Long Range UAS Operations for Volcanic Monitoring

By

BENJAMIN J. SCHELLENBERG

Department of Aerospace Engineering
UNIVERSITY OF BRISTOL

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of DOCTOR OF PHILOSOPHY in the Faculty of Engineering.

JULY 2020

Word count: 44,000 (approx.)

Abstract

This thesis considers the effective long range operation of fixed-wing Unmanned Aerial Systems (UAS) for volcanic monitoring, focusing on the automation of plume sensing, route planning, and plume re-interception. Ash sampling from within volcanic plumes provides vital data for ash dispersion models, the outputs of which are used to manage aviation hazards and monitor risks to populations. Volcán de Fuego is a highly active stratovolcano in Guatemala with a summit altitude of 3800 m Above Mean Sea Level (AMSL), and just 40 km from Guatemala City International Airport. This thesis describes a robust method of operating UAS in hazardous environments such as that found around Volcán de Fuego, presenting Beyond Visual Line of Sight (BVLOS) fixed-wing flights for the collection of airborne ash. Reaching ranges in excess of 10 km, and altitudes of 5000 m AMSL, the automation of these UAS missions remains highly desirable in order to maximise efficiency and reduce operator workload. Key components of the presented missions include the identification and sensing of volcanic plumes, efficient flight path planning, and coordinating plume re-interception for the collection of comparative ash samples. A metric is presented for detecting plumes autonomously onboard using lightweight and low power sensors, based on a series of flight tests. The development and implementation of an onboard Real-Time Trajectory Planner (RTTP) is also discussed, in the context of calculating efficient climb paths to the summit area of Volcán de Fuego. The real-time element of this system means obstacles may be avoided and the route calculated mid-flight, and running the system onboard the aircraft effectively increases the UAV's level of autonomy. Finally, a Coordinated Plume Interception (CPI) scheme is presented for the efficient repeat sampling of a volcanic plume some prescribed distance downwind. A series of routing options are developed and discussed, considering their dependence on wind conditions and the impact of practical BVLOS flight zone limits. The themes of autonomy, efficiency, and volcanic monitoring run throughout this thesis, developing technical and practical solutions to real-world problems.

Acknowledgements

I would like to thank everyone who has supported me over the last three and a half years. In particular, thank you to my supervisor, Dr. Tom Richardson, for his support and guidance throughout the various phases of this PhD. Thanks also to my secondary supervisors; Professor Arthur Richards for his guidance on all things optimisation, and Dr. Matt Watson for his volcanology expertise and excellent company on field trips. Thanks too to Professor Jim Freer, for his practical and rigorous approach to scientific challenges and writing, and to Professor Mark Lowenberg for conducting my annual reviews. I would like to thank everyone from the Bristol Flight Lab for their company and support, in particular thank you Dr. Colin Greatwood and Dr. Kieran Wood for helping me find my feet in the early days, and to Rob Clarke for patiently helping me make sense of various coding issues. Thanks also to Matt Edwards, whose ever-present friendship has kept me company throughout this journey. I would also like to acknowledge the work of the ArduPlane developers; this project would not have been possible without their commitment to open source sharing and development. My trips to Guatemala have been thoroughly rewarding, a fact for which I must thank Matt Pervis of GeoTravel Guatemala whose practical support and good company made many a task easier. Thank you too to Gustavo Chigna and the whole team at INSIVUMEH, without whom this project would not have happened and whose support has been invaluable. I would also like to acknowledge and thank Ivan Velasquez of DGAC (Guatemala) who granted BVLOS flight permissions across all five trips, the CASCADE program grant for funding these campaigns, and the EPSRC for funding my PhD.

Thank you to my parents, Joanna and David, whose commitment to helping others has always been a source of inspiration, and to my brothers, Louis and Oliver, for their continued love and support. Thanks also to the Stewart family, who have been there for me throughout and provided a safe haven when needed.

Finally, my most sincere gratitude to my partner and closest friend, Madeleine Stewart, whose love and support has seen me through all the highs and lows of this journey. It means the world to me, thank you.

AUTHOR'S DECLARATION

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: . [redacted] .... DATE: 21st January 2021

TABLE OF CONTENTS

List of Tables

List of Figures

Figure **Page**

NOMENCLATURE

alt Altitude

lat Latitude

lon Longitude

AMP Avian Meteorological Package

AMSL Above Mean Sea Level

BVLOS Beyond Visual Line of Sight

CAA Civil Aviation Authority

CDR Conflict Detection and Resolution

CPI Coordinated Plume Interception

DEM Digital Elevation Model

DGAC Dirección General de Aeronáutica Civil, Guatemalan National Aviation Authority

EA Evolutionary Algorithm

ESC Electronic Speed Controller

ESP Eagle Sensor Package

FAA Federal Aviation Administration

FBWA Fly By Wire (A)

FPV First Person View

GA Genetic Algorithm

GCS Ground Control Station

GPS Global Positioning System

HITL Hardware In The Loop

IC Internal Combustion

IMU Inertial Measurement Unit

INSIVUMEH Guatemalan Institute of Seismology, Volcanology, Meteorology, and Hydrology

LiIon Lithium Ion

LiPo Lithium Polymer

mAh milliampere hour

MAV Micro Air Vehicle

MTOM Maximum Take Off Mass

NAA National Aviation Authority

PSD Particle Size Distribution

PSO Particle Swarm Optimisation

PWM Pule-Width Modulation

RH Relative Humidity

RRT Rapidly expanding Random Tree

RTL Return to Launch

RTTP Real-Time Trajectory Planner

SA Simulated Annealing

SEM Scanning Electron Microscopy

SITL Software In The Loop

SRTM Shuttle Radar Topography Mission

STOL Short Take-Off/Landing

sUAS Small Unmanned Aerial System

TECS Total Energy Control System

UAS Unmanned Aerial Systems

UAV Unmanned Aerial Vehicle

VLOS Visual Line of Sight

VTOL Vertical Take-Off/Landing

WP Waypoint

CHAPTER 1

INTRODUCTION

Unmanned Aerial Systems (UAS) are a rapidly developing area of aerospace, commerce, and research. Expanding technical capability, the lack of humans onboard, and developing regulations mean UAS can be used for an ever growing range of applications that would otherwise be costly, dangerous, or outright impossible. The term UAS refers to an entire unmanned system, including both Unmanned Aerial Vehicles (UAVs) and operator Ground Control Stations (GCS). UAVs, also known as Drones, come in various types and sizes; from small electric multi-rotors [7] to large petrol-powered fixed-wing planes [5], and are used for tasks ranging from filming and mapping to high-level surveillance and warfare [14–17].

The role of UAS in academia includes use-type cases, where commercially available platforms with little to no modifications are used to collect research data, and technical system development. Research for the latter focuses on exploring and developing UAS capabilities, including areas such as airframe design, development of novel control systems, and associated flight testing. Examples of use-type cases include using multi-rotor UAVs to study giraffe running kinematics [18], and using fixed-wing UAVs for hydrological wetland mapping [19]. Examples of technical academic UAS development include the design and flight testing of vertical take-off/landing (VTOL) platforms [20], and designing multi-rotor control systems for robotic arm payloads [21].

An emerging area of multidisciplinary work combines these research types; developing technical UAS performance and capability in order to collect scientific research data. This thesis describes one such project, exploring and developing the technical limits of UAS within the context of volcanic monitoring. Extensive real-world operations are presented, testing novel UAS navigation methods, collecting scientific datasets, and developing methods for the effective use of UAS as research tools in extreme environments.

1.1 Research Motivation

Scientific research around active volcanoes is inherently dangerous, and often limited to sites with low level activity or short high risk visits with rescue support. UAS have been used increasingly in the volcanic monitoring and research community to reduce human risk and enable work around some of the more active volcanoes across the globe. Typically highly eruptive, these volcanoes have historically been restricted to macro-scale and satellite based monitoring methods. The unprecedented access offered by UAS has led to the emergence of new and exciting research topics in the fields of volcanology and engineering [22]. UAS have been used in this context as part of early-warning hazard alert systems [23] and data sources for atmospheric modelling tools [24]. They have also been used for aerial volcanic imaging [25] and airborne sampling of volcanic outputs [26].

This thesis describes the development of a fixed-wing UAS, and an accompanying operations methodology, for research and monitoring use in hazardous volcanic environments. Specifically, the work presented here concerns the highly active Volcán de Fuego, Guatemala, and involves extensive UAS operations at long range (> 9 km) and high altitude (up to 5000 m / 18,000 ft Above Mean Sea Level (AMSL)). Targeting ash collection from within volcanic plumes, this work seeks to address UAS performance and automation within the context of robust and repeatable volcanic monitoring in extreme environments.

1.2 Smart Volcanic Monitoring

Datasets required from sites such as Volcán de Fuego include visual imagery of the summit area for mapping the crater and assessing changes, sampling gas emissions for chemical analysis, and collecting ash samples for atmospheric and geological modelling. Various UAS types could be used to collect these data, typically involving some level of human involvement. This might include manually piloting the aircraft or planning automated missions. Manually piloting UAVs can involve high pilot workloads and be difficult to repeat consistently for scientific experiments, however conventional autopilot missions limit aircraft to following pre-determined flight plans. The unpredictable nature of active volcanoes often requires operational flexibility, either targeting or avoiding eruptions depending on mission aim, eruption magnitude, and eruption frequency. There is therefore a value associated with UAS-based monitoring systems which combine the flexibility of manually piloted operations with the repeatability and robustness of automated missions.

The dominant volcanological motivation for the work presented here is the sampling of ash from within Volcán de Fuego's plumes, described in detail in Chapter 2. Volcanic ash is known to be highly damaging to manned aircraft, meaning that even beyond the ballistics zone of Volcán de Fuego there are significant risks involved with traditional aerial monitoring techniques. The airborne collection of volcanic ash therefore presents a challenge to which UAS are a

natural solution, offering extensive in-plume sampling configurations at a range of downwind locations while maintaining low human risk levels. Automating these UAS builds experimental robustness and repeatability, reduces the impact of natural human error, and creates a number of interesting engineering challenges. This 'smart' volcanic monitoring capability is the common theme throughout the work presented here.

1.3 Research Objectives

The research objectives of this PhD are:

- To develop a reliable and repeatable method of operating UAVs for volcanic monitoring
- To design a practical automated UAV-based volcanic plume sensing technique
- To design and implement a real-time trajectory planner for UAVs in volcanic environments
- To design an autonomous volcanic plume re-interception scheme for UAV-based comparative sample collection

Combining these objectives builds towards a system that is capable of rapid and effective deployment at new volcanic sites, reducing the time, effort, and cost associated with building operations experience in order to achieve scientific targets. Each of these objectives is described in further detail below.

1.3.1 Research Aim: Method Development

Simulation has been used extensively as a tool for development between testing phases, however the value of this work lies in the practical application of the developed systems. Real-world UAS operations are rarely trivial and are often challenged by regulations, environmental and atmospheric conditions, and the experimental robustness of systems. Additional challenges around Volcán de Fuego include the physical effects of high altitude on flight performance and the impact of long range operations on communications and safety protocols. Success in the field therefore relies on a robust and effective method of operating UAS; resilient to the extreme natural disturbances found around volcanoes and the harsh take-off and landing conditions. This means reliably achieving mission aims while also assuring adequate levels of safety for the system, the operators, and third parties.

In comparison to manually flown missions, automated flights offer an unparalleled repeatability; a waypoint mission plan may be re-flown on separate occasions resulting in a consistent flight path with comparatively low workloads for the flight crew. By controlling for navigational and human error in this way, missions may be flown with a confidence that the aims will be achieved. This reliability also contributes to building a robust safety case; once a mission profile is

developed and tested it then becomes a safe reference point for operational decisions. For example, these mission plans might take into account the location of local villages and the flight track may be developed such that there is always a safe landing option, with the preferred location being the operations base. Any deliberate deviation off the mission plan, i.e. for intercepting a volcanic plume, can be recovered to a known level of safety and predictability by returning the aircraft to the predetermined waypoint path.

While not a primary engineering aim of this project, the development of a reliable operations methodology was intrinsic to the safe and successful use of UAS in volcanic environments. This refined naturally across five trips to Volcán de Fuego, between February 2017 and April 2019, with each trip building on the progress made in the last.

1.3.2 Research Aim: Plume Sensing

Automated volcanic plume sampling requires some metric by which the plume can be recognised by the system. The first primary engineering aim was to develop a sensing metric which clearly identifies plumes without requiring human input. This would allow autonomous plume tracking and sampling methods to be explored using a consistent and scientifically rigorous approach.

The size and weight of the sensors used for this plume detection metric must be kept to a minimum. Aerospace design and aircraft performance is dominated by component size and mass, and the same principles apply to UAVs. The work in this area therefore combines lightweight sensors such as accelerometers, temperature, and humidity sensors to characterise flight through a plume and develop an appropriate metric for future use.

1.3.3 Research Aim: Real-Time Trajectory Planning

The terrain immediately surrounding active volcanoes is often steep and unsafe, forcing the location of UAS operations points to lower altitudes. UAVs are therefore often required to gain significant altitude in order to carry out volcanic sampling tasks, and this is often done in an automatic flight mode. These flight plans are typically designed manually and carefully planned so that the aircraft flies efficiently given the predicted flight conditions. The scale of operations around Volcán de Fuego means environmental and atmospheric flight conditions can change significantly within the time frame of a single flight.

Flight efficiency within the context of this thesis can be characterised by the amount of energy used to reach the summit region of Volcán de Fuego, given a consistent launch location, and is sought to be maximised in order to extend the time available in the region of interest. Variation in mission aims and paths, largely due to the unpredictable nature of volcanic emissions, means it can be difficult to quantify mission efficiency reliably. The terrain around Fuego means the energy used to reach the summit area is heavily influenced by climb performance, so parameters such as time spent above summit altitude and cumulative altitude gain may be used as indicators of overall mission efficiency. Flight duration may be used alongside these parameters for added

context, as can the Air-Distance. This is calculated by integrating the airspeed across the flight and is equivalent to the distance covered over the ground in zero wind conditions.

The second distinct engineering aim was to develop an optimisation based route planning algorithm for application in a long range, high altitude gain environment. Route planning is a popular area of research, however literature indicates that most publications only test algorithms in a simulated world space, with a majority of them limited to two dimensions. Real-world implementation of such a system involves increasing the UAS autonomy level, removing human input from the route planning process and limiting operator involvement to ground-based system monitoring.

1.3.4 Research Aim: Coordinated Plume Interception

Sampling ash from within a volcanic plume generates useful data for volcanologists, as described in Chapters 2 and 7. Furthermore, collecting multiple ash samples during a single flight allows comparison between proximal plumes (i.e. close to the crater, considered here as < 500 m) and distal plumes (i.e. far from the crater, considered here as 5 - 20 km). These data suggest a relationship between airborne ash particle size distribution and time post-eruption; a useful input for ash dispersion models and related aviation management tools. Coordinating plume interception such that these samples may be collected autonomously was the third primary engineering aim of this work.

1.4 Summary of Contributions

The main contributions of the work presented in this thesis, including publications, are summarised below:

1. The development of a reliable method for UAS operations in challenging environments, including long range and high altitude gain mission profiles, and 'hot & high' conditions. A conference paper titled 'BVLOS Operations of Fixed-Wing UAVs for the Collection of Volcanic Ash Above Fuego Volcano, Guatemala', written by the Author, has been published on this topic in the American Institute of Aeronautics and Astronautics SciTech conference proceedings [27] (DOI: 10.2514/6.2020-2204).

2. The design of a metric for detecting volcanic plumes using lightweight and low power UAV-based sensors. A journal paper titled 'Remote sensing and identification of volcanic plumes using fixed-wing UAVs over Volcán de Fuego, Guatemala', written by the Author, has been published on this topic in the Journal of Field Robotics [28] (DOI: 10.1002/rob.21896).

3. The design, development, and flight testing of a real-time onboard trajectory planner for long range and high altitude gain fixed-wing UAV operations. A journal paper titled

'Onboard Real-Time Trajectory Planning for Fixed-Wing Unmanned Aerial Vehicles in Extreme Environments', written by the Author, has been published on this topic in the MDPI journal Sensors [29] (DOI: 10.3390/s19194085).

4. The design of a coordinated plume interception scheme, for autonomously sampling a single volcanic plume multiple times while maximising efficiency. Variables here include wind speed and direction, and a prescribed distance between sample locations. This scheme incorporates restrictions placed on the permissible flight zone and energy modelling, and has been developed using software in the loop testing for real-world implementation. A conference paper titled 'Automated Real-Time Volcanic Plume Interception for UAVs' on this topic, written by the Author, has been accepted to the American Institute of Aeronautics and Astronautics SciTech 2021 conference (DOI pending).

1.5 Thesis Structure

This thesis is structured as follows. Chapter 2 presents the background for the project, including a literature review which considers the use of UAVs around volcanoes, long range UAS operations, and path planning optimisation for UAVs. Chapter 3 presents a summary of the operating environment and the hardware used throughout the project, and Chapter 4 describes the development of the systems and methods used across all five trips. Chapter 4 also presents flight and mission efficiency metrics, demonstrating a significant development as the project progressed. Chapter 5 presents a method of plume sensing and identification, addressing the first primary engineering aim. Chapter 6 introduces and develops a real-time trajectory planner, including real-world testing and subsequent development. Chapter 7 presents a coordinated plume interception scheme, designed for integration with the UAV command system that has been developed alongside the real-time trajectory planner. Finally, Chapter 8 concludes and summarises the thesis, suggesting areas of future work.

CHAPTER 2

BACKGROUND & LITERATURE REVIEW

This chapter gives background information and builds depth for this project. This includes discussing Unmanned Aerial Systems (UAS) in a general sense, volcanic research, the applicability and usefulness of UAS in volcanology, and introducing Volcán de Fuego, Guatemala. The literature review will also consider publications relevant to the core technical chapters of the thesis, including plume sensing, real-time trajectory planning and optimisation, and rendezvous problems for coordinated plume interception.

2.1 Unmanned Aerial Systems

This section describes the dominant features of various UAS types and components, including examples, use cases, advantages and disadvantages. These include fixed-wing aircraft, multi-rotor aircraft, VTOL fixed-wing aircraft, and ground control stations. The terms UAV and Drone will hereafter be used in reference to the aircraft, while UAS refers to the entire system. This includes any ground station and additional componentry.

There are records of powered model aircraft from as early as 1871, with radio controlled models first appearing in 1937 [30]. Flying radio controlled model aircraft gained popularity amongst hobbyists by the 1970s [31], however they were not fitted with autopilots and onboard computing until much later. UAVs were developed predominantly by the military, with uses ranging from surveillance missions to guiding missiles [32–34]. It wasn't until the mid-2000s, when the United States Federal Aviation Administration (FAA) first issued a commercial drone permit, that demand for industrial drone applications began to grow. Government agencies began by exploring the use of UAVs for disaster relief and industrial inspections, however they were generally limited by public perception and acceptance [34, 35]. With strong military connotations, and public concerns including safety and privacy, it purportedly took 10 years

before the popularity of drones began to show in the commercial and consumer markets, with Amazon first announcing plans for delivery drones in 2013 [36].

In a military context, the term 'drone' often refers to a large unmanned fixed-wing vehicle, however civilian use of the term relates mostly to small multi-rotors equipped with cameras for recreational and professional filming. Multi-rotors require an autopilot unit for control and stability, however these autopilots can also be used on small fixed-wing aircraft for radio control augmentation or full automation. The most popular consumer drone brand of multi-rotors are the Chinese manufacturers, DJI. The company began in 2006 when a student at the Hong Kong University of Science and Technology started developing an autopilot system that would help prevent his fixed-wing model aircraft from crashing [37]. This technology was applied to the development of multi-rotors and in 2013, the same year Amazon released plans for delivery drones, DJI had their first major success with the Phantom quad-copter [38].

2.1.1 Fixed-Wing UAVs

Fixed-wing UAVs resemble an aeroplane, with forward thrust generated by a propeller or jet and lift generated by fixed-wings. The relationship between these aircraft and multi-rotors is similar to that between aeroplanes and helicopters; for a given size or capacity they offer a greater range and endurance, however they require space to take-off and land. Four examples are shown in Figure 2.1 with accompanying information in Table 2.1.

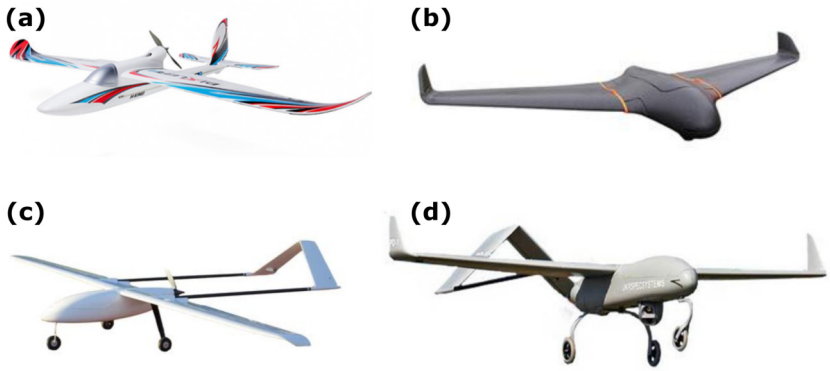

Figure 2.1: Fixed-wing UAV examples: (a) Bixler2 [2], (b) Skywalker [3], (c) Albatross [4], (d) PD1 [5]. More info needed here for each of them.

Aircraft A is a Bixler 2, commonly used in the model flying community as a trainer aircraft. The next aircraft shown is a Skywalker X8, a delta-wing foam-based airframe with a wingspan of 2.1 m, which is used extensively in this project (see Chapter 3). Aircraft C is an Applied Aeronautics Albatross, a composite airframe with an inverted-V shaped empenage. These three aircraft can all be powered electrically, however aircraft D (the PD-1) is has an Internal Combustion (IC) motor for extended operations [5]. The PD-1 is marketed predominantly for long-range military

Table 2.1: Details of the aircraft pictured in Figure 2.1

Manufacturer/Aircraft	MTOM	Wingspan	Notes
Hobbyking/Bixler 2	0.75 kg	1.5 m	Training aircraft, foam, electric
Skywalker/X8	4.5 kg	2.1 m	Delta-wing, foam, electric
Applied Aeronautics/Albatross	10 kg	3.0 m	Fibreglass, long-endurance, electric
UKRSPEC Systems/PD-1	40 kg	4.7 m	Internal combustion engine

surveillance operations, and while the Albatross and Skywalker may be used similarly they are also be used for scientific testing and research.

2.1.2 Multi-rotor UAVs

Multi-rotor UAVs rely on multiple rotating propellers for lift, and a differential in propeller thrust for translational flight. These platforms usually have 4, 6, or 8 motors; termed quadrotors, hexacopters, and octocopters respectively. While climb performance and endurance are often limited, compared to fixed-wing vehicles, these platforms offer flexibility in terms of payload and required operating space. Most commonly used for filming, they are able to take-off and land vertically and can be flown precisely in order to perform specific manoeuvres. Figure 2.2 shows four examples, with accompanying information in Table 2.2.

