_2}\cdot\cos(\psi_2-\psi_{12})-V_{a_2}\right)-\left(V_{c_1}\cdot\cos(\psi_1-\psi_{12})+V_{a_1}\right)} \Leftrightarrow \tag{B.82}$$

We can reuse the result from equation B.70, in theorem B.13, and state that:

$$(\Delta t - t)|_{\Delta \mathbf{v}_{12}^{12}=0} = t_k. \tag{B.83}$$

Furthermore, as in theorem B.13:

$$d_{min} = \left\| \Delta \mathbf{x}_{12} \right\| (t_k) = \left\| \Delta \mathbf{x}_{12} \right\| (t_0) + \int_0^{t_k} \Delta \mathbf{v}_{12}^{12}|_{\psi_2=\pi+\psi_{12},\psi_1=\psi_{12}} \, dt, \tag{B.84}$$

and from equation B.70:

$$\begin{aligned} \int_0^{t_k} \Delta \mathbf{v}_{12}^{12} dt = \frac{V_{a_2}+V_{a_1}}{k_a} \ln\left(\frac{V_{c_1}+V_{c_2}}{V_{a_2}+V_{a_1}}+1\right) - (V_{c_2}+V_{c_1})\left(\Delta t + \frac{1}{k_a}\right) \Rightarrow ... \\ d_{min} = d_{s_{12}} + \frac{V_{a_2}+V_{a_1}}{k_a} \ln\left(\frac{V_{c_1}+V_{c_2}}{V_{a_2}+V_{a_1}}+1\right) - (V_{c_2}+V_{c_1})\left(\Delta t + \frac{1}{k_a}\right). \end{aligned} \tag{B.85}$$

Therefore

$$d_{c_{12}} \leq d_{min} \Rightarrow d_{s_{12}} \geq d_{c_{12}} - \frac{V_{a_2}+V_{a_1}}{k_a} \ln\left(\frac{V_{c_1}+V_{c_2}}{V_{a_2}+V_{a_1}}+1\right) + (V_{c_2}+V_{c_1})\left(\Delta t + \frac{1}{k_a}\right). \tag{B.86}$$

□

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] S. E. Peal, "The soaring of birds," *Nature*, vol. 23, pp. 10–11, 4 November 1880.

[2] R. Nelson, *Flight Stability and Automatic Control.* New York: McGraw-Hill Inc., October 1997.

[3] N. R. Lawrance and S. Sukkarieh, "Wind energy based path planning for a small gliding unmanned aerial vehicle," in *AIAA Guidance, Navigation, and Control Conference*, (Chicago, Illinois), AIAA, Aug. 10-13 2009.

[4] M. J. Allen, "Updraft model for development of autonomous soaring uninhabited air vehicles," in *44rd AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, Nevada), AIAA, 9 - 12 January 2006.

[5] A. N. Vul'fson and O. O. Borodin, "Size distribution of convective thermals in an unstable stratified layer," in *18th Mechanical French Congress*, (Grenoble, France), August 27-31 2007.

[6] Y. J. Zhao, "Optimal patterns of glider dynamic soaring," vol. 25, p. 67–89, Apr 2004.

[7] G. Sachs and O. da Costa, "Dynamic soaring in altitude region below jet streams," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Keystone, Colorado), Aug. 21 - 24 2006.

[8] G. Sachs, "Minimum shear wind strength required for dynamic soaring of albatrosses," *Ibis - The International Journal of Avian Science*, vol. 147, no. 1, pp. 1–10, 2005.

[9] J. Langelaan, "Gust energy extraction for mini and micro uninhabited aerial vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 32, pp. 464–473, July-August 2009.

[10] K. Kesner, "Unmanned aircraft hit military milestone," March 18 2010.

[11] D. J. Edwards, "Implementation details and flight test results of an autonomous soaring controller," in *AIAA Guidance, Navigation, and Control Conference*, (Honolulu, Hawaii), AIAA, Aug. 10-13 2008.

[12] D. J. Edwards, "Aloft places 1st overall in cal valley soaring race," May 23rd 2009.

[13] P. Teixeira and J. Gomes, "Personal communication," Feb 2010.

[14] EarthTrends, "Marine jurisdictions: Coastline length," 2010.

[15] J. Elston and E. Frew, "Reduction of computational complexity for guidance of unmanned aircraft through strong wind fields," in *AIAA Guidance, Navigation, and Control Conference*, Aug. 10-13 2009.