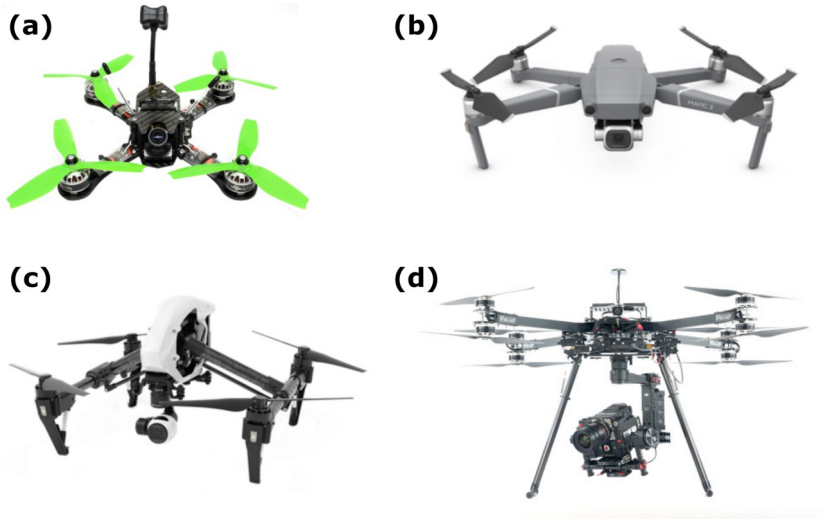

Figure 2.2: Multi-rotor UAV examples: (a) Mini FPV race quad (custom) [6], (b) DJI Mavic 2 Pro [7], (c) DJI Inspire 1 [8], (d) Vulcan UAV Raven [9].

All multi-rotors shown in Figure 2.2 are powered electrically, as is typical for these platforms. Aircraft A is an example of a custom-built First Person View (FPV) race quad, typically lightweight and flown for recreational use. Aircraft B and C are DJI quadcopters (the Mavic 2 Pro and Inspire 1 respectively), developed specifically for filming. While the Mavic 2 Pro weights < 1 kg and packs

Table 2.2: Details of the aircraft pictured in Figure 2.2

Manufacturer/Aircraft	**MTOM**	**Notes**
Custom/FPV Race Quad	< 0.25 kg	Quadcopter, for racing and acrobatics
DJI/Mavic Pro 2	0.9 kg	Quadcopter, for filming
DJI/Inspire 1	3 kg	Quadcopter, for filming
Vulcan UAV/Raven	25 kg	Octocopter, for heavy payloads/filming

down for portability, the Inspire is better suited to professional film-makers, with a 360° pan modular camera. Aircraft D is the Vulcan UAV Raven - an octocopter built for lifting and operating heavy cameras and equipment. The ability to customise aircraft such as the Raven means they are useful in a great variety of situations; from filming with specific camera equipment to deploying scientific sensors.

Single-rotor (helicopter) UAVs also exist, however they tend to be more complex than multi-rotors, and are used less frequently. Mimicking manned helicopters, single-rotor UAVs combine a fixed-RPM (revolutions per minute) and a relatively complex swashplate mechanism for control. Comparatively, multi-rotors usually have simple variable-RPM motors with fixed-pitch props. Some examples of single-rotor UAVs are discussed in this text, however they tend to be early UAV projects which involve large platforms for heavy lifting, powered by IC engines [39].

2.1.3 VTOL UAVs

The space requirements for fixed-wing UAV launch and recovery have driven the development of Vertical Take-Off/Landing (VTOL) platforms. The configurations of these aircraft can vary significantly, including tilt-rotor style platforms that mimic manned aircraft like the Osprey [40], and the combination of fixed-wing and quadrotor UAVs [41]. The additional componentry required for hovering a fixed-wing UAV inherently adds mass to the system, penalising flight performance outside the take-off and landing phases. This means the limits of performance for these aircraft are usually restricted significantly, however their development is an area of ongoing research [42].

2.1.4 Regulations & Long Range UAV Operations

Aircraft with a Maximum Take-Off Mass (MTOM) greater than 20 kg in the United Kingdom are required to satisfy Civil Aviation Authority (CAA) airworthiness requirements [43]. Most aircraft used for recreational and research purposes have a MTOM significantly less than this limit, and can be classified as small UAS (sUAS). Vehicles weighing less than 7 kg have historically been further classified as Micro Air Vehicles (MAVs).

Generally, the operation of UAVs can be split into Visual Line of Sight (VLOS) and Beyond Visual Line of Sight (BVLOS). Figure 2.3 describes some of the key limits for VLOS operations, as

defined by the CAA in the United Kingdom. In addition to these distance limits, VLOS rules also include proximity limits to members of the public, built up areas, and roads [44]. Multi-rotors excel in VLOS operations; their ability to hover and travel slowly, coupled with vertical take-off and landing, makes them ideal platforms for satisfying these regulations.

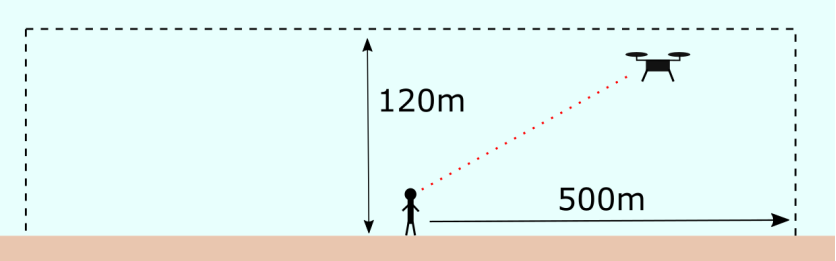

Figure 2.3: Some of the key features of VLOS UAV operations; 500 m lateral separation, 120 m maximum flight altitude, and maintaining visual contact (shown by the red dotted line).

BVLOS operations involve flying without visual contact and are often necessary for search and rescue, extended mapping, disaster relief, and scientific research missions. Just as multi-rotors are well suited to VLOS operations, fixed-wing UAVs are often better suited to BVLOS missions. BVLOS operations are not permitted in the United Kingdom, and many other countries, without approval from the appropriate authority, evidence of a valid need to fly, and an extensive safety case; should the aircraft malfunction or an unexpected situation occur there must be protocols in place to minimise damage or injury to third parties.

Safe UAV operations revolve around the development and utilisation of safety cases, from setting up fail-safes on the throttle hardware to autopilot home-point allocation and operational decision making. Flying within VLOS means the safety case can be built around maintaining visual contact and manually taking control should the need arise. This is not possible for BVLOS flight, hence additional care must be taken to ensure a robust safety case is in place. The primary risks around UAV flight are collision with aircraft, and loss of control leading to collision with people or property. Secondary risks involve loss of equipment, sensors, and data. BVLOS safety cases are usually built by proving the reliability of the aircraft and onboard systems, and specifying areas of operation where collision with members of the public is either not possible or extremely unlikely. BVLOS flights require coordination with the CAA in the United Kingdom, and the appropriate National Aviation Authority (NAA) overseas.

UAVs are typically not permitted to fly in the airspace shared by general manned aviation, also known as unrestricted airspace. Early literature regarding BVLOS operations focus on the integration of UAS with manned airspace, and goes back as early as 2011 [45, 46]. European BVLOS operations were first authorised in 2012; Colomina and Molina review UAS for photogrammetry and remote sensing in their 2014 paper, linking early BVLOS flights to potential future applications [47]. Sweatte *et al.* published a whitepaper in 2015, introducing UAS as a tool for Arctic science and recommending a series of guidelines for generic BVLOS operations in

the Arctic [48]. While still limited, the popularity of BVLOS UAS in research has continued to increase since then. Johnson *et al.* published an evaluation of a UAS traffic management system in 2017, including extensive BVLOS flights using a variety of aircraft types [49]. On the 31st of July 2019, a team from the Kansas Integration Pilot Program flew a groundbreaking BVLOS mission in unrestricted airspace in the United States, having been granted a waiver by the FAA [50]. The team was required to prove that the automated collision avoidance system was safer than human intervention; this system was entirely on-board and used a combination of computer vision and a breakthrough low-cost radar system.

Further review of long range flight testing is included in Section 2.2.3, and the safety measures taken in this project are summarised in Chapter 4.

2.2 Volcanic Research & UAS

2.2.1 Motivation

Volcanoes are significant point source emitters of ash, gas, and aerosols to the atmosphere. Volcanic ash, particularly fine ash measuring < 63 microns, has been proven to have a profound effect on the operation of both civil and military aviation [51]. Airspace used by aviation is managed operationally using two primary tools: advection-diffusion dispersion models [52], driven by volcanic and meteorological source terms, and satellite remote sensing [53, 54]. The first offers a forecast, which is vital for flight planners, although forward projection in time increases uncertainty. The latter, whilst also being uncertain, provides snapshots that may be used to validate predictions at a known time but with some notable latency. Both techniques require some information *a priori*; in particular the Particle Size Distribution (PSD) and shape of volcanic ash [55, 56]. Atmospheric conditions (both cloud and thermal effects) can make satellite retrievals challenging. Combined with significant uncertainty around ash PSD and shape, this reduces the accuracy of satellite retrievals and the reliability of dispersion model predictions [57]. Representative ash samples are therefore critical for the effective use of both airspace and industry management tools, in order to respond to volcanic events accordingly. Furthermore, the knowledge gained by analysing such samples can help mitigate the effects of large eruptions on the local population by better informing local decision-makers.

2.2.2 Volcán de Fuego, Guatemala

Volcán de Fuego (henceforth referred to as Fuego) is an active volcano in Guatemala with a summit altitude approximately 3800 m Above Mean Sea Level (AMSL) (Figure 2.4). Short-term, small explosions (also described as 'strombolian') occur multiple times every hour, with larger, more sustained, eruptions every 3-5 weeks (referred to as a 'paroxysm'). Occasional major eruptions (e.g. June 2018, September 2012, October 1974) cause large-scale changes to the area's topography and have a significant impact on the local population. These eruptions can be

characterised by large pyroclastic flows; a fast moving and devastating mixture of hot lava, ash, and gases, reaching speeds in excess of 22 m/s (80 km/h) [58].

Figure 2.4: A map view in Google Earth showing the location of Volcán de Fuego within Guatemala, including neighbouring countries for context [10].

Over one million people live within a 30 km radius of Fuego [59]. With the country's international airport less than 45 km away, the volcano poses a significant risk to the Guatemalan population, aviation industry, and economy. Whereas some active (non-strombolian) volcano craters are relatively accessible, Fuego ejects large projectiles on ballistic trajectories. This makes it a particularly challenging volcano to monitor and from which to collect samples. Ground-based ash sampling consists of collecting ash that has fallen out of the plume, however these samples cannot have a PSD that is representative of the in-plume PSD. This is because smaller ash particles will remain in the plume for longer. The only way to collect a sample with a PSD that is representative of the entire plume is to collect ash from within the plume itself. Given the inherent risks to manned aviation around Fuego, and ash-dense air in general [60], this sampling challenge poses an interesting engineering problem to which UAS are a possible solution.

Figure 2.5 shows a typical view of Fuego and the visible ashy parts of the plumes from two eruptions. There are a number of challenges associated with collecting ash from within these plumes, discussed in detail in Chapter 4. An important secondary use for UAS around Fuego involves imaging the summit area; photographs and Digital Elevation Maps (DEMs) yield useful information for volcanologists and the Guatemalan government. These updates tend to show the fullness of the crater, suggesting the stage of the volcano in the paroxysm cycle, and the location

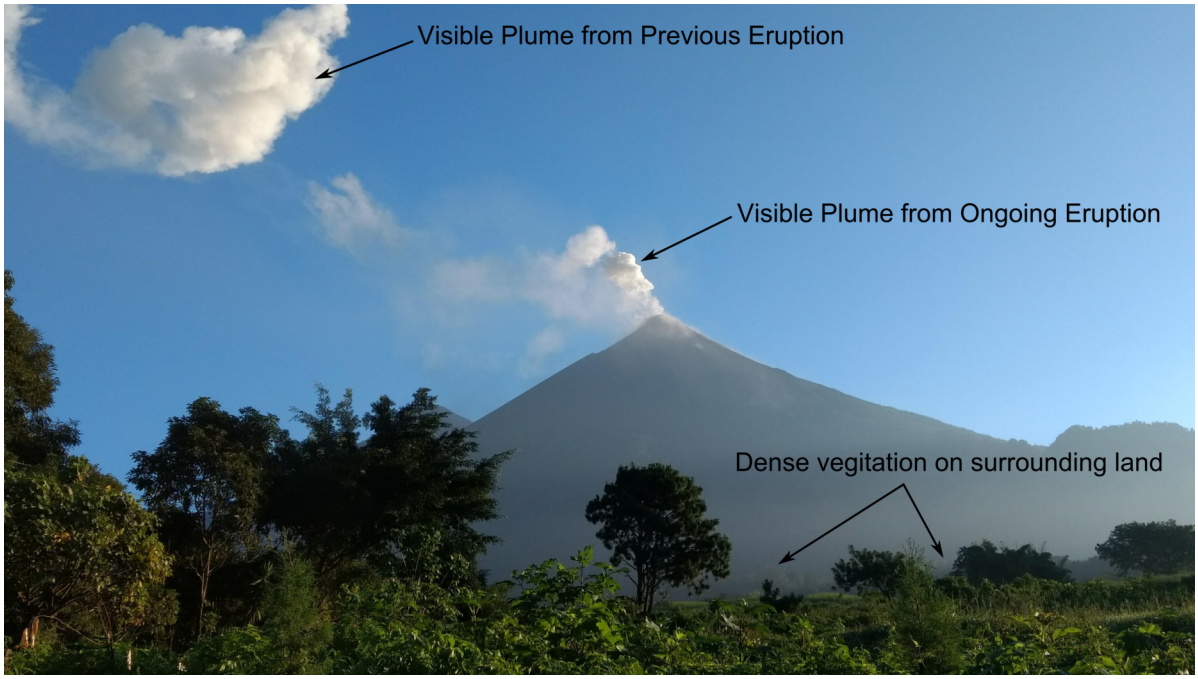

Figure 2.5: A typical view of Fuego. Note the visible plumes of two small eruptions, and the thick brush in the foreground.

of vents within it, suggesting which valleys (or 'barrancas') may have imminent lava-flows.

2.2.3 UAVs for Volcanic and Environmental Research

UAVs enable a wide range of operations that would otherwise not be possible due to restrictions on human safety and physical limitations, and do so at a lower cost than most alternative means of gathering data. Wegener, Everaerts, and Klemas present early papers on the potential use of UAVs for scientific missions [61–63], with Everaerts highlighting the usefulness of UAVs for scientific remote sensing and mapping due to the low cost and ease of access to platforms. Klemas compares UAVs for remote sensing over coastal areas with the manned aircraft that were previously used at great expense. UAVs give excellent access to extreme environments, as demonstrated by di Stefano who used a multi-rotor to monitor and gather data over the Lusi mud crater in Indonesia [64].

Ramanathan *et al.* pioneered the use of lightweight fixed-wing UAVs to investigate atmospheric phenomena, equipping three large fixed-wing aircraft with aerosol, soot, and solar radiation instrumentation to measure the heating effect of brown carbon layers between 0.5 km and 3 km above the Indian Ocean [65]. They were flown in stacked formation for simultaneous data collection from different altitudes. The additional development of a turbulent flux measurement system for UAVs by Thomas *et al.* [66] enabled further investigations above the Indian Ocean, including the discovery that solar absorption by black carbon particles suppresses

boundary layer turbulence [67]. As Thomas notes in the 2012 paper, instruments for taking representative measurements of atmospheric phenomena often need to be out of the boundary layer of the aircraft. These instruments can be a significant component of the overall take-off weight, leading to challenges with balance and flight control when mounted as required.

Villa *et al.* reviewed the use of UAVs for air quality studies, assessing their suitability and challenges in their implementation [68], and later present a developed UAV based system for air pollution measurements [69]. Although a fundamentally desirable method of collecting air data, their paper highlights a few key challenges, including a necessity for sensors that are both small and suitably accurate, and overcoming policies and regulations such as import systems and airspace rules. These factors translate into most other applications of UAVs, particularly overseas. Schuyler and Guzman review the various UAV platforms available for sampling tropospheric gases, and conclude that fixed-wing aircraft with a wingspan under 3 m and payload < 5 kg provide the best compromise between cost and convenience [70]. Of note here is the value associated with the hours of preparation and testing in UAS research and development, a factor which is often ignored by those planning to use UAVs for novel research applications.

Greatwood *et al.* used multi-rotor UAVs to collect high-altitude non-volcanic scientific data, sampling the atmosphere above Ascension Island at approximately 3000 m AMSL [1, 71]. Large (25 kg) fixed-wing UAVs were developed by Altstädter to observe ultrafine particle distributions within the atmospheric boundary layer, however they only flew to a maximum altitude of 1080 m AMSL [72].

UAVs offer outstanding new sampling opportunities for volcanic emissions [73]. The Handbook of Unmanned Aerial Vehicles discusses the application of UAVs to volcanic monitoring and sampling, giving some examples of early projects and noting that their results are still preliminary and limited due to the harshness of the environment [74]. Gas and ash sensors have been miniaturised enough to be flown on UAVs, are able to provide real-time information and, in principle, can capture and retrieve samples. UAVs allow direct and remote measurements much closer to volcanic vents than previously possible at volcanoes such as Fuego, leading to better plume characterisation. Autopilot hardware and navigation algorithms improve repeatability which should serve to better validate satellite and ground-based remote observations. Different UAV types may be used depending on sensor requirements and research aims. For example, sampling a single location near a vent would be suited best to a multi-rotor UAV, however missions involving sampling at various distal ranges or with high temporal variation are better suited to fixed-wing UAVs.

The first reported use of UAVs over volcanoes for scientific purposes was by McGonigle *et al.* in 2007 [39], who used a helicopter UAV with a payload capacity of 3 kg at La Fossa crater in Italy. Amici *et al.* reported development of multi-rotor and fixed-wing UAV flights over Stromboli volcano in Italy; however, they focus on the successes of a hexacopter with a thermal camera onboard [75]. Jordan presented a short summary of UAVs in geology, focusing on small

multirotors such as DJI Phantoms and the challenges surrounding their use in scientific fieldwork [76]. Gomez and Kennedy presented 3D volcanic plume data from a successful trip to Vanuatu in October 2016, using a DJI Phantom 3 fitted with a Real Time Kinetic (RTK) global positioning system (GPS) [25].

Whereas Fuego's activity involves large ballistics, active volcanoes such as Turrialba and Masaya in Central America are safer to approach, hence operating sites can be found relatively close to the craters [77]. Stix *et al.* discuss the suitability of multi-rotors as platforms for static sensor placement (e.g. for gas measurements), when minimal altitude gains and short ranges are required. These works consider volcanoes which are relatively accessible and note flight times between 12 and 15 minutes. Volcanoes like Fuego are dangerous to approach, meaning that flight times must be longer or cruise speeds higher in order to reach them from a ground station located at a safe distance. Fixed-wing UAVs are a natural solution to this issue, as demonstrated by Pieri *et al.* who flew them over Turrialba while also flying 'tethersonde' meteorological balloons for data verification [24]. Primarily sensing gases, these experiments validated the use of UAVs in scientific data collection over volcanoes, specifically delta-wing aircraft.

Further publications relating to UAS and volcanoes since the start of this project show that it is a growing area of academia and UAV use. Lai *et al.* demonstrate the effectiveness of geothermal exploration using UAS in their 2018 paper [78], and BVLOS flight of a fixed-wing UAV for volcanic monitoring is presented by Flores *et al.* in their 2019 paper [79]. Furthermore, Wood *et al.* recently presented a method of determining the 3D structure of volcanic plumes using UAS imagery (2019) [80].

2.2.4 UAVs at Volcán de Fuego

With an estimated summit altitude of 3800 m AMSL, any UAV that flies over Fuego must be equipped with the appropriate components to allow for safe, reliable, and useful operations at high altitude. Paredes *et al.* investigated the effects of altitude on UAV performance, validating theories generated from background aeronautical knowledge with fixed-wing flights above 5500 m AMSL [81]. Paredes concludes that energy requirements increase with target altitude, decreasing flight time for a fixed energy capacity.

INSIVUMEH is the Guatemalan Institute of Seismology, Volcanology, Meteorology, and Hydrology. They work to monitor sites such as Fuego in order to better understand the underlying activity, and work with other government departments to recommend evacuations when and where they are necessary. Prior to this project, the summit of Fuego had last been closely observed in 2012 by manned aircraft during a period of low activity (G. Chigna, personal communication). Although macro-scale behaviour of the volcano is already monitored with seismometers, cameras, spectrometers, and satellites, close-range behaviour and topography of the summit changes regularly and remains largely unknown, as does the make-up of volcanic emissions. This project primarily targets the development of UAS for collecting in-plume ash, however also collected

imagery of the summit area when possible to help inform INSIVUMEH regarding the state of the crater.

The Guatemalan NAA (DGAC) is responsible for the safety and organisation of aircraft under 5486 m (18,000 ft) AMSL [82]. In order to conduct the initial BVLOS missions around Fuego, a safety case was taken to DGAC outlining the purpose of the work, the zones of operation, and (self-imposed) off-limit areas. Subsequent trips involved amendments to the documentation and flight zones, correlating with development of the team's operational capability and the evolution of the research aims. These details are described in Chapter 4, where the development of the flight operations method across the project is addressed.

2.2.4.1 Engineering Aims

The conditions and environment around Fuego, combined with the volcanological motivation for using UAS, generate a series of interesting engineering challenges. Reliable real-world UAV use requires a robust method of operations and a refined overall system; these vary according to the nuances of the location and project aims. The first aim is therefore to develop an effective method of operating around Fuego, which can be measured by combining metrics relating to flight performance and mission success.

Fundamentally, it is possible to use a UAV to achieve the minimum volcanological aims by flying manually. This restricts the use of the system to those with the necessary skills and is likely to be heavily influenced by human error. Conventional UAV avionics systems can help automate some elements of the mission, however a fully automated solution is preferable and provides an interesting challenge.

Automated UAV-based plume sampling requires some method of plume detection. This forms the basis of the first technical engineering challenge, of which some background is given below and details are presented in Chapter 5. Another technical challenge, and one that is applicable to other applications, is the navigation of the aircraft from the launch location to the area of interest. Natural limits are placed on possible operating locations around the volcano, by way of a danger zone around the crater and terrain conditions beyond that. The altitude gain and lateral distance to the summit is therefore significant in the problem considered here, meaning that any inefficiencies in manual route designs are amplified. Additionally, the time it takes to execute missions on this scale introduces the possibility of changing atmospheric conditions. Chapter 6 describes the development of an onboard real-time trajectory planner. This project applies some of the findings from trajectory planning in the context of computer science research to a practical UAS case. The final technical challenge, discussed in Chapter 7, involves the re-interception of plumes some time after initial sampling. This area of research developed as a result of early experiments and method development, and involves coordinating the UAV location and plume sampling system such that two useful and physically distanced ash samples are collected from within the same plume. The comparison of these data would give the change in PSD with time,

indicating the rate at which ash drops out of the plume, which in turn is a useful input for ash-dispersion models.

2.3 Automating Plume Detection

The lack of literature in this area suggests that UAS have not previously been used to detect volcanic plumes, manually or automatically. There are, however, a number of publications that relate to the use of UAS around forest fire and gas leak plumes.

Montes and Letheren *et al.* developed an algorithm in 2014 to track forest fire plumes using a multi-rotor [83], and presented the results of a series of flight tests in 2016 [84]. Their work focused on finding the source of the fire, processing real-time data on the ground station. Specifically, Letheren's system detected the plume by measuring carbon dioxide (CO_2), and used a bio-inspired search algorithm to track upwind to find the source. Conversely, the source of the plume at Fuego is known and the aim is to fly through discrete plumes to sample from within. Furthermore, given that they sometimes rely on near-zero forward speeds, the algorithm(s) used by Montes and Letheren are only applicable for multi-rotors. Despite this, their work on gas sensing remains relevant to volcanic plume sensing.

Forest fire plume detection was also considered by Ramanatha *et al.* in 2016, who used a hexacopter to sense forest fires and send out SMS text alerts [85]. Rather than sense CO_2, an MQ2 gas sensor was used to detect stannic oxide (SnO_2). Plume tracking was not addressed as the flight plans were predetermined.

Airborne optical and thermal remote sensing was reviewed by Allison *et al.* in 2016, with a focus on wildfire detection and monitoring [86]. While not exclusively focused on plumes, Allison *et al.* review various aircraft types; they note a need for the onboard sensors to match the intelligence and scope of the vehicle. Heat, light, and smoke are considered the main indicators of wildfires. As justified in Chapter 5, the heat difference between clear air and distal volcanic plumes is expected to be negligible. While wildfires might be detected by sensing the presence of light or flickering, it may be possible to apply a similar but converse technique to the detection of plumes due to their naturally dark colour. However, as the distal range increases the visibility of volcanic plumes tends to decrease.

Sieber *et al.* presented a tracking system for hazardous plumes around accident sites, using swarms of small multi-rotors [87]. Sensing the plume using air quality measurements, multi-rotors surrounded the area after initial detection and thereby gave first-responders an immediate visual reference point and collected data for later analysis. The system was tested indoors and showed promising results. It may be practical to deploy swarm technology around some volcanoes, but the scale of operations around Fuego mean that small multi-rotors such as those used by Sieber *et al.* are not applicable to plume tracking in this instance. The use of air quality sensors reiterates the findings of the fire-plume research above, which also use some form of gas/aerosol

detectors to sense plumes.

Gas sensing was also carried out by Yang *et al.*, who presented a method of finding the source of natural gas leaks using a quadcopter, however they used laser backscatter rather than a specific gas sensor [88]. Presenting extensive flight testing, Yang *et al.* concluded that near real-time localisation of the leak source can be found using their system, which was designed such that the UAV could be deployed at the site of a reported gas leak. The quadcopter was therefore likely to be flown within VLOS and with a strong telemetry link, unlike the aircraft around Fuego. The sensor was mounted underneath their aircraft, monitoring laser backscatter off the ground to determine the presence of methane. There is some evidence that backscatter could be used to sense clouds, using the reflectivity of the clouds themselves [89], however there is no evidence yet that this would work for volcanic plumes.

More recently, Liu *et al.* used a large multi-rotor UAV to monitor volcanic emissions over Volcán Villarrica in Chile [26]. The platform was a Volcan 'Black Widow'; with a maximum take-off mass of 16 kg and a maximum flight time of 13 minutes. The system was clearly able to detect the plume, however the data required post-processing in order to produce meaningful results. A similarly sophisticated gas sensing system could be used for plume sensing around Fuego, however the 0.8 kg payload used in Liu's work limits the aircraft to relatively close-range operations. While it is important to robustly detect the plume, the primary aim of the mission remains the collection of ash. This means that an ash collection mechanism must form the primary payload, and secondary payloads must not unnecessarily compromise the flight performance or efficiency of the system.

The methodology used for the sensing and identification of volcanic plumes in this project is discussed in depth in Chapter 5, and the ash collection mechanism is presented in Chapter 4.

2.4 Real-Time Trajectory Planning

As introduced in Section 2.2.4.1, optimisation of the routes flown to reach Fuego's summit is likely to result in efficiency savings and reduce the impact of human error. The ability to account for environmental factors, such as turbulence or areas of strong wind, would allow the aircraft to avoid these areas in future route plans, thereby increasing flight efficiency and decreasing extreme loads on the aircraft. A natural solution to often poor long range telemetry links between the ground control station and aircraft is to conduct the route optimisation calculations onboard. Rapid temporal variation in atmospheric conditions means these calculations must take place in real-time, therefore the system is referred to here as a Real-Time Trajectory Planner (RTTP).

At the core of this section of work is the trajectory planning algorithm and its real-world implementation. A number of teams have investigated fixed-wing UAV path planning; however, none of this previous work involved real-world use. Some related real-world testing has taken place with multi-rotor UAVs [84], however these aircraft are innately different to fixed-wing

platforms as they have the ability to hover and gain computational time. Operating around Fuego requires route solutions that span a large space in 3D, operating BVLOS.

The practical use of successful 3D trajectory optimisation is a natural area of development in the UAV community; increasing UAV flight performance, developing UAV automation, and increasing UAV operators' efficiency. The application of reliable real-time path planning methods would allow UAV teams to save the time spent developing efficient and effective flight plans in order to complete specific tasks. It is also of interest to the air traffic conflict detection and resolution (CDR) community [90], both for integration of UAVs into existing airspace, and development of UAV-specific air traffic management. Kuchar and Yang review CDR modelling methods in their 2000 paper, concluding that future work in the area must address issues of uncertainty, multiple potential conflicts, and coordination, as well as pilot and controller acceptance [90]. A number of these issues have since been addressed by those active in UAV trajectory planning development, and the usefulness of any successful real-time trajectory algorithms to air traffic CDR is clear.

The RTTP problem is broken down in the following section; generic path and trajectory planning is introduced first, then a series of optimisation methods are discussed before concluding with a review of literature relating to real-time UAV navigation.

2.4.1 Path and Trajectory Planning

Motion planning algorithms are surveyed within the context of UAV guidance by Goerzen *et al.* [91], and simulated implementation of some of these methods are demonstrated by Beard and McClain [92]. Goerzen *et al.* introduce the area and terminology before surveying work in path planning without differential constraints, and trajectory planning with differential constraints. Path planning, without consideration of differential constraints such as the equations of motion, can generally be applied to simplistic models in 2D. Some methods of note are:

- **Voronoi Roadmap** A skeleton-based method which finds the minimum distance path on a map while maintaining maximum separation from obstacles. It is computationally efficient in 2D, and Howlet *et al.* have presented a way of expanding the method to 3D but only applied it to multi-rotors [93].

- **Potential Field Methods** Based on the concept of assigning a potential function to free space, this method simulates additional forces on the vehicle attracting it to the goal point. Various modifications have been made to the original method [94], including the addition of Gradient Descent and searches such as the A*.

Three-dimensional trajectory planning problems for UAVs can be considered as dynamically constrained, and therefore differential constraints must be applied [91]. Simple kinematic models can be represented by implementing constraints, however this leads to conservative models and

relatively poor predictions of vehicle performance. Kong *et al.* show that the performance criteria for an aircraft (in this case a multi-rotor UAV) have a significant effect on real-world performance, and present a method of mapping routes in complex terrain considering the difference between proposed aircraft performance and real-world performance optima [95]. Previous work compared computational performance metrics but did not relate those metrics to experimental flight data. Of the trajectory planning methods reviewed by Georzen *et al.*, some methods of note are:

- **Sampling-Based Trajectory Planning: Grid-Based State Space Search** Discretises the state-space into a grid and then searches the grid for the cost-optimal path. Practical real-time application would be difficult to implement due to the number of state-space dimensions.
- **Sampling-Based Trajectory Planning: Rapidly-expanding Random Tree (RRT)** A stochastic search starting at the UAV in the body-centre coordinate frame. Computational cost is greatly increased in environments with long passageways due to the nature of the branching, and is not well suited to 3D.
- **Decoupled Trajectory Planning** Several methods are presented, all following a similar approach. First, a path is generated in 3D using a common algorithm such as a Voronoi Roadmap or A* over a grid. The path is then processed to generate a dynamically suitable trajectory for the vehicle. Several methods have been suggested for robustness, including inflating obstacles so that a larger safety margin is considered and the inclusion of a trajectory checker that restarts the process should the path be unsuitable for the vehicle.
- **Mathematical Programming** Methods such as Mixed Integer Linear Programming (MILP), Infinite Horizon Control, and Receding Horizon Control fall under this umbrella term. These methods usually constrain the problem with equations of motion and use a mathematical search approach to find the optimal route to the goal, accounting for the resolution of the planner.

Some of these methods are optimisation techniques, and others can be implemented alongside optimisation techniques. Section 2.4.2 reviews various types of optimisation algorithm, within the context of applying an RTTP to long range missions over Fuego.

2.4.2 Optimisation Methods

Optimisation methods typically consist of an algorithm to minimise some cost function. Cost functions can have single or multiple objectives and in the context of UAV motion planning they may, for example, include a function of path length and proximity to terrain. Certain limits can also be imposed which form 'hard' constraints on the problem, for example separation from an obstacle being positive (i.e. never impacting it), or the continuity of the path being maintained (to ensure it is flyable).

Methods such as Simulated Annealing and Genetic Algorithms have existed in their more basic form for over 30 years. As time has passed these methods have evolved into newer and perhaps more relevant versions of themselves. In addition to the classical methods, newer algorithms are also considered such as the Ant Colony Optimisation and the Grasshopper Algorithm Optimisation. Many modern optimisation methods are based on the behaviour of various animals and insects in the natural world, and some bear particular relevance to airborne sensing and tracking.

Trajectory optimisation has been an area of interest for a long time, with Betts' survey of numerical methods for trajectory optimisation dating back to 1998 [96]. Betts formulates the problem generically, including various methods of problem set-up, before discussing a variety of solution methods. Various shooting and collocation methods are presented, and compared briefly to Genetic Algorithms and Simulated Annealing; randomness in these methods makes them attractive to certain practical cases. While Betts considers trajectory optimisation in a more general sense, Hoy *et al.* consider algorithms for robotic navigation through cluttered environments; a survey that is more relevant to the UAV case, despite just considering 2D cases [97]. Path planning algorithms such as Rapidly Exploring Random Trees [98], Graph Search Algorithms [99, 100], and Artificial Potential Field methods [101] are mentioned, as well as Evolutionary Algorithms, Simulated Annealing, and Particle Swarm Optimisation. Sensor-based techniques, such as Boundary Following, are discussed; however, they are not suitable to the UAV problem considered here. The world-space in the UAV problem is large and the aim is not to specifically avoid obstacles but rather to achieve an efficient flight to the target location, avoiding any obstacles that may be present.

Patle *et al.* review path planning strategies for mobile robots in their 2019 paper; however, again this is restricted to 2D [102]. A significant number of recent, biologically inspired, methods are discussed below; for example, the Ant Colony Optimisation [103] and the Artificial Bee Colony Algorithm [104]. Publications limited to 2D applications are considered, despite the inherently 3D problem of route planning around Fuego. Methods that are listed as suitable for aerial navigation include Genetic Algorithms and Particle Swarm Optimisation.

2.4.2.1 Simulated Annealing

Simulated Annealing (SA) is a stochastic optimisation method that was originally presented by Davis [105], but developed and implemented for the path planning of robots by Martínez-Alfaro and Gómez-García [106], with a fuzzy logic control applied to keep the robots tracking the path. SA takes its inspiration from the process of annealing used to tailor crystal growth in metals by controlling their temperature. The principle of SA is a series of iterative searches, changing the configuration of the solution each time by (usually small) variable amounts. The magnitude of these changes can be controlled by a decaying 'temperature', and the resulting configuration is evaluated for cost. The change in the cost function relative to the prior configuration is tested

against some acceptance function and, should it pass this test, the changes are accepted. One of the key characteristics of SA is the existence of a decaying temperature in the acceptance function, so exploitation is more preferable to exploration as the number of iterations increases.

An example of a generic SA algorithm is given in Algorithm 1, where s is the state, s_0 is the initial state, k is the step number, T is the temperature, P is the acceptance probability function, and E is the cost (or 'Energy') function.

Algorithm 1: Example of a generic Simulated Annealing algorithm.

Result: Final state s
let $s = s_0$;
for $k = [1, k_{max}]$ **do**
  $T \leftarrow$ temperature $\left(\frac{k+1}{k_{max}}\right)$;
  pick a random neighbour, $s_{new} \leftarrow$ neighbour(s);
  if $P(E(s), E(s_{new}), T) \geq$ *random[0, 1]* **then**
    $s \leftarrow s_{new}$;
  end
end

Martínez-Alfaro *et al.* generate a B-spline curve for their robots to track, and use SA to optimise the path of this curve by dictating the B-spline control point positions [106]. A fuzzy logic control algorithm is then applied to the system to keep the robot on track. Their work is applicable to UAVs however certain aspects cannot be transferred; for example the robot velocity decreases in line with the forward-facing sonar readings. Although it is possible to modulate fixed-wing UAV speed to some extent, there is a non-zero minimum airspeed which means stopping in front of an obstacle is not a viable option.

2.4.2.2 Purely Probabilistic

Many of the papers reviewed here discuss algorithms that incorporate some element of randomness, so many of the algorithms can be considered probabilistic. Dogan, however, approaches UAV path planning problem from a different perspective by mapping the flight area with a probability density function (pdf) [107]. With a clear military motivation, Dogan considers obstacles in the configuration space as threat points, each with an individual probability distribution representing the likelihood of the threat disabling the UAV. Dogan claims that path and mission planning using probability theory has a number of benefits, including:

- a well-established theoretical framework;
- suiting the non-deterministic nature of the problem;
- the effect of different threat sources, or optimisation objectives, can be analysed; and
- the probability of success or failure can be quantified.

Dogan suggests that the strategy would be easy to implement in real-time, however does not give evidence of flight testing. The method involves different target points depending on the evaluation of the pdf from a given point. The process is as follows:

Algorithm 2: Pseudocode representing the purely probabilistic approach to path planning as suggested by Dogan [107]

```
find target direction;
step forward in target direction;
evaluate the pdf and compare to threshold value f_min;
while pdf < f_min do
    step forward;
    if pdf ⩾ f_min then
        take evasive action;
        enter local-minimum search behaviour to reduce pdf;
    end
end
```

A risk found with this method is entering a limit cycle when approaching the target, but at a range which does not meet the 'target-reached' criteria. The algorithm limits the maximum allowed change in heading at each decision point in a similar manner to other methods, so it is possible for the UAV to be unable to turn any closer towards the target and therefore start an infinite loop. Dogan implemented a successful prevention logic which, after a full cycle, increases the maximum turn angle and then gradually increases the threshold value f_{min} to the value of the pdf at the target.

It is argued that the method developed and simulated here can be used during multiple stages of UAV mission and path planning - both before and during flight. While suited to a hostile military environment, this method requires some knowledge about the flight area and so is not particularly applicable to the optimisation of flight plans around Fuego. The UAV behaves in either an evasion mode or a target-seeking mode, and modern optimisations develop paths which would ideally eliminate the need for the evasion mode completely. The principle behind the evasion mode is useful in the proposed environment and such reactive behaviours are worth noting, regardless of the overall method usage.

2.4.2.3 Evolutionary and Genetic Algorithms

Another form of stochastic optimisation technique is the Genetic Algorithm (GA) [105], widely used for applications such as the design of minimal phase digital filters and surface grinding processes [108]. GAs simulate the process of genetic evolution found in nature and so the terminology associated with the process is bio-inspired; the external characteristics (phenotype) are determined by the configuration of genes ('genotype') [109]. Individuals who adapt poorly to the environment tend to be eliminated from the population while fitter individuals survive and,

as generations pass, processes such as mutations and recombinations ensure a positive evolution occurs. Within the optimisation method, each individual is represented through a series of bits, integers or numbers ('chromosomes') thereby making up the information within a genotype. Classical GAs use binary coding for the representation of the genotype.

Nikolos *et al.* present a form of GA called the Evolutionary Algorithm (EA), in the context of UAV path planning [109]. They describe the EA method in detail, and use B-splines to model the UAV path-line. The operators used on the genotypes are a combination of selection, recombination, and mutation of chromosomes. The chromosomes take the form of B-spline control points, but the cost function is evaluated over the actual B-spline path. After a random initialisation, optionally bounded by limits, initial paths are generated and evaluated. The selection scheme used is an elitist truncation model where a threshold parameter T allows some of the 'children' showing the best fitness to be chosen as 'parents' for the following generation/iteration. Of the top $T\%$ solutions, the probability of being chosen as future parents is directly proportional to the solution's fitness, compared to the average fitness of the selected $T\%$. When $T = 100\%$, the selection scheme acts like a roulette scheme, where population members are chosen at random. When $T \approx 10\%$, it acts like a traditional truncation model whereby the fittest members are selected almost exclusively. Nikolos *et al.* found $40\% \leq T \leq 70\%$ as suitable and sufficient to avoid entrapment in local minima. The results presented are 3D simulations with a cost function that considers proximity to terrain and vehicle capabilities such as minimum path curvature angles and maximum flight height. The computational cost of the process is not discussed; however, the team concludes that the method is feasible under forced constraints and in a limited time period, given the low number of iterations needed to optimise the locations of the B-spline control points.

NSGA-II is a Non-dominated Sorting Genetic Algorithm that was first published by Deb *et al.* in 2002 as an alternative to traditional multi-objective non-dominated EAs [110]. A solution is said to be non-dominated if no aspects of the cost function can be further improved without worsening other parts of the cost function. Mittal and Deb use the NSGA-II EA and local searches as an offline method of finding paths for a UAV in 3D, considering two main objectives [111]. They define two problem types:

- UAV navigation with no restriction on the path; and
- UAV navigation with a requirement for the aircraft to pass through one or more specified points.

Once again, B-splines are used to generate a smooth path and the free-to-move control points are the source of change in the path. Mittal and Deb argue that combining multiple metrics into a single objective optimisation problem holds risks. Firstly, the combination of multiple objectives requires that all objectives must be of the same type (i.e. minimisation or maximisation), and secondly that weighting vectors requires normalised objectives, meaning an implicit user input of comparative importance between objectives. The objective functions considered here

are minimisation of the length of path, and the often contradictory minimisation of risk due to ground proximity. A number of constraints are placed on the problem, including not colliding with the terrain and maintaining smooth changes of direction. Mittal solves the issue of combining objectives by finding N non-dominated solutions to each objective and applying a local search procedure to reduce N to between 8–10.

Although Mittal and Deb find a way around the implicit user-weighting during the combination of objectives, the process presented serves only to move the user input to the end of the process where the paths must be qualitatively compared. The process is arguably suitable to problems with two objectives; any more than that would make the the final user-decision difficult to visualise. The computational time of the presented form was several minutes at the time of publication. Additionally, an automated decision-making process would need to be appended in order to select a single path from the generated candidates.

Wise *et al.* present a method of real-time path planning in tactical scenarios using an evolution-based algorithm [112]. Tested using a high-fidelity UAV simulator, the 2D algorithm finds a quasi-optimal path on demand within a given time window. Two test cases were primarily used; the first with a requirement to orbit a target within a 5% error in radius, and the second to loiter outside of radar detection at a given radius from a known target location. One of the novel aspects of the presented method is the incorporation of a planning horizon. This means that, after an initial offline initialisation, the aircraft has the transit time to the horizon in which to calculate the next segment of the flight path. This approach means that only small sections of the path must be generated at any given stage, making real-time computation possible. The appropriate choice of constraints and cost functions, and some knowledge of the environment, would determine the success of the method in reality. No real-world flights are presented in this work.

More recently, MahmoudZadeh *et al.* using evolutionary algorithms to solve a rendezvous problem in a cluttered environment for an autonomous underwater vehicle [113]. Xue and Sun also present an Evolutionary Algorithm in their 2018 paper, using a multi-objective cost function to optimise paths for simulated mobile robots [114]. The use of evolutionary-type methods in these recent publications, despite not relating directly to UAVs, suggests continued relevance of this established optimisation method in the field of robotics and path planning.

2.4.2.4 Particle Swarm Optimisation

The Particle Swarm Optimisation (PSO) method originates from the social behaviours of animals such as fish and birds [115]. A population of particles is called a swarm, with every particle representing a candidate solution within the search space. The swarm and space exist in d dimensions for a problem defined with d dimensions. Every particle has a position $\mathbf{x} \in \mathbb{R}^d$ and a velocity $\mathbf{v} \in \mathbb{R}^d$, both randomly initialised within the search space. The particles' positions are updated based on their velocities with each iteration. The individual velocity of each particle

is affected by the previous velocity of the particle, the best ever position held by the particle ('personal influence'), and the best ever position held by any particle ('global influence'). The equations of the method tend to follow the form shown in Equations 2.1 and 2.2, where i is the iteration number, n is the particle number, r_1 and r_2 are random values between 0 and 1, $\mathbf{g}$ is the best ever position of any particle within the swarm, $\mathbf{p}$ is the best position occupied by particle n, c_1 and c_2 are the personal and global influence parameters, respectively, and ω is the 'inertia' of the particle:

$$\mathbf{v}_{i+1}(n) = \omega \mathbf{v}_i(n) + r_1 c_1(\mathbf{p}_i(n) - \mathbf{x}_i(n)) + r_2 c_2(\mathbf{g}_i - \mathbf{x}_i(n)), \tag{2.1}$$

$$\mathbf{x}_{i+1}(n) = \mathbf{x}_i(n) + \mathbf{v}_{i+1}(n). \tag{2.2}$$

It is worth noting that the original form did not include an inertia term, however it was added in 1998 by Shi and Eberhart and has a significant and effective impact on the optimiser [116]. Salamat and Tonello use a PSO to generate trajectories for quadrotor UAVs, and compare the performance of the PSO to the standard A* algorithm, Rapidly-exploring Random Tree algorithm, and a GA for the minimum length path in a 2D problem [115]. Salamat defines the location of the start and end points, and the number of control points used. The execution time was of the order of 5–7.5 s depending on the specific problem, with swarm sizes of 50–100, 4–5 control points, and 250–400 iterations. The application of these methods to multi-rotor UAVs is distinctly different to using them for fixed-wing UAVs; however, the comparison of these results is pertinent to the selection of an initial method for trial.