[16] M. V. Ramana, V. Ramanathan, D. Kim, G. C. Roberts, and C. E. Corrigan, "Albedo, atmospheric solar absorption and heating rate measurements with stacked uavs," *Quarterly Journal of the Royal Meteorological Society*, no. 133, pp. 1913–1931, 2007.

[17] G. Bland, T. Miles, M. Logan, M. Motter, and J. Bretthauer, ""mini uavs" for atmospheric measurements," in *AIAA Infotech@Aerospace 2007 Conference and Exhibit*, May 7-10 2007.

[18] J. Reuder, P. Brisset, M. Jonassen, M. Müller, and S. Mayer, "Sumo: A small unmanned meteorological observer for atmospheric boundary layer research," in *IOP Conference Series: Earth and Environmental Science*, vol. 1, IOP Publishing, May 2008.

[19] J. Reuder, P. Brisset, M. Jonassen, M. Müller, and S. Mayer, "The small unmanned meteorological observer sumo: A new tool for atmospheric boundary layer research," vol. 18, pp. 141–147, April 2009.

[20] "Wave glider - http://liquidr.com/technology/wave-glider-concept/."

[21] J. Kim and C. Park, "Wind power generation with a parawing on ships, a proposal," *Energy*, vol. 35, no. 3, pp. 1425 – 1432, 2010.

[22] L. Rayleigh, "The soaring of birds," *Nature*, vol. 27, pp. 534–535, 5 April 1883.

[23] W. R. Manley, "The soaring of birds," *Nature*, vol. 28, p. 198, 28 June 1883.

[24] G. Sachs, J. Traugott, and F. Holzapfel, "In-flight measurement of dynamic soaring in albatrosses," in *AIAA Guidance, Navigation, and Control Conference*, (Toronto, Ontario, Canada), Aug. 2-5 2010.

[25] G. Sachs, J. Traugott, and F. Holzapfel, "Progress against the wind with dynamic soaring: Results from in-flight measurements of albatrosses," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Portland, Oregon, USA), Aug. 8-11 2011.

[26] H. Weimerskirch, O. Chastel, C. Barbraud, and O. Tostain, "Frigatebirds ride high on thermals," *Nature*, vol. 421, pp. 333–334, 23 January 2003.

[27] J. Wharington, *Autonomous control of soaring aircraft by reinforcement learning.* PhD thesis, Royal Melbourne Institute of Technology, Faculty of Engineering, Department of Aerospace Engineering, November 1998.

[28] M. J. Allen, "Autonomous soaring for improved endurance of a small uninhabited air vehicle," in *43rd AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, Nevada), AIAA, 2005.

[29] K. Andersson and I. Kaminer, "Stability of a thermal centering controller," in *AIAA Guidance, Navigation, and Control Conference*, (Chicago, Illinois), AIAA, Aug. 10-13 2009.

[30] A. T. Klesh, P. T. Kabamba, and A. R. Girard, *Optimal Cooperative Thermalling of Unmanned Aerial Vehicles*, vol. 381/2009 of *Lecture Notes in Control and Information Sciences*, pp. 355–369. Berlin / Heidelberg: Springer, October 18 2008.

[31] J. Langelaan, "Some clarifying remarks on gust energy extraction for mini- and micro- uninhabited aerial vehicles," 2010.

[32] Z. Hazen, "Cooperative autonomous soaring techniques." Engineering Independent Study Presentation.

[33] K. Andersson, I. Kaminer, K. D. Jones, V. Dobrokhodov, and D.-J. Lee, "Cooperating uavs using thermal lift to extend endurance," in *AIAA Unmanned Unlimited Conference*, (Seattle, Washington, USA), Apr. 6-9 2009.

[34] M. J. Allen, "Guidance and control of an autonomous soaring uav," in *Technical Memorandum NASA/TM-2007-214611*, NASA Dryden Flight Research Center, February 2007.

[35] N. R. Lawrance and S. Sukkarieh, "Simultaneous exploration and exploitation of a wind field for a small gliding uav," in *AIAA Guidance, Navigation, and Control Conference*, (Toronto, Ontario, Canada), Aug. 2-5 2010.

[36] D. H. Lenschow and P. L. Stephens, "The role of thermals in the convective boundary layer," *Boundary-Layer Meteorology*, vol. 19, pp. 509–532, December 1980.