The PSO was first quantitatively compared to a GA by Roberge *et al.* [108], who conduct a thorough comparison of the methods in a real-time 3D UAV path planning context. This is one of only a few papers that do not use B-splines for paths; instead, they use line segments, circular arcs, and vertical helices between points. The solutions found were feasible and near-optimal; however, the computation time proved to be an issue so the team parallelised the algorithms. The result was computation times as short as 3.63 s; however, an execution time of 10 s was justified as allowable for online application considering the typical speed of fixed-wing UAVs and the scale of the paths generated. It is important to note here that although real-time use and calculation is mentioned, it was only tested in simulation and the required computing power limited route calculations to being ground-based. With terrain models from GeoBase (Canada), extensive results are presented summarising PSO and GA performance over 60 trajectories in 40 scenarios, applied to two fictitious and six real terrain maps.

Roberge *et al.* carried out a statistical t-test on the cost distributions of the trajectories and, based on the results, the GA was shown to be slightly more preferable [108]. The fixed computation time of 10 s was applied throughout, and both methods had a population size of 256. The difference in results between this paper and the paper by Salamat *et al.* shows how, when applied to different problems, the preferable algorithm type is not easily predictable.

2.4.2.5 Grasshopper & Adaptive Grasshopper Optimisation

The Grasshopper Optimisation Algorithm (GOA) was developed by Saremi *et al.* in the context of challenging structural engineering optimisation problems [117]. The concept of the algorithm stems from the behaviour of grasshoppers as swarm creatures, with each 'grasshopper' representing a potential solution. Sharing some similarities to PSO, the search process is separated into exploration and exploitation, investigating the global and local scales respectively. Grasshoppers are initialised randomly and the location of the best grasshopper is shared globally after the cost function is evaluated for each grasshopper. An attraction or repulsion force is felt by each individual with respect to the best positioned grasshopper. The attraction and repulsion forces correlate to the exploitation and exploration of the space respectively, and the point at which the two forces are equal is labelled the comfort zone. The radius of the comfort zone can be controlled mathematically, thereby determining the weight applied to exploration and exploitation. This weighting can be dynamic and usually favours exploration initially before narrowing down to local exploitation as the number of iterations increases. It is conceivable that the random initialisation causes the grasshoppers to be near a local minimum and far from the global minimum, thereby causing the algorithm to fail. Saremi *et al.* compare the GOA to various methods including PSO, GA, Differential Evolution, and Gradient Search Algorithms, and it is shown to perform better on the benchmarks and tests used.

The GOA was applied to solar UAVs in simulation by Wu *et al.*, finding trajectories through a hypothetical 3D urban environment while maximising flight through non-shadowed regions [118]. The team note deficiencies in the GOA, such as a sensitivity to initial positions and a lack of feedback on the weighting of exploration and exploitation in consecutive iterations. Consequently, the Adaptive Grasshopper Optimisation Algorithm (AGOA) was developed. This incorporated an element of natural selection, democratic decision making, and a dynamic feedback mechanism based on the 1/5 Principle [119]. The natural selection method used picks x grasshoppers of the group G with a probability P, and eliminates them before reinitialising them randomly [118]. The probability P is related to grasshopper fitness, with larger cost functions increasing the chance of selection. The democratic decision making method used means the direction of the swarm is decided by the location of the three best grasshoppers, rather than the single best, thereby reducing the likelihood of falling into local minima. The dynamic feedback strategy stems from evidence that the best evolutionary rate of a biological population is approximately 20%, with rates much above or below this having detrimental effects on the population growth. The rate in this case is defined as the ratio of grasshoppers whose fitness has improved from the last iteration, and the exploration/exploitation weighting is adjusted accordingly to keep the rate between 15% and 30%, while maintaining a shift of weight towards exploitation as the number of iterations increases.

2.4.2.6 Ant Colony Methods

Ants are able to find the shortest route between points by utilising pheromone information in a positive feedback loop. Individual ants reach a two-way decision point and the population will be divided roughly in half with no recent experience of the route. Travelling at a near constant speed, ants on the shorter path will reach the opposite decision point sooner than the longer path. The same happens on the ants' return to the nest. Ants leave a pheromone trail, which naturally builds up quicker on the shortest path. The choice of arriving ants is then affected by the existing pheromones, choosing the shorter path over the longer path. This feedback mechanism accumulates and the effect on the colony is rapid.

Dorigo *et al.* developed the Ant Colony System (ACS) in 1997 and applied it to the classic Travelling Salesman Problem (TSP) [120]. Each 'ant' in the system is initialised at a random node, or 'city' as they are known in the TSP. The pheromone here is a desirability measure, analogous to a cost function. Each ant finds a solution to the TSP, choosing shorter paths between cities and taking into account the pheromone information left behind from previous iterations. Once all ants have completed their tours, a global update of the pheromones takes place where a portion of the pheromones evaporates on all paths. The ants then add pheromones to the complete path of its tour from the previous iteration. The process is iterated, resulting in paths which are not frequently updated (i.e. travelled) becoming less desirable. Dorigo *et al.* compared the ACS to other heuristic methods, such as SA and GA, and for a symmetric TSP problem the ACS proved most desirable on 50-, 75-, and 100-city problems.

The ACS was further developed into the Ant Colony Algorithm (ACA) by Ma *et al.* for the purpose of global optimal trajectory planning for UAVs [121]. Ma highlights issues with the original ACS method, including convergence on local minima and comparatively long computation times, and suggests the following improvements:

- The addition of an elitist strategy; reserving the optimum tour at the end of each iteration.
- Limit the strength of the pheromone trail laid down between both lower and upper bounds, to help prevent stagnation.
- An adaptive pheromone decay parameter, weighting exploration more heavily towards the beginning of the process and decaying towards exploitation as the iteration number increases.

Ma *et al.* compared their ACA to the ACS and concluded that their ACA is better at avoiding local minima, although there are a lack of numerical tests presented in the paper. The simulations provided are limited to 2D.

Zhang *et al.* also use an Ant Colony method, applying it to a UAV path planning problem [103]. Obstacles are modelled as terminal and non-terminal threat factors, and the yaw angle change of the path is limited as part of the optimisation. Renamed the Ant Colony Optimisation

(ACO), this method is also only applied in 2D, however the computational costs involved for the problems provided are low. Zaza and Richards [122] describe an Intelligent Foresight strategy for the vehicle routing problem, using the optimal solution from an ACO as a warm-start for their method. This method focuses on the introduction of additional tasks along a TSP-type problem, with a previously optimal solution in place. The method is compared to the Gurobi solver over 15 scenarios and the results suggest that it provides near-optimal solutions at reasonable computational costs. There are no known papers which used an Ant Colony based method for UAV path planning in 3D.

2.4.2.7 Differential Evolution

Differential Evolution (DE) holds similarities to EA and GA methods, including use of selection, crossover, and mutation processes, however the details of these functions are more variable than the precise operations inherent to classical GAs. It was first presented by Storn and Price as a preferable alternative to adaptive SA for the minimisation of continuous and potentially nonlinear space functions [123]. Similar to EAs, DE is easily parallelised so significant speed-ups are possible. Du *et al.* continue developing the DE method by integrating an event triggered impulsive control mechanism [124]. The aim of this is to instantly adjust the states of some nodes at certain instants and is applied here to further enhance the scope of DE systems. Four components were developed:

- **Stabilising Impulses:** forcing individuals towards promising areas, aiding convergence
- **Destabilising Impulses**: moving individuals with poor fitness, maintaining diversity
- **Ranking Assignment:** selecting individuals for injection with impulsive controller inputs
- **Adaptive Mechanism:** counting the individuals receiving impulsive controller inputs

When applied to a benchmark problem, Du *et al.* show their method to be simple and effective. Kok *et al.* apply DE to UAV path planning problems, specifically the optimisation of the control parameters for UAV path planning problems [125]. It has been claimed that constant path planning control parameters do not perform as well as dynamic parameters, so Kok *et al.* study the effect of DE control parameters in detail, applying a DE algorithm to find optimal parameter values such as differential weight and population size.

2.4.2.8 Mixed-Integer Linear Programming

The Mixed-Integer Linear Programming (MILP) method is described in generic detail by Floudas [126], and is applied to UAV path planning problems in 2D by Richards [127] and Bellingham [128]. The basic method as presented by Floudas consists of linearising the optimisation problem such that it follows the form of Equation 2.3 and is constrained by expressions like Equation

2.4, where $\mathbf{x}$ is a vector of continuous variables, $\mathbf{y}$ is a vector of binary variables, c and d are parameter vectors, A and B are constraint matrices, and b is a vector of inequalities:

$$min(c\mathbf{x} + d\mathbf{y}) \tag{2.3}$$

$$\begin{aligned} A\mathbf{x} + B\mathbf{y} &\leq b \\ \mathbf{x} &\geq 0 \\ \mathbf{y} &\in 0,1 \end{aligned} \tag{2.4}$$

Richards, Bellingham, and How linearise the UAV dynamics necessary for a 2D path planning problem in various forms, qualifying the speed as constant and placing a limit on the turn rate or forces of the aircraft such that any course change sees the maximum possible turn rate [127, 128]. Richards focuses on collision avoidance using MILP, including simulations run with multiple aircraft each travelling across the same space with unique start and end points. The results for a single aircraft include optimisation solution times of 49 s and 65 s for the cost function with a force penalty. Optimisations that were run without the force penalty just searched for a minimum time solution and took 142 s and 227 s respectively when applied to the same problems. Across 783 feasible problems, the average solution time was 11 s with a maximum of 68 s.

Bellingham uses a similar approach, however applies a receding horizon to the method in order to reduce computational time [128]. Two dynamics models are implemented, with differing levels of resolution. The result of this is a long horizon low resolution controller that decides which gaps between obstacles to pass through, and a short horizon high resolution controller that generates dynamically feasible time-optimal routes around nearby obstacles. Entrapment is avoided by the aircraft path connecting to a cost map of the environment beyond the planning horizon, and constraints then being placed on the points where the planned path crosses the horizon. The presented simulation took a cumulative time of 313.2 s with an average of 2.97 s spent designing each trajectory segment. There must be some knowledge about the environment in this case, a potentially limiting factor to implementation in the real world with UAVs on long range BVLOS missions.

Richards continues his work in MILP trajectory optimisation by ensuring that avoidance constraints are applied at all times between discrete sampling points [129]. Previously, inter-sample resolution issues were avoided by adding a margin to the size of the obstacles such that any incursion into the obstacle space in the plan would still result in a suitable separation from the actual obstacle [130]. Richards observes that inter-sample avoidance can be achieved by applying the constraints specific to time step t to time step $t-1$, and includes methods of taking vehicle dynamics into consideration [129]. The ability to maintain inter-sample obstacle avoidance permits a reduction in time-steps and computational cost.

Borelli *et al.* compare the use of MILP and Non-Linear Program (NLP) methods for 3D conflict resolution for multiple UAVs [131]. The MILP approach is found to always be faster, particularly

in simulations with large numbers of UAVs and obstacles, however the flyability of the result was not guaranteed due to the use of linear UAV models. NLP methods are similar to MILP but, where MILP discretises time uniformly over the upcoming horizon, NLP discretises time as a function of the element spacing in that area. A main conclusion of Borelli's paper is to find initial results using MILP before introducing NLP solvers which develop the solutions with the inclusion of aircraft dynamics and constraints.

Raap *et al.* explain the application of MILP to a K-step lookahead UAV based search algorithm [132]. Using a hexagonal-grid probability-based search, they take into account the turn rate limitations of a fixed-wing UAV with promising results. The computational cost increases dramatically with the number of decision variables; commercial MILP solvers become insufficient for effective implementation of the method. The work presented only considers a 2D flight envelope.

The MILP methods and applications considered here are purely 2D and an extension to 3D is, in theory, relatively straightforward. The application of MILP methods to UAV planning depends on the linearisation of aircraft dynamics, and a successful real-world outcome depends on the UAV performing as modelled. This is an issue when it comes to flyability, both because of the occasional need to include non-linear dynamics and the potential lack of continuity in the path.

2.4.2.9 Branch and Bound

Eele presents a method of finding optimal trajectories for a 2D problem with fixed obstacles; the Branch and Bound method involves branching, that is the problem of avoiding obstacles; and bounding, ensuring that the resulting path fits certain length criteria [133]. Breaking the problem into sub-problems, the global optimality of the solution is guaranteed, provided that the optimal solution to each sub-problem is found. A multitude of branching strategies are available, 9 of which are tested in Eele's paper. The overall method is compared to RRT and MILP solutions of the same problems; the Branch and Bound cases presented had a higher computational overhead however found solutions to 100% of the cases as opposed to 64.3% that the MILP solver was able to solve. RRTs had the quickest computation times, however found sub-optimal paths. Allowing the RRT to search for longer would improve the optimality of the results, however this increased the overall solve time. The developed Branch and Bound method is claimed to combine the best features of both; the fast solution times of the RRT and the high performance of the MILP. There is no mention of practical implementation in this paper, and the problems are constrained to 2D.

Dai *et al.* consider optimal path planning for solar UAVs, representing aircraft dynamics as quadratic functions in terms of quaternions and solving using a Branch and Bound method [134]. Rather than consider obstacle avoidance or path length, Dai *et al.* optimise for maximum solar energy harvesting while travelling from point A to point B in a cluttered city environment. Two simulations are presented, one for level flight and one for a 3D case, demonstrating the hypothetical viability of the approach. The Branch and Bound method is suited to solving these quadratically-formed equations in a known and constrained environment, however little work

has been done applying it to large problems in unfamiliar world-spaces because it does not seem to scale well.

2.4.2.10 Model Predictive Control

Model Predictive Control (MPC), also referred to as Receding Horizon Control (RHC), is an iterative process which in the context of UAV path planning attempts to optimise a trajectory over a limited number of iterations into the future [135]. Traditional 'centralised' MPC schemes solve a single optimisation problem at each time step for a number of steps into the future. This method works well for small problems, however does not scale well with system size and results in high computational costs. Richards and How present a Decentralised MPC (DMPC) method, where each sub-system solves a problem only for the relevant sub-problem [135]. The sub-systems solve their problems sequentially, so any pertinent results from earlier systems can be used in later systems. The objective function for each sub-system is assumed to be decoupled and independent of other sub-systems. Richards and How later published a paper with a developed, more robust version of the DMPC method, including problems with coupled constraints [136]. The results of each sub-system are proven to be feasible with no iterations between sub-systems required at that time step, and overall the method is shown to be faster and more scalable than the equivalent centralised MPC controller. Results and examples are presented in 1D, demonstrating an improvement to similar past methods however a significant amount of work would be required to develop a 3D version for real-world implementation.

Bo *et al.* implement DMPC along with a Sparse A* algorithm (SAS) in a hierarchical system to plan the 2D path of multiple UAVs [137]. The SAS algorithm is described as a version of a classic heuristic algorithm, selecting the next node according to the lowest evaluation of some cost function with the form:

$$f(\mathbf{x}) = a.g(\mathbf{x}) + b.h(\mathbf{x}) \tag{2.5}$$

where $\mathbf{x}$ is the position vector, $g(\mathbf{x})$ is the cost from start position to current position, $h(\mathbf{x})$ is the expected cost from current position to end position, and a, b are scalar weightings. The step size used varies depending on the level of danger in the area, thereby decreasing computational cost significantly but increasing the risk of accidentally missing threats.

Bo *et al.* did not conduct any real-world testing, however the method presented would require real-time computing from a ground station unit in order to run the computation in a reasonable time, and coordinate the flights of multiple UAVs [137]. The team simulated the system using MATLAB optimisation toolbox, using both linear and non-linear methods depending on the status of the collision avoidance manager.

2.4.3 Real-Time Navigation

Modern day UAV autopilot control systems conduct a form of real-time navigation, however the mission route is often predetermined. This implies an element of certainty about the route, as opposed to the uncertainty around the routes developed by an RTTP where the operator expects to take-off without knowledge of the full mission route. Some groups have researched real-time path planning for UAVs, however it seems common to develop the method and test in simulation without experimental testing, particularly for fixed-wing configurations.

Conventional autopilots involve some combination of inertial sensors, GPS, and a Kalman filter to determine the state of the aircraft [138]. Vision-based alternatives have been researched, with a series of experimental results presented by Conte and Doherty [139]. The UAV in this case was an IC helicopter with a maximum take-off mass of 95 kg, disqualifying this as sUAS or MAV research. The computation required for the real-time navigation aspect was conducted onboard, however the path was planned in advance. Flight along that path was controlled using the developed real-time vision-based system, and compared to a traditional inertial/GPS system. Tests included simulating GPS failure; this showed that the vision-based system was an effective alternative to conventional flight controllers, allowing the flight to continue as planned along the pre-programmed path.

Ai *et al.* generate a mixed reality environment in which the UAV and generated real-time data were mapped together [140]. This project did not consider automated navigation, however data was collected and processed in real-time in order to map the environment for the operator. Qualitative tests were carried out using an indoor multi-rotor.

Developing something similar to traditional autopilots, Mohanta *et al.* use a microcontroller to control a quadcopter [141]. Rather than just follow a prescribed path, their system used infrared sensors to detect obstacles in real-time and alter the route accordingly. Initial tests and development were conducted in simulation environments, and then verified using real-world flight tests.

Lee *et al.* present a method detecting and navigating to objects using a quadcopter in an indoor environment [142]. High-detail object recognition took place off-board, making use of a cloud based neural network, however low level object detection and short term navigation took place onboard. The tests presented include a target search scenario, where the aircraft searched for and navigated to an object. This constitutes a similar type of real-time navigation to the intended use of an RTTP, albeit in a small indoor environment and in 2D.

Letheren and Montes worked together over several publications to present a real-time moth-inspired navigation algorithm for finding the source of forest fire plumes, using multi-rotor UAVs [83, 84]. In order to test the performance of the algorithms used, Letheren *et al.* developed a novel way of modelling the plume for UAV simulations by adding a variable wind speed to the Holzbecher model [143]. A stochastic element was introduced to the modelling of the tracker, tested on a randomly generated plume. Letheren *et al.* developed the system using a

computational and simulation based approach, then troubleshooted the experimental set-up by running Hardware In The Loop (HITL) simulations. They then used a 3DR IRIS quadcopter UAV to track a plume in 2D in real-time. The payload of the aircraft is sufficient for the sensor and an additional telemetry unit, with the computation of the algorithm taking place on the ground station. Although this is sufficient for the example presented in this paper, the telemetry links to small UAVs are often not sufficiently reliable when operating at long range in large world environments, motivating onboard computation.

2.4.4 RTTP Summary

Of the methods reviewed in this section for the purposes of real-time trajectory planning, just three were considered for further exploration. Based in part on the thorough research presented by Roberge *et al.*, the methods selected for initial further consideration were the SA, GA, and PSO [108]. After basic initial testing, the GA was selected for full-scale implementation. Further justification, initial exploratory development, and real-world use is described in full in Chapter 6.

2.5 Coordinated Plume Interception

The PSD of ash within a volcanic plume varies with time post-eruption, as heavier ash particles fall out. Given non-zero wind conditions, the PSD of ash within a plume also varies with distance from the crater. The rate of change of ash PSD in the plume is an area of interest for volcanologists, and can be used to verify and refine ash propagation models for aircraft management tools. Collecting the necessary data requires sampling twice from the same plume, with distances between samples ideally ranging from 5 km to 20 km. While it would be possible to do this with two separate aircraft and flights, the simplest method involves a single with the ability to collect two distinct samples.

Sampling the plume a second time, some distance downwind, requires a prediction of the plume position as a function of time. The aircraft must then be navigated such that the UAV and plume intercept at the given location. The automation of this Coordinated Plume Interception (CPI) would lead to an experimentally consistent approach to ash collection, and reduce pilot workload. The real-time navigation aspect of the RTTP can be made use of here for aircraft control, and the automated plume sensing work introduced in Section 2.2.4.1 could also be incorporated to further automate the system.

Predicting the future position of the plume depends heavily on the wind conditions. Precise atmospheric measurements can be taken using instruments such as ultrasonic anemometers mounted on multi-rotors or fixed-wing UAVs, however the prop-wash must be corrected for in order to generate reliable data [144]. Hollenbeck *et al.* conduct simultaneous flight tests using a multi-rotor and a fixed-wing in order to compare estimation methods, and conclude that results from the quadcopter are more reliable. The data for the fixed-wing tests were found to be noisy

and significantly influenced by crosswind aerodynamics, however this was likely due to the location of the sensor under one of the wings.

Rodriguez *et al.* presented a system for generating a wind map in real-time, using a fixed-wing UAS, in their 2016 publication [145]. The motivation behind this work is rooted in energy efficient trajectory planning, which is applicable to the cases in this project work. Instantaneous wind data was collected using two methods; the first made use of the onboard autopilot wind estimator, and the second was calculated on the ground station. These data were then fitted to a Weibull model [146] and extrapolated onto a grid space representing the real world. This method meant that wind data could be collected at inconsistent altitudes, and gave promising results, however is susceptible to the influence of gusts and turbulence.

Once a position and time has been defined for re-sampling the plume, the CPI method becomes similar to a rendezvous problem, such as that presented by Wilson *et al.* [147]. This paper initially develops the problem in simulation before validating the results by flight testing both multi-rotor and fixed-wing aircraft in an indoor flight arena. The first aircraft was set to fly a predetermined flight path, and the scheme commanded the second platform in order to coordinate rendezvous. While relevant to the volcanic CPI problem, this approach relied on a motion capture system to determine aircraft positioning. This means that factors such as variation in wind speed and direction could not be taken into account. McClain *et al.* also present a method of planning UAV trajectories such that multiple aircraft will arrive at their respective destinations at the same time [148]. A Voronoi roadmap is used to avoid threat areas (in 2D), and then aircraft speeds are varied for flyability and to keep arrival times consistent.

Lee and Shim also present a path planning approach for fixed-wing UAV rendezvous problems, however they consider an individual aircraft with arrival time and direction constraints [149]. While they use an RRT for the path planning element of the solution, once again the time of arrival is adjusted by modifying the inertial speed of the aircraft. The extreme conditions of the plume interception problem around Fuego mean that efficiency must be a key aspect of the solution, requiring the airspeed to equal best-endurance velocity. The CPI problem differs from the rendezvous problem, as presented by McClain, because only one of the agents can be manipulated; the velocity of the plume cannot be altered. While it could be considered an optimisation problem, it is possible to find valid trajectories using a series of geometric calculations. Further development of the problem, and various solutions, are considered in Chapter 7.

CHAPTER 3

HARDWARE & OPERATING ENVIRONMENT

Hardware selection must always be driven by mission requirements within the context of the operating environment. This chapter first considers the conditions around Volcán de Fuego before addressing the mission requirements. With these in mind, the UAV airframe and systems are described alongside the ground control station and onboard computing systems.

While hardware selections naturally evolved over the course of the project, this chapter gives a summary of the key elements, focusing on those which satisfied the mission requirements. Chapter 4 describes the underlying processes behind the selection of these systems, and describes the progression of the system across five field trips (labelled Trip A – Trip E).