[37] J. L. Grenestedt and J. R. Spletzer, "Optimal energy extraction during dynamic jet stream soaring," in *AIAA Guidance, Navigation, and Control Conference*, (Toronto, Ontario, Canada), Aug. 2-5 2010.

[38] K. Turkoglu, Y. Zhao, and B. Capozzi, "Real-time insitu strategies for enhancing uav endurance by utilizing wind energy," in *AIAA Guidance, Navigation, and Control Conference*, (Chicago, Illinois), AIAA, Aug. 10-13 2009.

[39] S. M. LaValle, *Planning Algorithms.* Cambridge, U.K.: Cambridge University Press, 2006.

[40] V. Klein and E. A. Morelli, *Aircraft System Identification: Theory and Practice.* American Institute of Aeronautics and Astronautics, Inc., 2006.

[41] J. W. Langelaan, N. Alley, and J. Neidhoefer, "Wind field estimation for small unmanned aerial vehicles," in *AIAA Guidance, Navigation, and Control Conference*, (Toronto, Ontario, Canada), Aug. 2-5 2010.

[42] J. Parle, "Preliminary dynamic soaring research using a radio control glider," in *42nd AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, Nevada, USA), Jan. 5-8 2004.

[43] FAA, *Aviation Weather.* Aviation Supplies & Academics, Inc., October 1 1975.

[44] *U.S. Military Specification MIL-F-8785C*, 5 November 1980.

[45] C. D. C. Jr., "Thermal soaring of birds," *American Scientist*, vol. 50, pp. 180–209, March 1962.

[46] R. B. Stull, *An Introduction to Boundary Layer Meteorology.* Norwell, Massachusetts: Kluwer Academic Publishers, 1988.

[47] D. A. Konovalov, "On the structure of thermals," in *12th OSTIV Congress*, (Alpine, USA), OSTIV, 1970.

[48] F. Irving, *The Paths of Soaring Flight.* World Scientific Publishing Company, Jan. 2004.

[49] F. Irving, *The paths of soaring flight.* Imperial College Press, 1999.

[50] A. N. Vul'fson and O. O. Borodin, "The size distribution function for mixed-layer thermals in the convective atmosphere," in *IUTAM Symposium on Hamiltonian Dynamics, Vortex Structures, Turbulence* (A. V. Borisov, V. V. Kozlov, I. S. Mamaev, M. A. Sokolovskiy, G. M. L. Gladwell, and R. Moreau, eds.), vol. 6 of *IUTAM Bookseries*, pp. 227–236, Springer Netherlands, August 27-31 2008.

[51] J. W. Langelaan, "Long distance/duration trajectory optimization for small uavs," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Hilton Head, South Carolina, USA), Aug. 20-23 2007.

[52] S. Short, "Birth of american soaring flight: A new technology," *AIAA Journal*, vol. 43, pp. 17–27, January 2005.

[53] S. A. Jenkins and J. Wasyl, "Optimization of glides for constant wind fields and course headings," *Journal of Aircraft*, vol. 27, pp. 632–638, July 1990.

[54] B. Z. Cybyk, B. McGrath, T. M. Frey, D. G. Drewry, and J. F. Keane, "Unsteady urban airflows and their impact on small unmanned air system operations," in *AIAA Guidance, Navigation, and Control Conference*, (Chicago, Illinois), AIAA, Aug. 10-13 2009.

[55] P. B. MacCready Jr., "Optimum airspeed selector," *Soaring*, pp. 10–11, January-February 1958.

[56] J. H. Cochrane, "Maccready theory with uncertain lift and limited altitude," tech. rep., Graduate School of Business, University of Chicago, Chicago, Illinois, May 4 1999.

[57] W. Schuemann, "The price you pay for mccready speeds," 1972.

[58] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, vol. 1. New York: Academic Press, June 11 1979.

[59] J. L. Casti, "Recent developments and future perspectives in nonlinear system theory," *SIAM Review*, vol. 24, pp. 301–331, Jul 1982.

[60] R. Hermann and A. J. Krener, "Nonlinear controllability and observability," *IEEE Transactions on Automatic Control*, vol. 22, pp. 728 – 740, Oct 1977.

[61] X. S. Zhou and S. I. Roumeliotis, "Robot-to-robot relative pose estimation from range measurements," *IEEE Transactions on Robotics*, vol. 24, pp. 1379–1393, Dec. 2008.