3.1 Operating Environment

As introduced in Section 2.2, Fuego is an active volcano in Guatemala with a summit altitude of approximately 3800 m AMSL. Missions around Fuego which aim to collect data from a static location, or collect high quality imagery from slow-moving vehicles, are best suited to multi-rotor UAVs. The greater range and endurance offered by fixed-wing UAVs makes them better suited to collecting data over longer time periods and large distances. Fixed-wing platforms also usually climb more efficiently and so are more appropriate for mission profiles with significant altitude gains. Figure 3.1 shows a to-scale map, highlighting the location of UAV operating sites and Guatemala City International Airport (GUA) in relation to Volcán de Fuego. Multi-rotors have been flown from Volcán de Acatenango for high resolution image collection, and fixed-wing UAVs have been flown primarily from the INSIVUMEH Observatory, but also from La Reunion Golf Course.

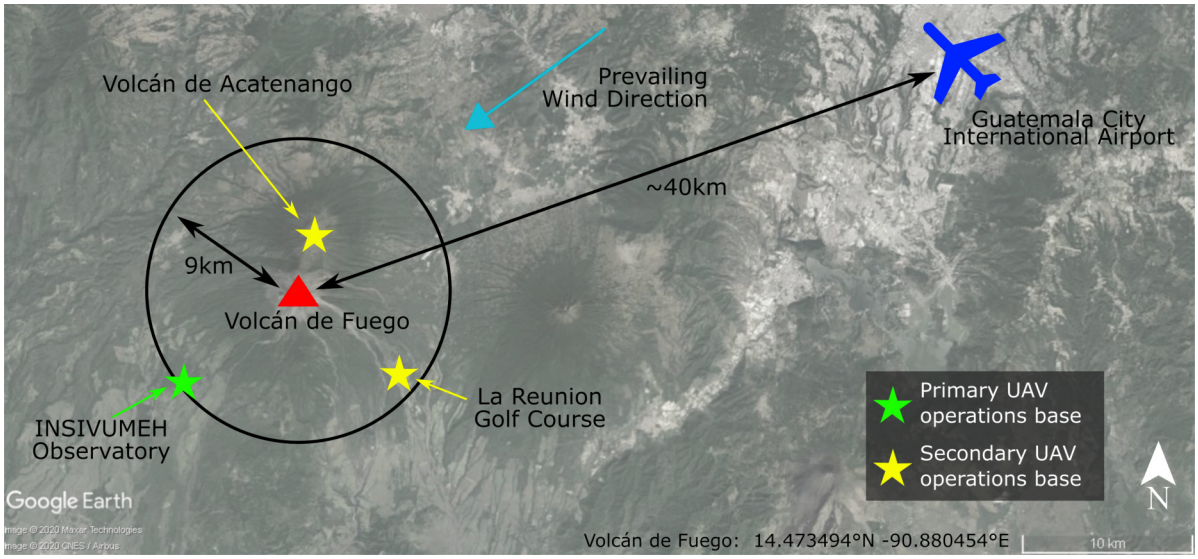

Figure 3.1: A scaled map showing the proximity of Volcán de Fuego (Red, Triangle) to some pertinent locations. The green star is the location of the INSIVUMEH Observatory, the primary operations point. Yellow stars indicate other locations from which UAVs have been flown to monitor Fuego. Note the prevailing wind direction means Fuego's plumes typically travel away from Guatemala City's international airport, however they have been known to affect both the airport and nearby air corridors. Base map source: Google Earth [10].

Given the prevailing wind direction, ash fall from Fuego does not usually impact GUA directly, however the flight corridors in the immediate vicinity are regularly affected. Although uncommon, a combination of altered wind direction and periods of high volcanic activity have lead to temporary closures of GUA [150].

The closest location to the summit of Fuego from which UAV operations have been based is Volcán de Acatenango. At an altitude of 3600 m AMSL and a distance of approximately 2 km to the crater, this is an ideal position from which to fly multi-rotors over the summit area for the collection of high spatial density datasets. Access to the site involves driving up a track that is impassable for much of the year, followed by a 30 minute walk. This means that any equipment required at the operating site must be carried on foot, as well as provisions such as food and water. While not prohibitive, the logistics of flying UAVs from Acatenango limits operations there to day trips with lightweight multi-rotor systems.

Plume sampling missions, which involve large temporal and spatial variation, are best suited to fixed-wing UAVs. The increased range and climb performance of these vehicles means that Ground Control Station (GCS) locations are not limited to Acatenango, and other more practical sites could be considered instead.

Maintaining good separation between UAVs and general aviation aircraft, and between UAVs and third parties, is critical to building a good safety case around BVLOS flights. This means that the operating site and GCS location must be close enough laterally to Fuego such that a

general aviation exclusion zone includes much of the operating area and ideally the airspace above the GCS too. Consequently, the operations site is likely to be at a relatively high altitude, reducing the required mission altitude gain but challenging the system during the take-off and recovery phases.

Should unexpected issues with the UAS or environmental conditions arise, the mission profile would ideally allow for a 'Return To Launch' (RTL) recovery. Locating the operations site close to the climb path, and Fuego, means the GCS should always be within gliding range of the aircraft; a useful safety feature. Other criteria include suitable areas available for take-off and recovery, the permission and cooperation of the relevant landowners, and the availability of facilities for the field team over an extended period.

Two of the fixed-wing operations sites initially considered for this work were the La Reunion Golf Course and the INSIVUMEH Observatory (Figure 3.1). While La Reunion initially looked promising, access and logistics meant that it was impractical to operate there while golf course clients were on-site.

Existing links with INSIVUMEH meant they offered the use of their Observatory in Panimache; Table 3.1 summarises the ways in which it was well suited to the requirements of the project. At 1137 m AMSL, and 9 km South-West of Fuego's summit, it offered an ideal location downwind of the crater from which to operate and sample the plume. That said, the altitude, surrounding environment, and tropical climate lead to a number of issues for aircraft take-off and recovery.

Table 3.1: Table summarising the key ways in which the INSIVUMEH Observatory satisfies the requirements of a Fuego-monitoring UAS operations site.

Operations Site Criteria	**INSIVUMEH Observatory**
Proximity to Fuego	9 km South-West (i.e. downwind) of Fuego
Limited overflight of third parties required	No settlements 'uphill'
Mission profile allows 'Return To Launch'	Yes; 9 km range and 3 km climb
Facilities available for extended field trip	Yes; living quarters, shops < 1 hr drive away
Landowner permission and cooperation	Positive relationship with INSIVUMEH
Take-off and recovery site options nearby	Several options available, but all limited

The latitude and altitude of the observatory results in challenging 'hot & high' conditions, impacting the lift generated by a wing. Pressure decreases with altitude and air temperature, which in turn reduces the air density. The standard form of the lift equation is:

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

where ρ is air density, V is airspeed (the speed of air relative to the vehicle), S is wing area, and C_L is the non-dimensional lift-coefficient. To maintain a given amount of lift, a reduction in air density must be compensated for by another term. The most effective and predictable

term to change is the velocity, due to the exponent, requiring more thrust in order to achieve the equivalent flight condition at sea level.

Table 3.2: Air and Lift data values at the Observatory GCS and approximately 200 m above the summit of Fuego, with ISA Standard Day values for comparison.

Parameter	**ISA Standard Day**	**Observatory**	**Summit Region**
Altitude (m AMSL)	0	1137	4000
Pressure (kPa)	101.3	89.3	63.4
Temperature ($^{\circ}C$)	15	34	10
Density (kg/m^3)	1.225	1.013	0.780
Equivalent Lift (%)	100	83	64

Table 3.2 compares the International Standard Atmosphere (ISA) standard day air-data values [151] to those found at the altitude of the INSIVUMEH Observatory, and approximately 200 m above Fuego's summit (4000 m AMSL). Note, the conditions at the University of Bristol Flight Lab test site are approximately representative of the ISA Standard Day values. All other parameters remaining constant, the lift generated at maximum altitude is 64% of standard day lift, and 77% of 'take-off lift' at the GCS. This creates challenges around the slowest points of flight, namely take-off and landing, and results in a significant increase in both stall and cruise speeds.

There are a number of challenges involved with operating from the INSIVUMEH Observatory, mostly related to the remote nature of the site. One of the main issues found in early work was take-off and recovery in the hot & high conditions. Much of the land around Fuego is taken up by plantations, with remaining space unfarmable due to dense brush. The effects of these conditions on the system and operating methodology are discussed further in Chapter 4.

3.2 Regulatory Compliance

Operating BVLOS in any country requires liaison with the National Aviation Authority (NAA) in order to ensure compliance with local regulations. All flights conducted in the United Kingdom by the University of Bristol Flight Lab comply with the Civil Aviation Authority (CAA) by following the appropriate rules and regulations regarding visual flight rules, altitude limits, and separation limits from third parties [44].

The field team worked closely with the Guatemalan NAA (DGAC) to develop a safety case for BVLOS UAV missions around Fuego. This involved the issuance of NOTAMs (NOtice To AirMen) for the times and areas of operation, as well as maintaining good communication with DGAC throughout each trip. The proposed airspace was initially conservative, but as mission capability grew the airspace developed while maintaining the same high levels of safety. An example of the flight zones submitted for the original safety case are shown in Figure 3.2. Note the exclusion

of low-level flight beyond the ingress/egress corridor, which joins the GCS with the area around Fuego's summit. The neighbouring peak of Acatenango, approximately 2 km North of Fuego, is also in an excluded zone. These restrictions were placed to include a buffer of space between the aircraft and any third parties on the lower flanks of the volcano or on Acatenango, and to help keep blockages of radio contact with the aircraft to a minimum.

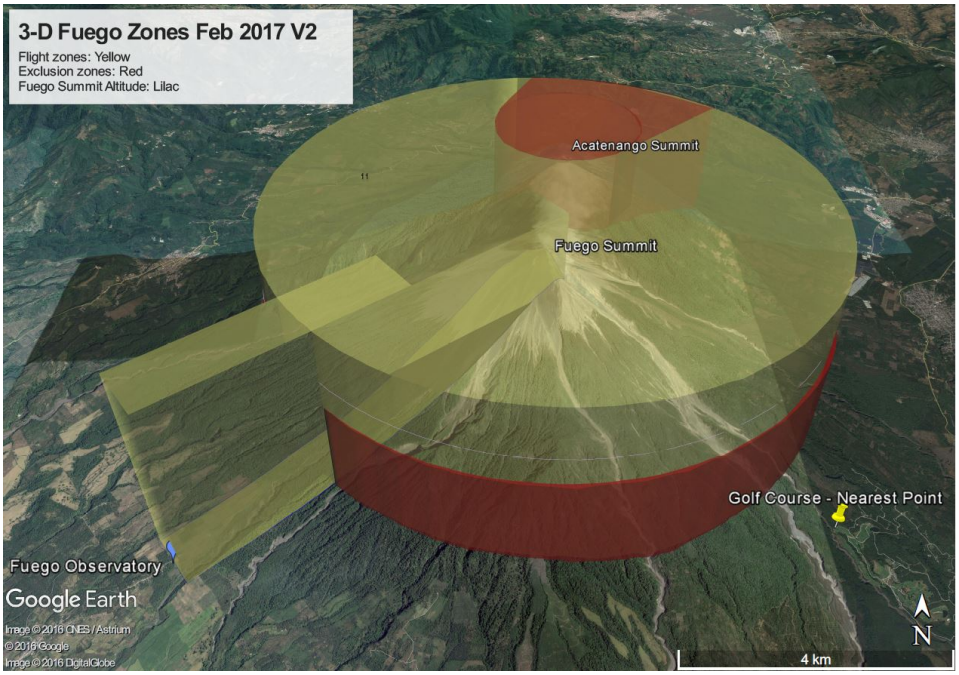

Figure 3.2: Example flight zones from part of the safety case originally approved by DGAC before the inaugural trip [10]. These zones changed on a trip-by-trip basis as the team developed an effective and reliable method of operating.

Part of the safety case submitted to and approved by DGAC involved presenting protocols for maintaining safe operations. These focused on the use of failsafes, for situations such as loss of signal or low battery, and responsible incrementation of flight time and range to work towards useful mission profiles and performance. Some of these protocols are summarised below:

- The Safety Pilot will retain overall control of each flight and will be able to take control of the vehicle at any point, as well as being able to independently trigger a 'Return To Launch' (RTL). RTL involves the aircraft automatically returning to the launch point and loitering at a specified altitude until landing is initiated.

- If the main controller safety link is lost for more than 20 seconds, then an onboard timeout will trigger the RTL flight mode.

- A low battery alarm triggers an RTL event, followed by a landing procedure.

- The maximum flight time for each vehicle, which is related to battery performance, flight paths, and environmental factors, will be assessed and updated after each flight.
- Flights will be planned and monitored such that they end with at least 20% battery capacity remaining.
- Live telemetry will be monitored throughout each flight, triggering an RTL should any values appear outside their expected range. These parameters will be analysed after each flight in order to safely maximise future operational time.
- Flights will progress incrementally, both in altitude and distance from the ground station.

These were progressed further as the team developed an effective *modus operandi* for operating safely and successfully in the area. The inherent danger to manned aviation in the airspace around Fuego, volcanic ballistics, is a natural benefit to building this safety case. Despite this, and the measures outlined above, light aircraft have occasionally been known to approach and sometimes encroach on the UAV operations area during active NOTAMs, typically without live transponders. A diagram showing the team's procedure in these rare emergency situations is outlined in Figure 3.3.

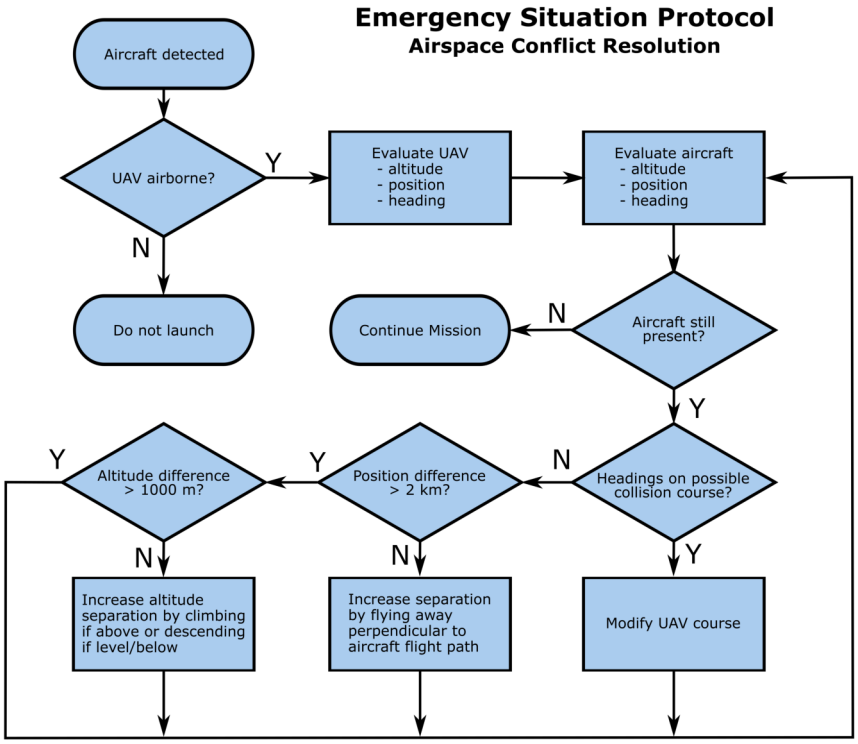

Figure 3.3: Flow diagram demonstrating the process used for dealing with manned aircraft during UAV operations around Fuego. This was developed as part of the emergency situation protocols in the safety case and was rarely required.

Missions around Fuego inherently focus on spending time at high altitude, usually in the region around the summit. This usually makes maintaining separation from other aircraft straightforward, with manned aviation naturally avoiding the dangerous ashy plume and summit regions. Most aircraft detected complied with the NOTAMs, but were monitored throughout their presence regardless. However, on one particular occasion an aircraft entered the defined UAV airspace during a mission, just as the UAV started its automatic descent from the summit region. The manned aircraft passed over the field team at the observatory with an estimated maximum altitude of 2000 m AMSL. Upon detection of the manned aircraft, the UAV was kept in a holding pattern, maintaining a large lateral and vertical separation until the manned aircraft had passed and the descent could be resumed.

3.3 Aircraft & Systems

This section introduces the aircraft and systems used through the project. Justification for these choices are made in Chapter 4, together with the methods developed for use. The first trip to Fuego, in February 2017, used a Zephyr II (RiteWing, USA) (Figure 3.4) as an initial proof of concept aircraft. Later trips used the X8 flying wing (Skywalker, China) (Figure 3.5), which has the ability to carry a variety of small scientific payloads.

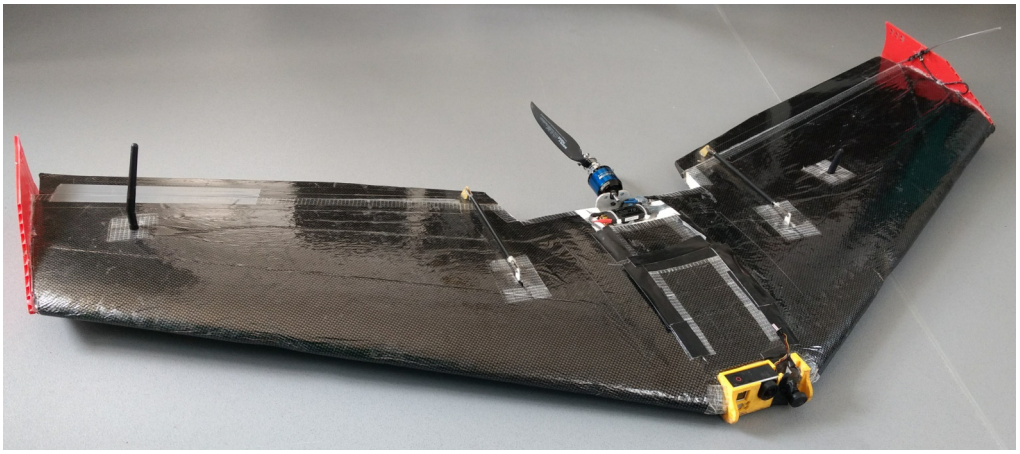

Figure 3.4: The RiteWing Zephyr II, used to scout the ability of drones to be used at Fuego. With the exception of a GoPro Hero 3 camera, no additional sensors were mounted.

Details of the vehicles flown in Guatemala are given in Table 3.3, including the avionics. Both aircraft were available as hobbyist kits and assembled in a bespoke configuration for this project, with carefully chosen long-range avionics to gain the desired performance. The use of commercially available airframes means the project could focus sooner on the challenges involved with operating in hazardous environments, maximising flight efficiency (as introduced in Section 1.3.3) by exploring different operational methodologies rather than aircraft designs. This in turn means the methods and results of the research conducted can be considered relevant for similar

mission types, regardless of aircraft specification. The small size of the vertical wing tips, and lack of conventional tailplane, means that flying wings must typically cruise at a relatively high airspeed to avoid dutch roll instabilities and tip stalls. The Skywalker was chosen because the airframe offers a large payload bay and their high cruise speed and efficiency suited the planned missions.

Table 3.3: Specifications of the RiteWing Zephyr II and Skywalker X8, based on real-world mission performance. Note, MTOM is the abbreviation for Maximum Take-Off Mass, LiPo is for Lithium Polymer, LiIon is for Lithium Ion, and ESC is for Electronic Speed Controller.

Parameter	**RiteWing Zephyr II**	**Skywalker X8**
Max Flight Time	30 mins	42 mins (LiPo) / 75 mins (LiIon)
MTOM	2.48 kg	4.2 kg
Wingspan	1.4 m	2.1 m
LiPo Battery	2 × 4000mAh, 14.8V	2 × 8000mAh, 14.8V
LiIon Battery	n/a	2 × 13600mAh, 14.8V
Motor	OS Motor OMA-3820-1200-W	Trips B-C: Overlander 5045/10 720kV Trips C-E: AXi 4120/14 660kV
ESC	Graupner 70 7237.D35	Trips B-C: Overlander XP2 80A Trips C-E: Jeti Adv 77 Pro Opto
Propeller	Aeronaut 10 × 6"	Aeronaut 14 × 6" and 14 × 8"
AutoPilot	Holybro PixFalcon	Unmanned Tech Pixhawk
AutoPilot Software	ArduPlane 3.7.1	ArduPlane 3.7.1
Pilot control link	DragonLink V3 Adv, 433MHz	DragonLink V3 Adv, 433MHz)
Telemetry link	RFD 868+, 868MHz	RFD 868+, 868MHz
Video link	ImmersionRC 700mW, 2.4GHz	ImmersionRC 700mW, 2.4GHz)
Video Camera	RunCam Eagle GoPro Hero 3	RunCam Eagle GoPro Hero 3 – 7

3.4 Ground Control Station

The Ground Control Station (GCS) comprised of a series of laptop computers and an automated antenna tracker (Figure 3.6, Left), powered by a 12V lead acid battery. The tracker was equipped with a Pixhawk hardware board, a two-axis pan and tilt mechanism, a 5.8 GHz video repeater, and a network switch for additional flight monitoring units. The video was repeated locally so that FPV goggles and wireless screens could be used.

The antenna tracker automatically maintained alignment between the aircraft and its high-gain directional antennas, providing two of the three system links to the aircraft: telemetry and FPV video. Computing onboard the tracker took place on a Raspberry Pi 3B+, running MAVProxy so that multiple people could view flight data at once. Aircraft telemetry and video was forwarded to computer terminals on an Ethernet network, with GCS computers running modified open

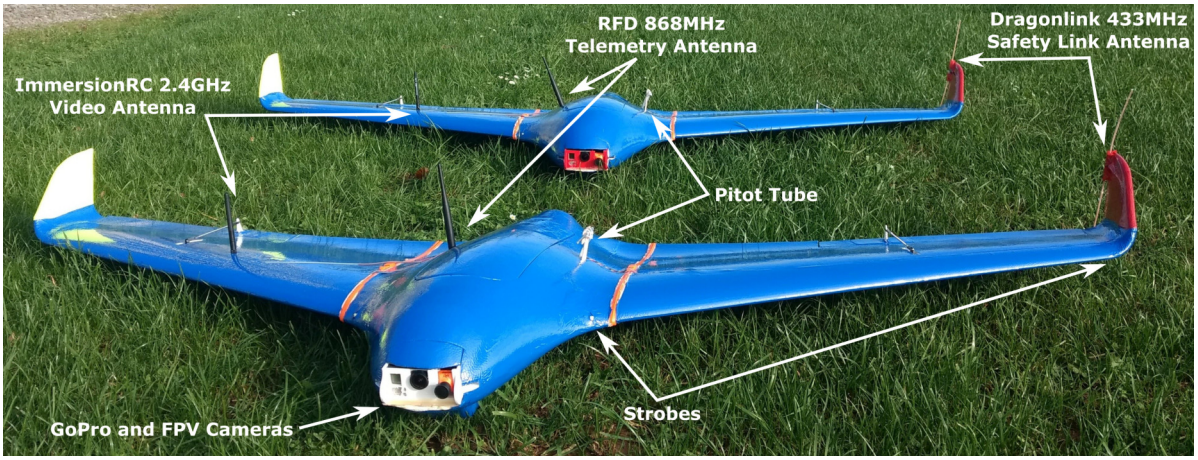

Figure 3.5: The Skywalker X8 vehicles that were flown in Guatemala, with labels of the main visible external features. Note: the video antennas shown here were used for transport and laboratory testing, and were replaced with skew planar wheel antenna for flight.