[62] M. V. Jakuba, *Stochastic Mapping for Chemical Plume Source Localization with Application to Autonomous Hydrothermal Vent Discovery.* PhD thesis, MIT Dept. of Aeronatics and Astronautics, Cambridge, Massachusetts, February 2007.

[63] D. Metzger and J. Hedrick, "Optimal flight paths for soaring flight," in *Proceedings of the 2nd International Symposium on the Technology and Science of Low Speed and Motorless Flight*, AIAA, 1974.

[64] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents).* Cambridge, Massachusetts: The MIT Press, 3rd ed., 2005.

[65] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications.* Artech House, 2004.

[66] E. A. Morelli, "Real-time aerodynamic parameter estimation without air flow angle measurements," in *2011 AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Portland, Oregon, USA), 8-11 Aug. 2011.

[67] N. R. J. Lawrance and S. Sukkarieh, "A guidance and control strategy for dynamic soaring with a gliding uav," in *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pp. 3632 –3637, may 2009.

[68] A. Bicchi and L. Pallottino, "Conflict avoidance in networked mobility systems: towards decentralization," in *IEEE International Conference on Robotics and Automation*, (Orlando, Florida, USA), IEEE Robotics and Automation Society, 15-19 May 2006.

[69] E. Williams, "Airborne collision avoidance system," in *SCS '04: Proceedings of the 9th Australian workshop on Safety critical systems and software*, (Darlinghurst, Australia, Australia), pp. 97–110, Australian Computer Society, Inc., 2004.

[70] D. Stipanovic, G. Inalhan, R. Teo, and C. J. Tomlin, "Decentralized overlapping control of a formation of unmanned aerial vehicles," *Automatica*, vol. 40, no. 8, pp. 1285–1296, 2004.

[71] A. Richards, J. Bellingham, M. Tillerson, and J. P. How, "Coordination and control of multiple UAVs," in *in AIAA Guidance, Navigation, & Control Conference, Monterey, CA*, 2002.

[72] W. B. Dunbar and R. M. Murray, "Model predictive control of coordinated multi-vehicle formation," in *41th IEEE Conf. on Decision and Control*, 2002.

[73] R. Olfati-Saber, W. B. Dunbar, and R. M. Murray, "Cooperative control of multi-vehicle systems using cost graphs and optimization," in *American Control Conference*, December 14-17 2003.

[74] L. P. L. F. Giulietti and M. Innocenti, "Autonomous formation flight," *IEEE Control Systems Magazine*, vol. 20, no. 6, pp. 34–44, 2000.

[75] M. Jardin, "Ideal free flight through multiple aircraft neighboring optimal control," in *American Control Conference, 2000. Proceedings of the 2000*, vol. 4, pp. 2879–2885 vol.4, 2000.

[76] F. Borrelli, T. Keviczky, and G. J. Balas, "Collision-free UAV formation flight using decentralized optimization and invariant sets," in *43rd IEEE Conference on Decision and Control, Paradise Island, Bahamas*, December 14-17 2004.

[77] C. Tomlin, G. J. Pappas, and S. Sastry, "Conflict resolution for air traffic management: A study in multiagent hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, pp. 509–521, April 1998.

[78] G. Hoffmann and C. Tomlin, "Decentralized cooperative collision avoidance for acceleration constrained vehicles," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pp. 4357–4363, Dec. 2008.

[79] G. M. Hoffmann, H. Huang, S. L. Waslander, and C. J. Tomlin, "Quadrotor helicopter flight dynamics and control: Theory and experiment," in *AIAA Guidance, Navigation, & Control Conference, Monterey, CA*, (Hilton Head, SC), August 2007.

[80] L. Pallottino, A. Bicchi, and E. Frazzoli, "Probabilistic verification of decentralized multi-agent control strategies: a case study in conflict avoidance," in *American Control Conference, 2007. ACC '07*, pp. 170–175, July 2007.

[81] E. A. Lee, "Soft walls - modifying flight control systems to limit the flight space of commercial aircraft," in *Technical Memorandum UCB/ERL M01/31.*, 15 September 2001.

[82] A. Cataldo, "Control algorithms for soft walls," in *Technical Memorandum UCB/ERL M03/42*, January 21 2004.