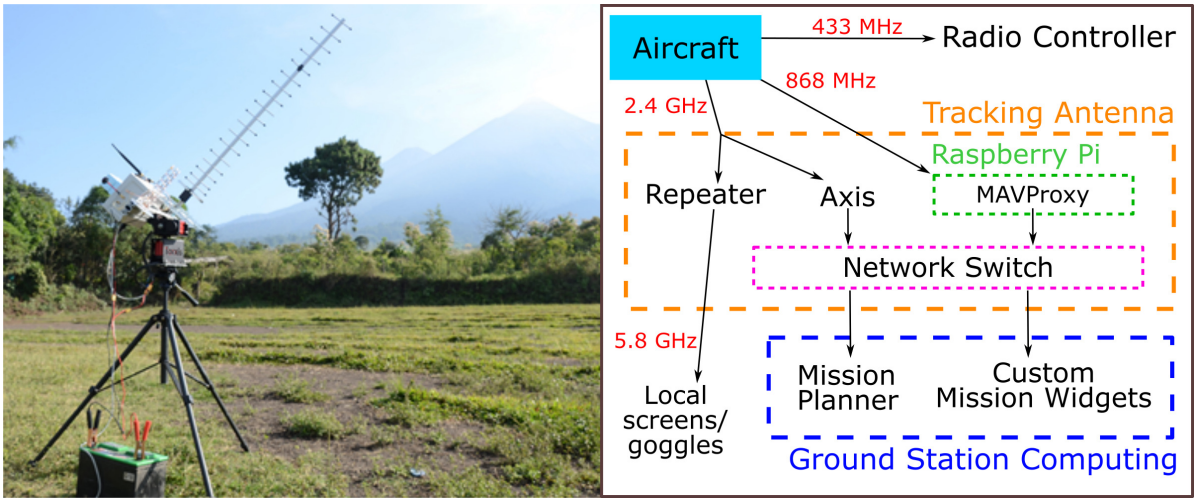

Figure 3.6: Left, Antenna Tracker equipped with high gain directional antenna. Right, system layout for data transmitted from the aircraft, with dashed lines indicating separate systems.

source software (Mission Planner, ArduPilot), a custom MavLink Terminal, and custom mission specific graphical user interfaces (GUIs).

3.5 Avionics & Computing

PixFalcon/Pixhawk autopilot units are common amongst UAV operators, usually running some version of ArduPilot code. A PixFalcon was used for the Zephyr II, and a Pixhawk for the Skywalker X8, both running ArduPlane 3.7.1 [1]. These autopilots log flight data such as altitude,

[1] `https://github.com/ArduPilot/ardupilot`

airspeed, radio channel output, and GPS location at frequencies of 10 – 50 Hz, depending on data type. A reduced-rate (approximately 1 Hz) version of the logs can be monitored in real-time from the GCS, and the full resolution logs can be analysed post-flight alongside the recorded videos. The avionics fitted are listed in Table 3.3, and were chosen for automating long-range flight.

Two cameras were mounted in the nose of both aircraft, viewing the direction of travel. A GoPro Hero 3 camera (and later Hero 4, 5, and 7) captured high resolution video for post-flight analysis, and a Run Cam Eagle first person view (FPV) camera streamed video to the GCS via a 2.4 GHz wireless link. The live video link proved effective for in-flight visual identification of the plume.

Onboard 'companion' computing took place on a Raspberry Pi 3B+ (Raspberry Pi, Pencoed, UK), mounted in the payload bay of the aircraft and powered by the aircraft flight batteries. The Raspberry Pi was connected to the Pixhawk autopilot via a serial connection, enabling the sending and receiving of MAVLink messages, which were used for controlling and monitoring the progress of the companion computer.

3.6 Summary

This chapter introduced the operating environment around Fuego and summarised the systems used to monitor the volcano. The natural hazards of operating UAS near Fuego, combined with the aim of collecting ash over large temporal and spatial intervals, means fixed-wing UAVs are the natural platform choice. These aircraft were operated from a GCS located at the INSIVUMEH observatory on the northern edge of Panimache, 9 km South-West of Fuego at an altitude of 1137 m AMSL.

The following chapter describes the development of these systems and operations in detail, including the methods developed for coping with hot & high conditions, mission planning, and hardware development.

CHAPTER 4

METHOD DEVELOPMENT

This chapter justifies the hardware selections presented in Chapter 3 and describes the development of the team's *modus operandi*. The underlying motivation behind the this work, as introduced in Chapter 1, was the automation of airborne volcanic ash collection. An effective method of operating UAVs for this purpose is presented in this chapter, detailing the progression in systems and approach across five field trips. Metrics for measuring this progress are presented in terms of flight performance and collected ash samples.

This chapter is based in part on a conference paper titled 'BVLOS Operations of Fixed-Wing UAVs for the Collection of Volcanic Ash Above Fuego Volcano, Guatemala', published by the Author in the American Institute of Aeronautics and Astronautics SciTech 2020 conference proceedings [27].

4.1 Motivation & Methodology

Ash sampling from within volcanic plumes is one of the main scientific drivers for the work carried out around Fuego, therefore the rate of ash sampling can be used as a metric for mission development across trips, changes to hardware, and changes in methods of operation. The system presented here was developed around Fuego volcano but could be implemented at other locations, making it relevant to a multitude of high-impact areas. Some of the transferable aspects of these missions include:

- High altitude gains over relatively short lateral distances, meaning that safety and efficiency can be built up around gliding back to the GCS (Ground Control Station), particularly for maximising battery usage.

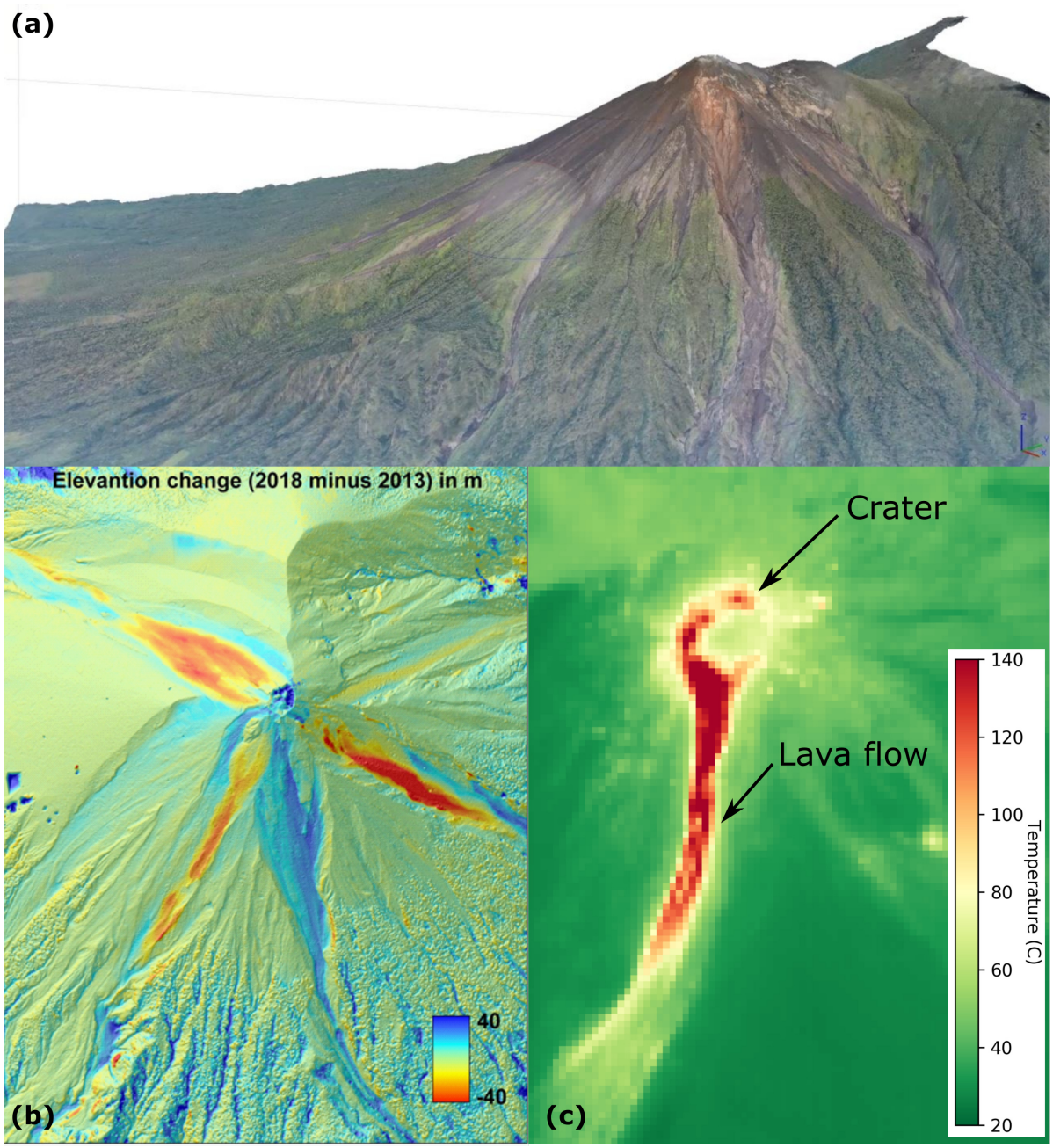

Figure 4.1: Example imagery datasets taken from various Skywalker X8 missions over Volcán de Fuego: (a) DEM generated from a single flight over the Fuego area, generated by Dr Rick Thomas (University of Birmingham, `r.thomas@bham.ac.uk`), (b) DEM subtraction comparing the elevation of Fuego's summit area between 2013 (satellite) and 2018 (UAV, Trip B), generated by Dr Rüdiger Escobar Wolf (Michigan Technological University, `rpescoba@mtu.edu`), (c) thermal infra-red image of a lava flow on Fuego taken from a flight on Trip E, generated by Dr Fred Prata (Aires, `fred@aires.space`).

- Precise aerial positioning for data collection, applicable to a multitude of experimental scientific research fields.
- High mission and flight efficiency requirements, developed here due to the low air density; applicable to any scientific missions that aim to maximise useful time in the air.

While primarily focused on ash sampling, other scientific datasets collected include visual imagery for Digital Elevation Models (DEMs) (Figure 4.1a, Figure 4.1b), infra-red images of lava-flows (Figure 4.1c), and gas sensing. Ground-based ash collection downwind of the volcano can give some indication of the ash deposition rate, however this is a form of 'inverse' problem; the desired variable is contained within the plume however the only measurable entity is the ash that falls out of it. Using UAVs to collect ash from within the plume is more directly representative of plume dynamics and a valuable input for plume dispersion models.

4.1.1 Ash Collection

An ideal ash sample would represent the mean PSD of the entire plume at the time of collection. Proving any sample to be truly representative of an unknown 3D turbulent entity for a given sampling approach is a challenging problem and in itself an emerging area of research. For the development of the UAV-based ash collection system presented here, a mechanism was devised which collected a size gradient of ash across a sticky carbon-based stub, for later analysis using Scanning Electron Microscopy (SEM). These SEM stubs (Figure 4.2) were used to collect targeted samples in forward flight, and analysed post-trip in a laboratory environment. The accuracy of

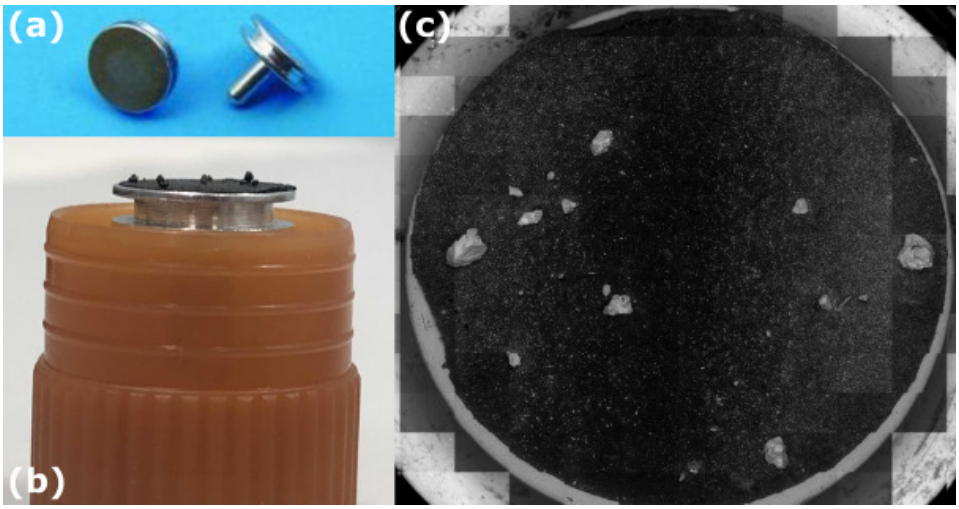

Figure 4.2: SEM stubs: (a) Two 'clean' SEM stubs (VWR International [11]), (b) a post-flight SEM stub from 25th March 2019, (c) a high resolution image of the stub in (b). The black surface is a sticky carbon-based pad, which is covered by a protective film until use, and the stalk visible in (a) is used for handling and fixing into containers (as pictured in (b)).

this sample as representative of the mean plume PSD cannot be commented on here, however it is discussed in an upcoming publication by Lucas and Watson *et al.* from the Department of Earth Sciences at the University of Bristol [Citation Pending].

SEM stubs are commonly used in volcanology for sample collection [152–154], and offer the airborne collection problem a small, lightweight way of sampling ash using a well defined method. The technique used for analysis of SEM stubs with scanning electron microscopes is well known and widely accepted in the volcanology community. The stubs consist of a small metal disc measuring 12.5 mm in diameter on an 8 mm stalk, weighing approximately 1 g each (Figure 4.2a). A sticky carbon-based pad, stuck onto the metal disc, is primed by removing a protective paper cover. It then collects material that comes into contact with it, introducing the possibility of contamination if care is not taken to isolate the stub from environmental factors away from the area of interest. After use, each stub is put into an individual container (Figure 4.2b), which is sealed and later analysed in a laboratory environment (Figure 4.2c).

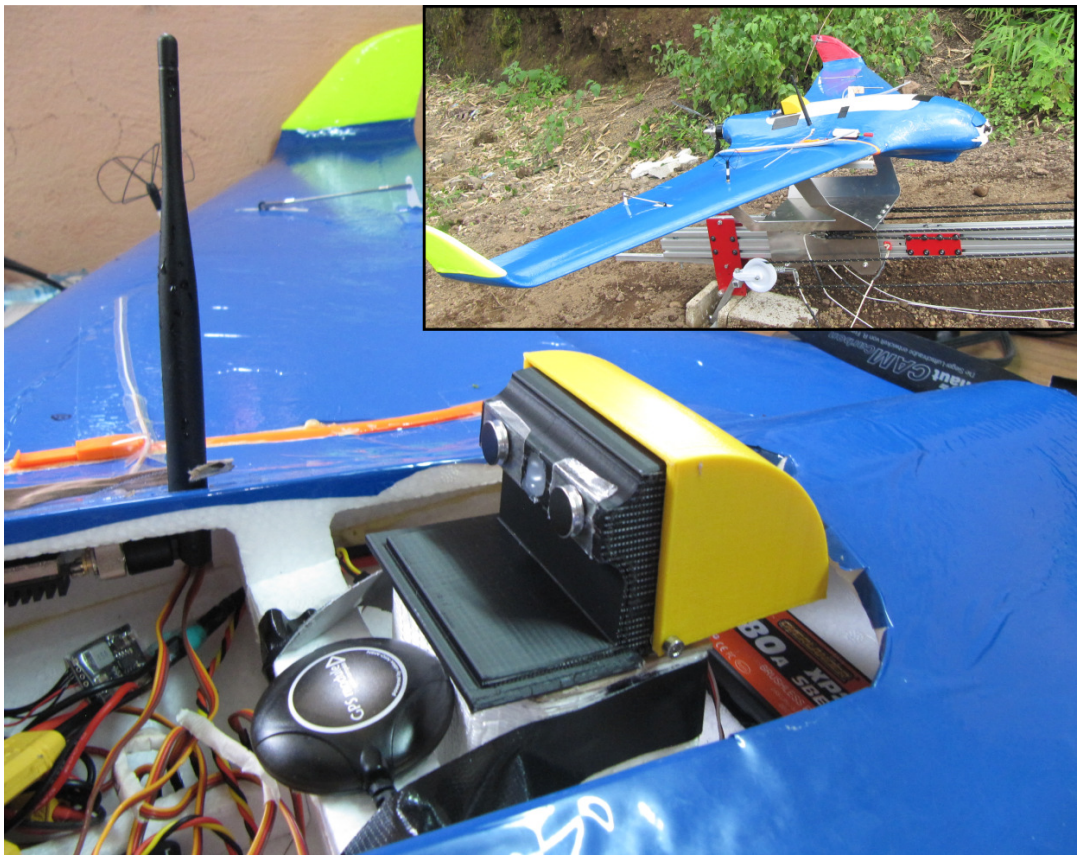

Figure 4.3: The first generation SEM stub deployment mechanism, with an inset image depicting it mounted immediately before take-off on a Skywalker X8. Note that the two stubs mounted in this mechanism must be deployed at the same time. The stubs are secured in place using double-sided tape between the stubs and the unit.

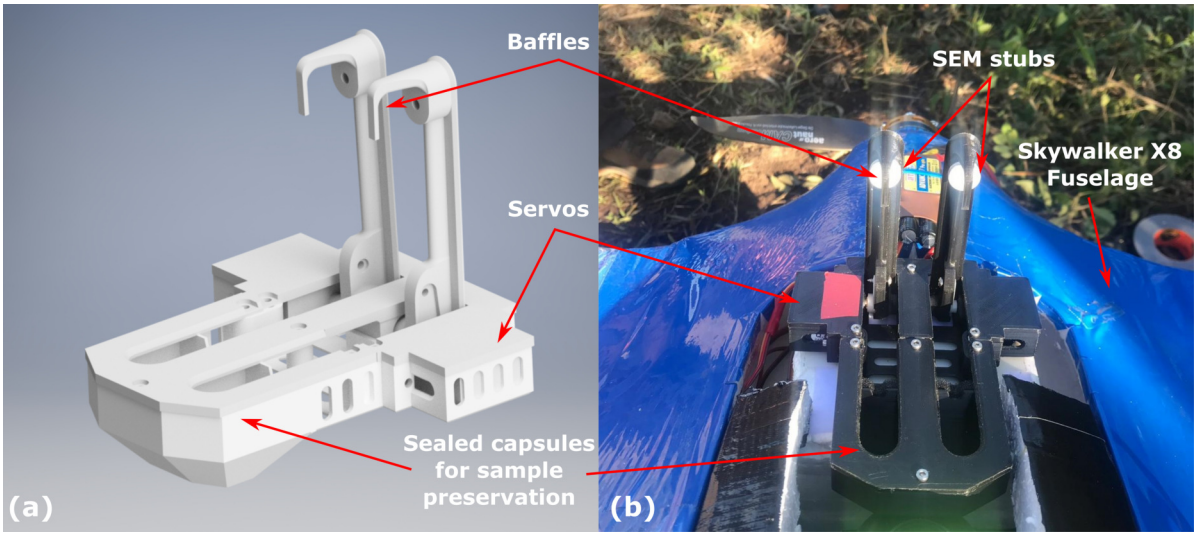

Figure 4.4: The second generation SEM stub deployment mechanism, with the ability to deploy and isolate two stubs individually. Image (a) is a CAD model for the stub deployment mechanism, and (b) shows it mounted on a Skywalker X8 pre-flight, with protective covers on the stubs. Note the baffles highlighted in both images.

Effective SEM stub integration with UAVs, particularly in environments such as that found around Fuego, requires a method of covering the stub during ground-based work, take-off, flight to and from targeted sampling, and landing. In addition to earth and 'normal' dust, the proximity of the operations centre to Fuego means there is a significant amount of ash on the ground. While biological contamination can be identified and disregarded in analysis, ash collected from the ground is extremely hard to distinguish from airborne samples and therefore must be avoided.

An ash-collection unit was developed for mounting SEM stubs on the vehicle in a position exposed to the airflow. The first version was mounted in the rear of the Skywalkers' payload bay, proud of the main hatch, exposing two stubs simultaneously upon activation of a servo-controlled cover (Figure 4.3). This cover meant the stubs could be isolated during take-off, climb, descent, and landing to prevent contamination, and was controlled manually by the operator from the GCS. This first generation unit was sized to hold both the small SEM stubs (pictured) and larger stubs, which have a diameter of 25 mm.

Following successful sample collection with the first generation mechanism, a second version was developed[1] for better aerodynamic flight efficiency and the targeted deployment of two stubs individually (Figure 4.4). The larger stubs were not used in early testing of the first generation unit, hence the second generation mechanism was designed only for the smaller (12.5 mm) diameter stubs. This second generation mechanism allowed the raising of single stubs into the airflow from individual isolating capsules, and implemented a baffle to disrupt the airflow in front of the stubs. This meant that traditional aerodynamic stagnation points on the stubs could

[1] in conjunction with Alex McConville, a fellow University of Bristol Flight Lab PhD student

be avoided, and a shadow effect of ash on the stub was formed. With little to no ash on the inner section and dense concentrations on the outer edges of the stubs, a gradient could be observed with ash densities suitable for analysis at various points (Figure 4.2c). The effect of the baffle on the measured PSD is unknown, hence careful analysis was required to ensure validity. Further modelling and testing could be done in this area to improve the validity of these results. Overall, this design was significantly more efficient, causing less drag in both the closed and the open configurations. Furthermore, the ability to directly target specific plumes per SEM stub allowed for a more controlled sampling design that could ensure variability, linking specific plume characteristics to factors such as distance from the crater and time since explosion.

Detailed volcanological analysis of the flights presented in this paper will be given in a pending publication by Lucas and Watson *et al.*, from the University of Bristol [Citation Pending].

4.1.2 Mission Requirements

A picture of the required mission type can start to be built up by combining the scientific requirements and the operating environment. With an altitude difference of some 2700 m between the GCS and the crater, missions necessitated climbs to over 4200 m AMSL while also reaching the crater area approximately 9 km from the GCS. This requires considered flight planning in order to reach the target area as efficiently as possible, maximising energy reserves for loitering and carrying out plume interceptions.

Additional factors in mission planning around Fuego are the immediate meteorological conditions; the prevailing wind means operations tend to be in the lee of the volcano, reducing the lateral distance to the plume but also causing large amounts of turbulence and high winds on summit approach flight paths. Meteorological clouds usually arrive around Fuego between 11 am and 2 pm, with the calmest and most efficient conditions found in the early morning. There is therefore value associated with operating close to the observatory and developing a pre-flight process which allows safe yet quick mission preparation, so that useful time in the air may be maximised.

Ash-collection can involve sampling in one of several permutations, for example both stubs could be exposed at the same time at a proximal or at a distal plume position. Alternatively the stubs could be exposed separately; a particular plume could be sampled at a proximal and a distal point, or perhaps consecutive plumes could be sampled at a given location. The key driver for aircraft endurance and range requirements here is the timing of Fuego's typical activity; the aircraft must be capable of spending at least 10 minutes in the vicinity of the crater in order to have an acceptable chance of intercepting a plume. Additional requirements include a nominal cumulative climb of 3000 m and scope to travel downwind with the plume for > 5 km.

The hardware chosen to conduct these missions is outlined in Chapter 3. The remainder of this chapter summarises the field trips, justifies the hardware selected, and outlines how the operation of these systems was developed to reliably achieve both scientific and engineering aims.

4.2 Trip Summaries

There were 5 field trips to Volcán de Fuego, Guatemala, between February 2017 and April 2019, the dates of which are given in Table 4.1. The primary aims of these trips are introduced in Section 4.2.1, and the secondary aims are introduced in Section 4.2.2.

Table 4.1: List of trip dates.

Trip ID	Dates
A	15th February - 24th February, 2017
B	29th October - 11th November, 2017
C	22nd June - 1st July, 2018
D	5th October - 11th October, 2018
E	22nd March - 3rd April, 2019

The composition of the team on these trips changed around a central core group of three people; the Author, supervisor Tom Richardson, and secondary supervisor Matt Watson. Contributions to the field trials presented in this thesis by those other than the Author are listed in Table 4.2. Note that all contributors were members of the University of Bristol at the time of travel, with the exception of Matt Purvis (GeoTravel Guatemala), Francis Salama (Blue Bear Systems), Rick Thomas (University of Birmingham), Mario Ferraro (University of Southampton), and Mehmet Erbil (University of Southampton).

Table 4.2: Summary of contributions made by people other than the Author to the field trips presented in this thesis.

Name	Trip	Contribution
Tom Richardson	All	Team Leader, Mission Controller
Matt Watson	All	Volcanology Expert, Ground Station Support
Matt Purvis	All	Fixer & Local Guide, Logistics Support
Colin Greatwood	A	Pilot, Systems Engineer
Kieran Wood	A, B	Ground Station Support
Francis Salama	B	Ground Station Support
Rick Thomas	B, D	Meteorology Expert, Ground Station Operator
Jim Freer	B, E	Ground Station Operator, Technical Advisor
Rob Clarke	C, D, E	Ground Station Operator, Systems Support
Mario Ferraro	E	Hardware & Data Support
Mehmet Erbil	E	Hardware & Data Support

The Author was responsible for the remaining roles and work not described by Table 4.2, including but not limited to: Ground Station Operator, Pilot, Technical Lead, Hardware & Systems Engineer, and Logistics Coordination.

4.2.1 Primary Aims

The equipment used varied across these trips as project aims progressed and a robust *modus operandi* was developed. A brief summary of the trip aims is given in Table 4.3. A list of all fixed-wing flights flown in Guatemala, including notes of the flights with SEM stubs onboard, can be found in Appendix A.

Table 4.3: Summary of field trip aims and achievements.

Trip ID	Purpose
A	Proof of concept, testing Zephyr II to summit altitude
B	Test Skywalkers, ash sampling, and in-plume temperature/humidity sampling
C	Collect photogrammetry data following major eruption on 3rd June 2018
D	Develop ash collection method and collect photogrammetry data
E	Repeated ash-collection missions, testing new mission hardware and software

Flights are hereafter identified and referred to using an alphanumeric system; the first character referring to the trip letter ID, and the following number increasing incrementally with the number of fixed-wing flights on each trip (Table 4.4). These numbers do not consider flights using other aircraft types such as multi-rotors.

Table 4.4: Example flight IDs.

Trip ID	On-Trip Flight Number	Flight ID
A	3	A3
B	8	B8
E	7	E7

4.2.2 Secondary Aims

In addition to the fixed-wing research and method development, the trips outlined above were also opportunities to collect photogrammetry data that did not focus on the summit area. The landscape around Fuego changes regularly, particularly during the wet season (May - October) when mudflows (lahars) cause extensive destruction. UAVs can be used to survey large areas with relatively high accuracy and at low cost, compared to manual surveying and satellite overpasses. Surveying was therefore conducted with both fixed-wing and multi-rotor UAVs to collect data for lahar modelling tools and updated DEMs.

In addition to the fixed-wing platforms already introduced, commercially available multi-rotor platforms were used on many of the trips listed above to survey barrancas and disaster zones. Figure 4.1a shows an example DEM generated from just one fixed-wing flight to the summit from Panimache.

A major eruption on the 3rd June 2018, with an estimated solid mass of 100 mega-tonnes, left significant deposits in the surrounding barrancas and destroyed the village of El Rodeo [155]. Figure 4.5 compares satellite images of the El Rodeo and La Reunion area before and after the 3rd June 2018 eruption. While some scientific and research tests were conducted on Trip C, the majority of the trip was spent conducting post-eruption survey work ahead of the oncoming wet season. The information gathered was used to inform national decision makers for lahar-based hazard mitigation.

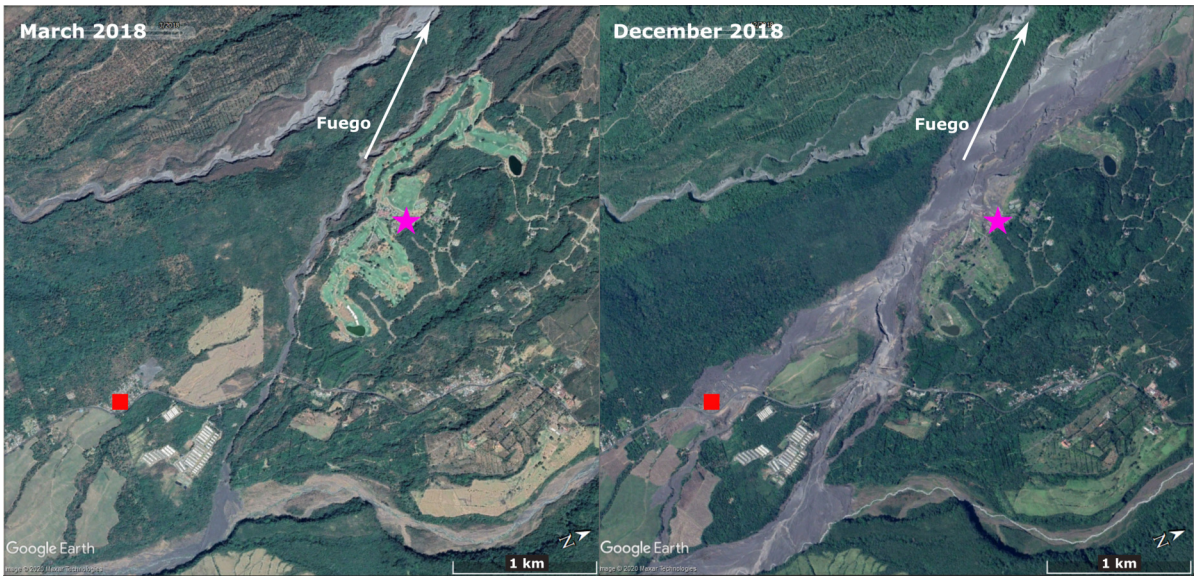

Figure 4.5: Google Earth images from before and after the eruption on the 3rd June 2018, specifically showing the area around La Reunion Golf Course (pink, star) and the village of El Rodeo (red, square). Map source: Google Earth [10].

4.3 Method Development

This section presents data from a series of flights conducted in March 2019, and compares some of them to older mission formats. Pioneering the method, SEM stubs were first flown through volcanic plumes on Trip B, with 4 samples collected for laboratory analysis. Trip E saw the team collect 24 usable samples during an 11 day campaign, 7 days of which involved fixed-wing operations, representing a significant increase in mission efficiency and system capability.

The discussion regarding the efficiency, reliability, and robustness of ash-collection UAV operations over Fuego can be separated into methods developed to date, and areas of future work. These categories can be split further, considering performance-affecting factors independently.

Initial launch attempts had a poor success rate, owing to difficulties outlined in Section 4.3.2. The development of reliable and repeatable take-off and landing methods, combined with

customised mission-specific hardware, resulted in a significantly increased launch and mission success rate. These factors are considered in the remainder of this chapter.

4.3.1 Airframe Selection

The choice of airframe was critical to the success of repeat missions. The aircraft was required to cover large distances with significant climbs at low aerodynamic and performance cost. Furthermore, it was necessary to have a rear mounted 'pusher' propeller so as to avoid the unnecessary disturbance of ash passing over the aircraft before sampling.

The terrain around Panimache meant the aircraft also needed to be robust and easily repairable. The best available landing configuration involved a minimum-speed steep approach onto cleared brush and bushes, the unevenness of the ground and vegetation giving a high likelihood of minor damage on every landing. For this reason, a foam-based airframe was chosen and covered with an iron-on wrap to help robustness and repairability.

4.3.2 Take-off

The lack of flat, open area eliminated a geared take-off run as a launching option. Initial attempts at take-off with the Zephyr II involved hand-launching and had a poor success rate. Several factors contributed to this:

- Low air density due to hot & high environment.
- High wing loading due to the combination of battery/payload/mission requirements.
- Pusher propeller so that ash flow around the aircraft was not disturbed unnecessarily. This meant the throttle could not be raised before the propeller cleared the arm of the launch-person.

Combining the above factors meant the Zephyr stall speed in this environment was relatively high. Variations on traditional hand-launch techniques were attempted, including from positions of height. Successful launches were achieved using this method, however the rate of failure remained too high for progression to larger payload-carrying vehicles. This was mainly because human consistency is relied upon when hand-launching repeatedly and, given the conditions, natural variation in launch quality proved to be unacceptably high.

An elastically-powered catapult was successfully introduced for the Skywalkers (Figure 4.6), which involved just a single point of coordination from the pilot; the aircraft would be switched to AUTO mode and, once the propeller had started to spin up, the catapult was triggered. Reducing human involvement to this level meant that automation of the launch procedure could be relied upon more heavily, and any deviation from the expected state of the aircraft in the early phases of flight were easy to recognise and respond to. System issues that are missed in pre-flight

Figure 4.6: The catapult system used for launching the Skywalker X8, being operated on Trip B by the Author.

checks often present themselves during the take-off phase of the flight, so making this phase as predictable as possible means irregularities can be quickly identified and reacted to. For similar reasons, it was desirable to fly the aircraft in AUTO mode for as much of the mission as reasonably possible.

4.3.3 Landing

Given the terrain and vegetation surrounding the observatory, the safe recovery of any traditional fixed-wing UAV (i.e. without Vertical or Short Take-Off/Landing (V/STOL) characteristics) initially proved extremely challenging. The developed method involves clearing a patch of vegetation between the launch point and the nearby barranca. This land is sloped, encouraging a single approach vector, but the go-around space is relatively clear and avoids overflying Panimache and electricity cables. There is often little to no wind, hence the single approach vector is not usually an issue, however the length of the 'land-way' is approximately 25 m and so requires precision if the aircraft is to be arrested without colliding with the vegetation at the end. This method has proved acceptable for robust aircraft with minimum speed approach vectors.

Figure 4.7 compares three consecutive auto-landing test flights (E12, E13, and E14), which took place on the morning of 27th March 2019 (Trip E), with 2 example Fly-by-Wire A (FBWA) landings from the same trip. The variation in approach altitude across consecutive auto-landing flights was significant, resulting in touchdown points of approximately +60 m, 0 m, -40 m from the middle of the landing area (measured along the line of final approach). The first and third

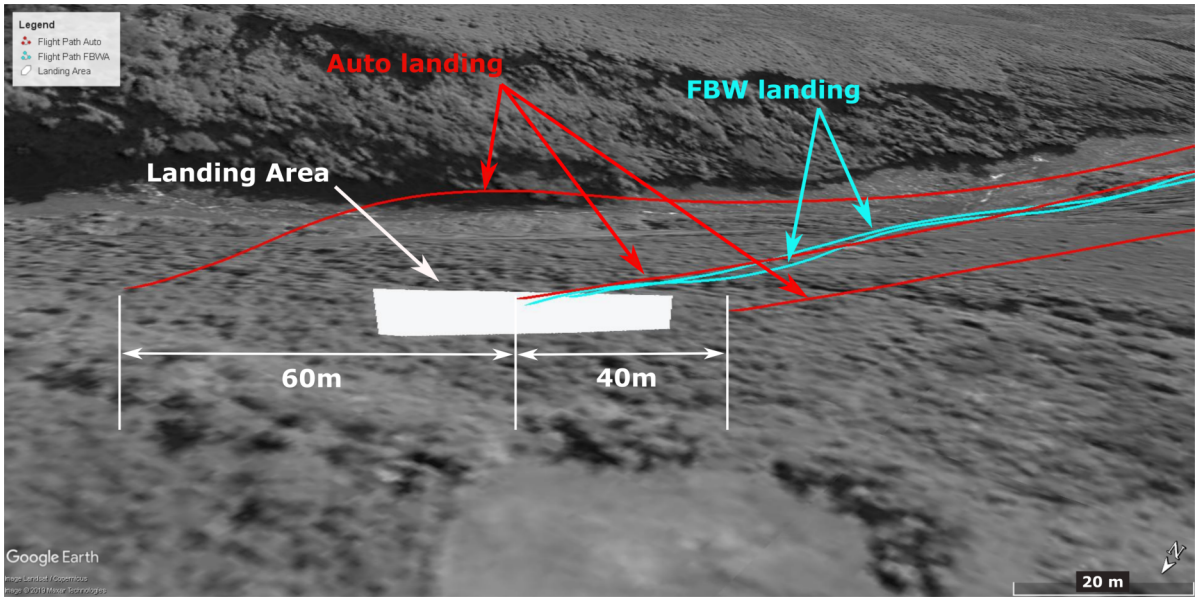

Figure 4.7: Side-view in Google Earth [10], taken from above the GCS location, comparing three automated landing attempts (red) with two example piloted (Fly-By-Wire A, see Table 4.6) landings. The distances between touchdown points for AUTO landings are given; this level of variability, which is due to barometric drift, is unacceptable and risks the airframe unnecessarily.

auto-landing flights resulted in a retrieval of the aircraft from dense vegetation. These differences in outcome are a direct result of the barometric pressure drifting significantly over the course of > 40 minute flights. The most consistent landing format thus far has involved the mission ending at an infinite circuit above the landing area, at which point the pilot takes control and lands in either Manual or FBWA mode; Manual mode allows the pilot full control over the aircraft position and speed, FBWA mode keeps the wings level unless otherwise commanded and limits both pitch and roll rates according to pre-set ArduPlane parameters (see Table 4.6).

Future solutions might involve correcting barometric drift using differential GPS systems, or implementing a UAV-specific Instrument Landing System (ILS) [156]. Other existing UAV landings methods include laser range-finders and parachutes. Their use here would be infeasible due to the uneven terrain and associated additional payload respectively.

4.3.4 Motor/ESC

The Skywalker X8 was originally fitted with a an Overlander 5045/10 720kV motor and an Overlander XP2 80A Electonic Speed Controller (ESC). The performance data of this first combination sufficed for sea-level operations which did not involve climbing great altitudes. However, a combination of characteristics meant it was poor for use around Fuego:

- Synchronisation issues meant the thrust could not be increased rapidly, limiting the thrust on take-off and causing launches with payloads to sometimes fail. Synchronisation issues

arise when the ESC timing falls out of phase with the motor coils' commutation.

- Testing at sea-level in the United Kingdom suggested suitable in-flight thrust and performance data, however a combination of the hot & high conditions and mission profile meant the aircraft climbed poorly in Guatemala, with a throttle setting between 90% and 100%.

The climb throttle setting meant there was little reserve available for unpredictable atmospheric conditions such as turbulence or headwinds. The revised motor/ESC combination, an AXi 4120/14 660kV motor and a Jeti Adv 77 Pro Opto ESC, was significantly better suited to the conditions in Guatemala and excellent performance was achieved. The revised combination typically climbed at 65% throttle, and the proportion of failed take-offs and aborted flights reduced significantly after their introduction.

4.3.5 Mission Plan

The most significant factor to affect mission success was the development of mission waypoint plans. Figure 4.8 shows three example flight tracks, and Table 4.5 gives some key mission data, demonstrating the evolution of mission planning.

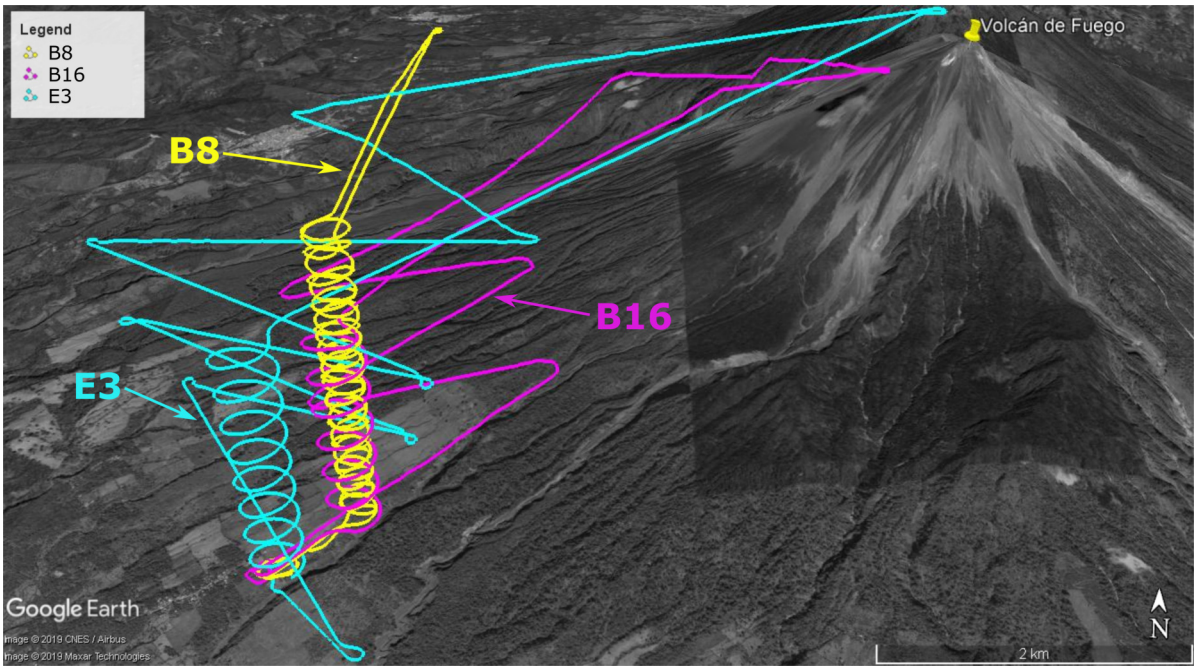

Figure 4.8: Evolution of flight tracks, shown in Google Earth [10]. The first mission plans involved spiral climbs and descents (B8). Leg-climb plans were trialled (e.g. B16) and reliable variations developed (e.g. E3). These involved climbing in long straight legs, with spiralling descents.

The take-off and climb of all flights took place in AUTO flight mode. Early flights involved a spiral climb to above summit altitude (Figure 4.8, B8), initially used as part of a conservative

Table 4.5: Key mission data relating to each of the flights shown in Figure 4.8. Comparing the values, Flight E3 gained the most altitude and travelled the most 'Air-Distance', with the least battery used, demonstrating significant progression in flight efficiency.

Flight ID	**Flight Time (mins:secs)**	**Battery Used (mAh)**	**Altitude Gained (m)**	**Air-Distance (km)**
B8	40:35 mins	9,616 mAh	3,033 m	49.6 km
B16	34:19 mins	10,099 mAh	2,858 m	40.6 km
E3	35:09 mins	8,455 mAh	3,086 m	51.2 km

mission development strategy. These initial flight plan styles were chosen so that lateral proximity to the aircraft could be maintained in the climb, maximising the signal quality from the aircraft and removing any chance of collision with terrain. In addition, the location of the spiral climb point meant that the aircraft was within gliding distance back to the 'home' location soon after launch. In the case of unforeseen circumstances or system issues the aircraft was then already in a position where chances of recovery could be maximised.

As a trade-off, however, performance penalties became apparent due to a loss of lift, proportional to the cosine of the roll angle (i.e. 25° roll reducing lift by approximately 10%), and a peculiar maximum-climb behaviour of the flight controller. When in AUTO mode, and commanded to climb to high altitude, the controller would increase the throttle to 100% and lift the nose as high as possible in an effort to climb. This would often lead to flying on the back of the drag curve, where the power required to overcome induced drag is greater than that required at a higher airspeed. These conditions were sometimes found to contribute to reaching a premature service ceiling.

The penalties associated with these spiral-climb flight plans were mitigated by placing waypoints such that the climb took place over a series of straight legs. Initial versions of this oriented the mission along the axis between Panimache and Fuego's summit (Figure 4.8, B16) however these flights showed problematic climb performance. This was likely a combination of the original motor/speed-controller and the alignment of the climb legs with the prevailing wind vector; the combination of the motor and speed-controller could not cope with the strong headwinds or turbulence in the lee of the volcano and resulted in high-drag flight states which wasted battery and limited mission success rates.

These straight-leg climb flight plans developed to look like E3 in Figure 4.8, with cross-wind climb legs gradually moving towards Fuego until summit altitude was reached. The climb legs were then directed towards the summit with significant coverage of the downwind cross-wind area in case of plume interception. A diagram demonstrating this approach to flight planning is given in Figure 4.9. Keeping the climb legs near the ground station also meant maintaining good communications signals with the aircraft and it remained straightforward to return the aircraft to 'home' in the event of unexpected behaviour or issues.