[83] J. Jackson, R. Bencatel, Z. Hasan, and A. Girard, "Stochastic patrolling and collision avoidance for two UAVs in a base defense scenario," in *AIAA Guidance, navigation, and Control Conference, Chicago, USA*, (Chicago, Illinois), AIAA, Aug. 10-13 2009.

[84] S. Ortmann, "Defence and freedom: Airspace deconfliction," July 22 2009.

[85] S. Owicki and L. Lamport, "Proving liveness properties of concurrent programs," *ACM Transactions on Programming Languages and Systems*, vol. 4, no. 3, pp. 455–495, 1982.

[86] N. Rescher and A. Urquhart, *Temporal logic.* New York: Springer-Verlag, 1971.

[87] L. Lamport, ""sometime" is sometimes "not never": on the temporal logic of programs," in *POPL '80: Proceedings of the 7th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, (New York, NY, USA), pp. 174–185, ACM Press, 1980.

[88] F. B. Schneider, "Decomposing properties into safety and liveness using predicate logic," Tech. Rep. 87-874, Cornell University, Ithaca, NY, USA, October 1987.

[89] D. Swaroop and J. K. Hedrick, "String stability of interconnected systems," *Automatic Control, IEEE Transactions on*, vol. 41, pp. 349 –357, Mar. 1996.

[90] C. N. Phana and H. H. Liu, "Dynamic mapping of forest fire fronts using multiple unmanned aerial vehicles," in *AIAA Guidance, Navigation, and Control Conference*, (Toronto, Ontario, Canada), Aug. 2-5 2010.

[91] M. J. Vachon, R. J. Ray, K. R. Walsh, and K. Ennix, "F/A-18 aircraft performance benefits measured during the autonomous formation flight project," in *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, (Monterey, California), Aug. 5-8 2002.

[92] J. M. Fowler and R. D'Andrea, "Distributed control of close formation flight," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 3, (Las Vegas, Nevada, USA), pp. 2972 – 2977, December 2002.

[93] K. Kondak, M. Bernard, F. Caballero, I. Maza, and A. Ollero, "Cooperative autonomous helicopters for load transportation and environment perception," in *Advances in Robotics Research* (T. Kröger and F. M. Wahl, eds.), pp. 299–310, Springer Berlin Heidelberg, 2009.

[94] I. Maza, K. Kondak, M. Bernard, and A. Ollero, "Multi-uav cooperation and control for load transportation and deployment," *Journal of Intelligent & Robotic Systems*, vol. 57, pp. 417–449, 2010.

[95] D. J. Lee, K. Y. Kam, I. Kaminer, D. P. Horner, A. Healey, S. P. Kragelund, K. Andersson, and K. D. Jones, "Wireless communication networks between distributed autonomous systems using self-tuning extremum control," in *AIAA Unmanned Unlimited Conference*, (Seattle, Washington), April 6-9 2009.

[96] J. S. Brinker and J. M. Barker, "Formation taxi and flight of the joint unmanned combat air system (J-UCAS) X-45A vehicles," in *2005 AIAA Guidance, Navigation, and Control Conference and Exhibit*, (San Francisco, CA, USA), pp. 1–10, 15-18 Aug. 2005.

[97] Z. Zheng, S. C. Spry, and A. R. Girard, "Leaderless formation control using dynamic extension and sliding control," in *17th IFAC World Congress*, (Seoul, Korea), July 6-11 2008.

[98] F. Sharifi, B. W. Gordon, and Y. Zhang, "Decentralized sliding control of cooperative multi-agent systems subject to communication delays," in *AIAA Guidance, Navigation, and Control Conference*, (Toronto, Ontario, Canada), Aug. 2-5 2010.

[99] Y. Gu, B. Seanor, G. Campa, M. R. Napolitano, L. Rowe, S. Gururajan, and S. Wan, "Design and flight testing evaluation of formation control laws," *IEEE Transactions On Control Systems Technology*, vol. 14, pp. 1105–1112, Nov. 2006.

[100] S. Bayraktar, G. E. Fainekos, and G. J. Pappas, "Hybrid modeling and experimental cooperative control of multiple unmanned aerial vehicles," tech. rep., December 10 2004.

[101] S. Leven, J.-C. Zufferey, and D. Floreano, "Dealing with mid-air collisions in dense collective aerial systems," *Journal of Field Robotics*.

[102] C. E. Miller, A. W. Tucker, and R. A. Zemlin, "Integer programming formulation of traveling salesman problems," *J. ACM*, vol. 7, no. 4, pp. 326–329, 1960.

[103] F. Borrelli, D. Subramanian, A. U. Raghunathan, L. T. Biegler, and T. Samad, "A comparison between mixed integer programming and nonlinear programming techniques for 3d conflict resolution of multiple aircraft," in *Department of Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, Tech. Rep.*, 2003.

[104] S. Waydo, J. Hauser, R. Bailey, E. Klavins, and R. Murray, "UAV as a reliable wingman: A flight demonstration," *Control Systems Technology, IEEE Transactions on*, vol. 15, pp. 680–688, july 2007.

[105] D. Swaroop, J. K. Hedrick, C. C. Chien, and P. Ioannou, "A comparision of spacing and headway control laws for automatically controlled vehicles," *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, vol. 23, no. 1, pp. 597–625, 1994.

[106] M. J. Allen, J. Ryan, C. Hanson, and J. Parle, "String stability of a linear formation flight control system," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Monterey, California, USA), Aug. 5-8 2002.

[107] B. Vaglienti, M. Niculescu, and J. Hammitt, *Piccolo Simulator.* Cloud Cap Technology (www.cloudcaptech.com), Hood River, Oregon, October 14 2011.

[108] M. Zanmiller and D. Miley, *Piccolo Software-in-Loop (SiL) Setup Guide.* Cloud Cap Technology (www.cloudcaptech.com), Hood River, Oregon, October 14 2011.

[109] T. Oliveira, G. Cruz, E. Marques, and P. Encarnação, "A test bed for rapid fight testing of uav control algorithms," in *Research, Development and Education on Unmanned Aerial Systems Conference*, (Seville, Spain), Nov.-Dec. 2011.

[110] B. Vaglienti, R. Hoag, M. Niculescu, J. Becker, and D. Miley, *Piccolo User's Guide*. Cloud Cap Technology (www.cloudcaptech.com), Hood River, Oregon, v2.1.2 ed., October 12 2011.

[111] T. Oliveira, "Desenvolvimento de metodologias para o projecto integrado de sistemas de condução e controlo de veículos aéreos não tripulados," September 2009.

[112] T. Oliveira and P. Encarnação, "Ground target tracking for unmanned aerial vehicles," in *AIAA Guidance, Navigation, and Control Conference*, (Toronto, Ontario, Canada), Aug. 2-5 2010.

[113] R. Bencatel, "Thermal localization," in *International Conference on Autonomous and Intelligent Systems*, (Póvoa de Varzim, Portugal), June 2010.

[114] R. Bencatel, A. Girard, M. Faied, and J. Sousa, "Shear wind estimator," in *2011 AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Portland, Oregon, USA), 8-11 Aug. 2011.

[115] R. Bencatel, M. Faied, J. Sousa, and A. Girard, "Formation control with collision avoidance," in *50th IEEE Conference on Decision and Control and European Control Conference*, (Orlando, Florida, USA), December 12-15 2011.

[116] E. Pereira, R. Bencatel, J. ao Correia, L. Félix, G. Gonçálves, J. Morgado, and J. ao Sousa, "Unmanned air vehicles for coastal and environmental research," in *10th International Coastal Symposium - ICS 2009*, (Lisboa, Portugal), April 13-18 2009.

[117] P. Almeida, R. Bencatel, G. Gonçálves, and J. ao Sousa, "Multi-uav integration for coordinated missions," in *Robótica 2006 national festival*, (Braga, Portugal), June 2006.

[118] P. Almeida, R. Bencatel, G. Gonçálves, J. ao Sousa, E. Pereira, J. Morgado, and D. Coutinho, "Apollo framework for unmanned air vehicle systems: experimental results," in *UVS UAV 2007 conference*, (Paris, France), UVS, June 2007.

[119] P. Almeida, R. Bencatel, G. Gonçálves, J. ao Sousa, and C. Ruetz, "Experimental results on command and control of unmanned air vehicle systems," in *IFAC Symposium on Intelligent Autonomous Vehicles 2007*, (Toulouse, France), IFAC, September 2007.

[120] R. Bencatel, J. ao Correia, J. ao Sousa, G. Gonçálves, and E. Pereira, "Video tracking control algorithms for unmanned air vehicles," in *ASME Dynamic Systems and Control Conference 2008*, (Ann Arbor, USA), ASME, October 2008.