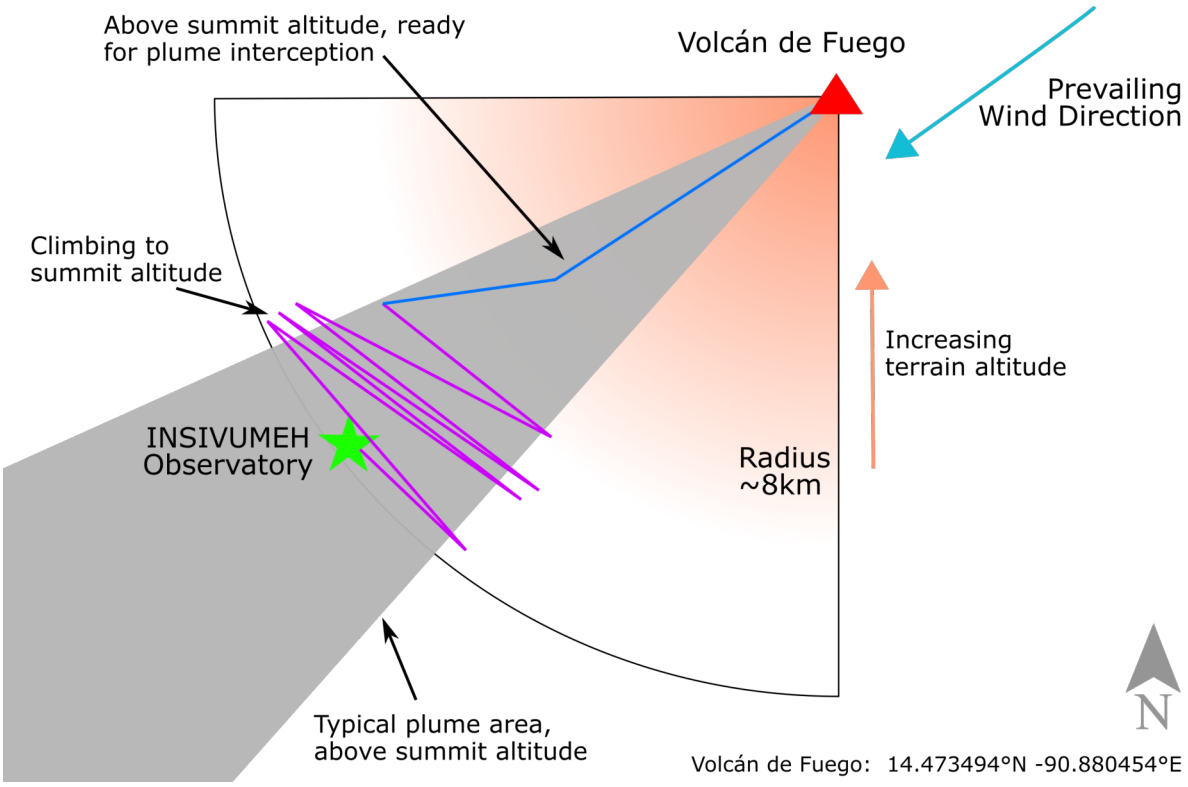

Figure 4.9: Figure demonstrating the approach taken for flight planning in the later form of straight-leg climb paths (i.e. E3). The likelihood of intercepting a plume was maximised by climbing in legs near the observatory to near-summit altitude, with the final climb section directed towards the summit and located in the prevailing plume zone.

An additional factor that was found to benefit flight efficiency was increasing the cruise airspeed during the climb and summit-approach phases. The mission aim usually depended on reaching the summit area, however the typical wind speed and direction meant that speed over ground could be severely limited. With ground-speeds as low as 2 m/s, raising the cruise speed parameter from 22 m/s to 25 m/s meant reaching the summit significantly quicker; this was found to be beneficial despite using more battery capacity. Further work in this area involves modelling the drag curve in quasi real-time, for optimising airspeed and flight performance during these long climbs.

4.3.6 Battery Chemistry

Lithium-Polymer (LiPo) batteries are a standard energy source for small UAS such as the Skywalker X8. Two 8,000 mAh LiPos typically yielded 10 minutes of loiter time at or above summit altitude, with a total flight time of approximately 40 minutes. Lithium-Ion (LiIon) battery technology has improved over recent years, now offering excellent energy density and sufficient, albeit limited, peak current output.

Bespoke four-cell LiIon batteries were introduced on Trip E, with sets of batteries combined in parallel to reach a peak current draw (approximately 70 Amps) comparable to the LiPos on take-off. A total onboard capacity of 27,200 mAh significantly increased the amount of usable battery, however also meant an unavoidable weight increase of 250 g. This consequently limited the payload capacity for these flights. LiIon climb performance was comparable to that of the LiPo batteries, however the increased capacity gave over 30 minutes of loiter time at and above summit altitude. The success of the LiIon configuration relied on the progress made before Trip E; the additional weight and limited current would have been severely limiting before the revision of the motor/speed-controller and significant experience operating the catapult.

More time at and above summit altitude allowed more precise ash sample collection, which suited the sample scheme of the SEM stubs. The impact of a greater energy reserve meant that the aircraft could spend more time in the useful area above summit altitude, downwind of the crater, leading to greater mission efficiency and previously unavailable scientific datasets. These primarily involved collecting ash samples from the same plume with large temporal and spatial separation.

4.3.7 Flight Mode

The ArduPlane code base used for both the Zephyr II and the Skywalker X8 has a number of flight modes, used to vary the level of autonomy and behaviour type. Some examples of commonly used modes are given in Table 4.6.

Table 4.6: Descriptions of some key ArduPlane flight modes. Note, PWM is the abbreviation for Pulse-Width Modulation.

Mode	Function
MANUAL	Direct control of servo PWM values from pilot controller, no autopilot involvement
FBWA	Fly-By-Wire A: Wings kept level unless commanded otherwise, rates and magnitudes of roll and pitch limited by autopilot, full pilot control of throttle
AUTO	Fully automated flight with no pilot input required, following a predetermined flight plan
RTL	Return-to-Launch: Overrides any autopilot commands and returns the vehicle to a pre-set 'home' location and altitude
GUIDED	Aircraft goes to a 'target' waypoint location and altitude, which can be modified externally during flight (i.e. from the ground station or a companion computer)

Different flight modes involve varying levels of pilot involvement, and therefore varying levels of autonomy. UAS autonomy can be described by a number of definitions, however they usually involve an autopilot unit controlling the actuators and throttle such that controlled flight is achieved. NATO define four levels to classify the autonomy of a UAV [157]:

- **Level 1, Remotely Controlled System:** System reactions and behaviour depend on operator input.
- **Level 2, Automated System:** Reactions and behaviour depend on fixed built-in functionality.
- **Level 3, Autonomous Non-Learning System:** Behaviour depends on fixed built-in functionality or upon a fixed set of rules that dictate system behaviour (goal-directed reaction and behaviour).
- **Level 4, Autonomous learning system with the ability to modify rules defining behaviours:** Behaviour depends upon a set of rules that can be modified for continuously improving goal directed reactions and behaviours within an overarching set of inviolate rules/behaviours.

For example, a UAV following a set mission plan in AUTO mode can be classified as Level 2 or Level 3 however a UAV in MANUAL mode is Level 1. AUTO mode was initially adopted for entire missions, reducing pilot workload and ensuring predictability. Reliance on automation is beneficial both in flight and post-flight when conducting analysis. Without direct pilot input, the behaviour of the autopilot can be broken down and causes for behaviours identified with relative ease. AUTO mode is also more efficient than manually piloted flight with less excessive thrust and better-coordinated manoeuvres.

For missions around Fuego there came a point when summit altitude was reached consistently but plumes were not being intercepted reliably. AUTO missions must be pre-programmed however plume altitudes vary from 200 m to > 1 km above the summit depending on the explosiveness of the eruption. Plume bearing (measured from the crater) can also vary significantly; wind directions can change rapidly, and wind-shear can cause eruptions with differing explosiveness to travel in very different directions. These factors make the prediction of plume locations and altitudes extremely challenging, particularly for distal sampling. While it was possible to keep some missions in AUTO mode for plume sampling (Figure 4.10, E3), it became commonplace to intervene using FBWA mode to intercept plumes as appropriate for the experimental sampling configuration.

A significant contributor to the successful and repeatable collection of ash was the capability of the system and crew to use FBWA mode when the aircraft was above summit altitude, flying first person view (FPV) using the live video feed. One successful approach involved orbiting a holding point close to the crater in AUTO. When an eruption occurred the vehicle would be switched to FBWA and the plume intercepted, before returning to the orbit point in AUTO if appropriate to the ash collection configuration (i.e. the mission involved a second plume interception).

Two flight tracks, both of which collected valuable ash data, are presented in Figure 4.10. The first flight (E3) took place entirely in AUTO mode however the second flight (E9) involved both

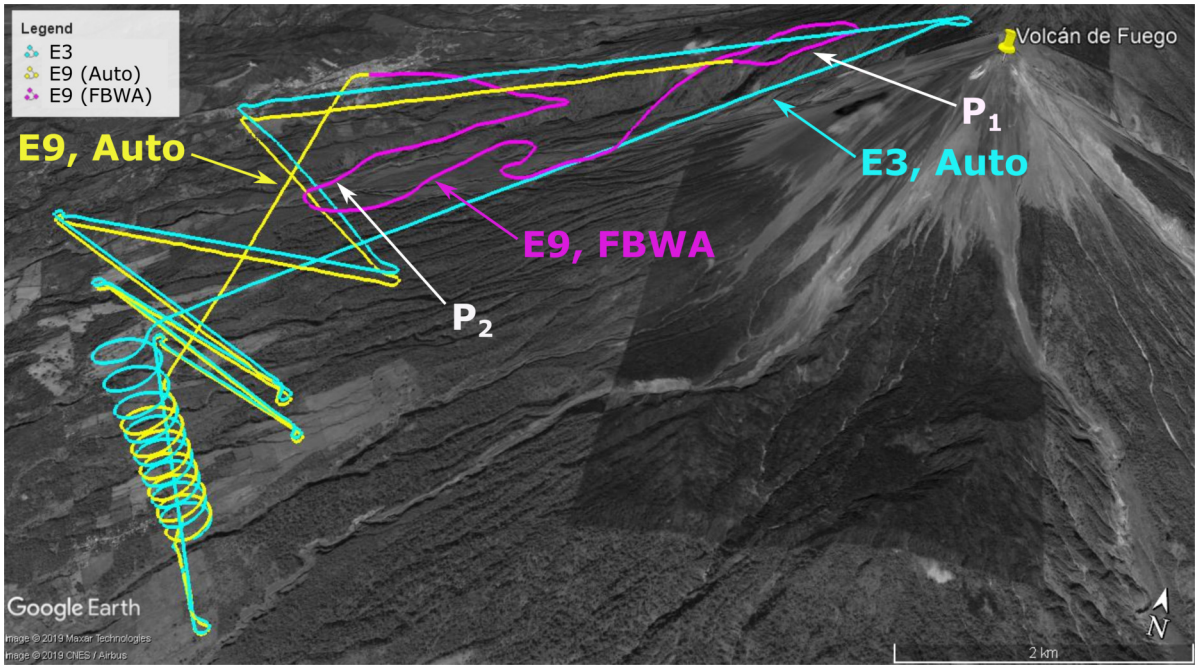

Figure 4.10: Two flight tracks from Trip E, each illustrating effective approaches to in-plume ash-sampling. Flight E3 took place on 25th March 2019 in AUTO mode, and Flight E9 took place on 26th March 2019 using both AUTO and FBWA modes to collect ash at points P_1 and P_2.

AUTO (yellow) and FBWA (pink) modes. Flight E9 was left in AUTO until it was near a plume, at which point FBWA mode was entered in order to change course such that the middle of the plume was intercepted (at position P_1). The aircraft was then turned and flown downwind such that it could re-intercept the centre of the plume approximately 3 km downwind of the first sample point (at position P_2). The flight mode was then switched back to AUTO in order to return the aircraft to the GCS. The ability to safely take control of the aircraft in BVLOS flight depends on strong connections on all three UAV/GCS communications channels, as well as an experienced flight team managing mission progress and aircraft state throughout.

While use of FBWA mode allows effective application of the UAV as a tool to collect ash, the behaviour and performance of the system is not repeatable due to the natural human factor. It remains desirable to run entire missions at a higher autonomy level, in order to maximise flight efficiency. A combination of elements are required to elevate the level of autonomy while regularly achieving the mission goal of sampling within plumes, including the automation of plume detection and a plume re-interception scheme. This involves detecting and sampling a plume, then flying a route which means the aircraft re-intercepts the plume at a given separation from the first sample location (see Chapter 7).

4.3.8 Measuring Progress

A combination of the factors outlined in this section led to a reliable and repeatable method of operations. Figure 4.11 demonstrates the effectiveness of the systems and method development, summarising key mission characteristics for 42 flights from Trip B to Trip E.

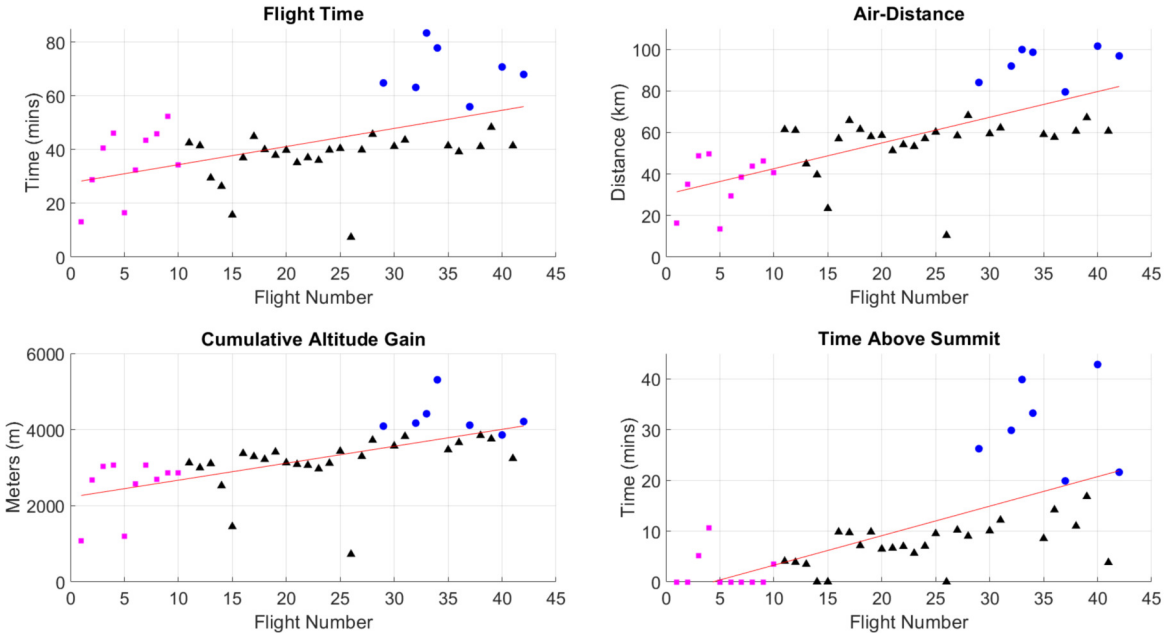

Figure 4.11: Comparison of some key flight metrics across 42 Skywalker flights, spanning Trips B to E. The data are Flight Time (Top Left), Air-Distance (Top Right), Cumulative Altitude Gain (Lower Left), and Time Above Summit altitude (Lower Right). Early motor/ESC (LiPo) flights are represented by pink squares, revised motor/ESC (LiPo) flights are represented by black triangles, and LiIon (revised motor/ESC) flights are represented by blue circles.

The team flew a total of 60 fixed-wing flights (Appendix A), 42 of which have been selected for consideration here based on comparability. The 18 flights that have been disregarded include those on Trip A with the Zephyr II, those with a GCS not located at the INSIVUMEH observatory (i.e. La Reunion Golf Course), and those with a mission profile/purpose that is distinctly different to the ash-collection mission aim. For example, barranca photogrammetry missions flown from La Reunion are not considered however crater photogrammetry missions flown from the INSIVUMEH Observatory are included because of the similarity in mission profiles.

Moving clockwise from top left, Figure 4.11 shows positive trends in flight time, distance travelled through the air, time spent above summit altitude, and cumulative altitude gain, against (total) flight number. Flight time is calculated by subtracting the time of take-off from the landing time, detected automatically using the highest and lowest accelerations respectively in the aircraft x-axis (as sensed by the autopilot's IMU). Air-Distance is calculated by integrating the airspeed of the aircraft between launch and landing, and is equal to the distance flown over

the ground in zero-wind conditions. The cumulative altitude gain is calculated by summing any positive gain in GPS altitude across all time steps between take-off and landing.

The principle aim of these missions was to spend at least 10 minutes above summit altitude in the vicinity of the crater in order to have a reasonable chance of collecting useful ash samples. Figure 4.11 demonstrates the progression of these missions; 10% of the first ten flights achieved 10 minutes above summit altitude, as opposed to 80% of the final ten flights. The minimum altitude gain required to reach the summit area is approximately 2600 meters, however environmental factors mean this value is typically closer to 3000 meters. The corresponding plot in Figure 4.11 shows inadequate climb performance for initial flights however the missions exceeded the minimum required gain by a considerable margin by the end of the project. The Air-Distance is a descriptive parameter which helps build an understanding of the progression in aircraft capability, as does flight time when combined with the other plots shown.

One notable outlier in Figure 4.11 is flight E8 (Flight Number 26 in Figure 4.11), which was aborted during the climb due to erratic airspeed readings affecting the flight controller. This was due to a grass seed in the pitot tubing that did not affect the pre-flight tests. The seed was removed and the day's flights continued without issue.

The development of reliable launch and recovery methods is not represented in Figure 4.11 however they were vital to the success of launch attempts and consequently to the development of a robust volcanic monitoring capability. This is evident in part by the fact that 23 of the 42 flights shown took place on Trip E. Some of the key factors that influenced the positive trends in Figure 4.11 include:

- Revision of the motor/speed-controller combination
- Climbing in straight legs, limiting the flight and climb path angle
- Re-orientation of the climb legs such that the climbing phase maintained an approximately constant lateral distance to Fuego
- Introducing LiIon battery packs

While Figure 4.11 shows progress in terms of flight performance and mission characteristics, it does not represent the progress made in terms of ash collection. Figure 4.12 shows the cumulative SEM stubs deployed per flight, considering the same 42 flights as above. Note that the stubs fitted on Flight E8 are included here despite not being deployed in an ash plume; this is because they are control samples.

12 SEM stubs were deployed on Trip B however this was after some initial test flights which were necessary in order to test the new platform. Trip C followed the large eruption on the 3rd June 2018 and therefore primarily aimed at gathering photogrammetry data for post-disaster analysis, however two important flights took place on the last day of the trip. These data points are shown by the two black points in Figure 4.12, and the first two black triangles in Figure

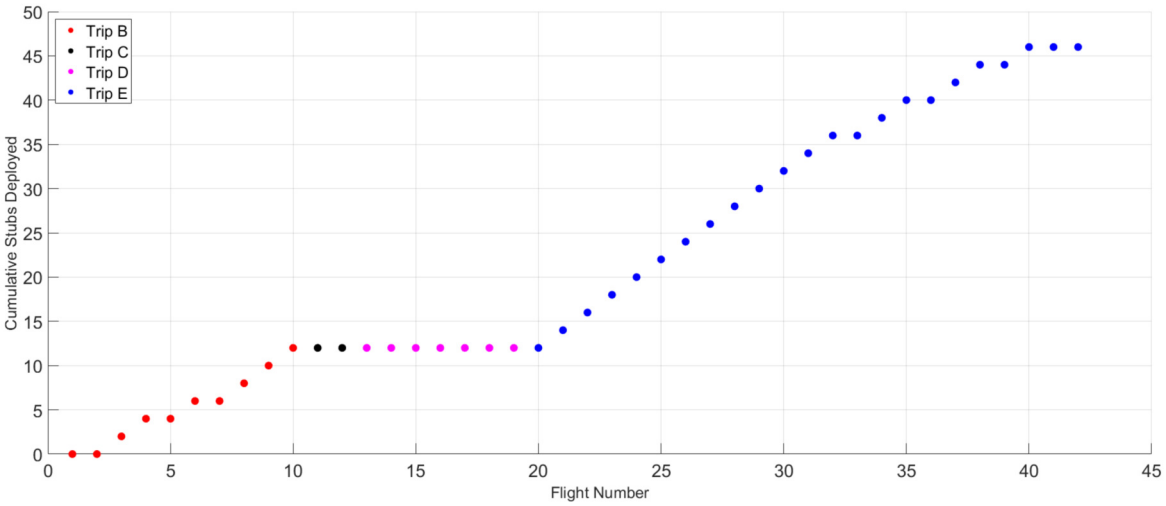

Figure 4.12: Graph showing the cumulative number of deployed SEM stubs against fixed-wing flight number for 42 missions around Fuego since the start of the project.

4.11; they were the first flights to the summit from the observatory using the revised motor/ESC combination. While no samples were collected on Trip D, the method of operations was developed extensively. This made it possible to collect a further 34 samples on Trip E, demonstrating significant progress in terms of achieving the scientific aims of the project.

4.4 Conclusions

This chapter presented a series of factors which were key to the development of a reliable and repeatable method of UAV-based in-plume volcanic ash collection, specifically around Fuego. The Skywalker X8 was used repeatedly in this environment, with hardware changes yielding positive results. The most significant factor that contributed to successful method development was the evolution of a safe, efficient, and reliable flight plan. With time above summit altitude becoming a reliable feature of missions, the accuracy with which plumes could be intercepted became the focus. The best results to date involved some pilot intervention (in FBWA mode) to steer the aircraft towards the plume. The effectiveness of the overall method is apparent when the number of collected ash samples is considered; 34 were collected in the 9 day operations window on Trip E, with 15 of the most recent flights around Fuego spending time above summit altitude. These are significant markers of successful missions, demonstrating a reliable and effective solution to the in-plume ash collection problem. Chapter 5 considers the autonomous sensing and identification of volcanic plumes using low power, light weight sensors.

CHAPTER 5

PLUME SENSING

This chapter describes a series of proof-of-concept BVLOS UAV flights which enabled the development of a plume sensing metric using small, light, and low-power sensor payloads. These flights reached a range of 9 km from the GCS and an altitude of 4410 m AMSL over Volcán de Fuego in Guatemala, interacting with the volcanic plume on multiple occasions across a range of differing conditions. In order to repeatably and autonomously target volcanic plumes, it is necessary to have some method by which to identify them. This chapter presents the results of four plume interception flights from Trips A to D, with the aim of characterising flight through a plume using a number of sensors. A plume detection metric was developed using a combination of flight data and atmospheric sensor data, and applied to the flight logs retrospectively to assess and identify flight through volcanic plumes.

This chapter is based on a journal paper titled 'Remote sensing and identification of volcanic plumes using fixed wing UAVs over Volcán de Fuego, Guatemala', published by the Author in the Journal of Field Robotics [28], and conference paper titled 'BVLOS Operations of Fixed Wing UAVs for the Collection of Volcanic Ash Above Fuego Volcano, Guatemala', published by the Author in the American Institute of Aeronautics and Astronautics SciTech 2020 conference proceedings [27].

5.1 Methodology & Sensors

The methods and sensors listed in this section were chosen for flight testing around Fuego in order to develop a way of automatically sensing and identifying a plume. The merits of each will be discussed individually, highlighting the ways each sensor can be used to distinguish in-plume flight and why they cannot be used individually for reliable and robust measurements.

Volcanic plumes, in particular those found at Fuego, are turbulent volumes of gas and ash that form after an eruption. While an initial eruption also includes ballistics, a plume as considered here continues to exist in the atmosphere for a significant length of time, dispersing gradually as it drifts downwind. It is this rate of dispersion, and in particular the variation in ash Particle Size Distribution (PSD) with time, that is of scientific interest and a key driver of this work. As described in Chapter 4, Scanning Electron Microscopy (SEM) stubs were used for the targeted collection of this airborne ash.

5.1.1 Turbulence

Visual observations of volcanic plumes suggest they are turbulent relative to ambient air due to the interaction of recently expelled hot ash and gases with the ambient atmosphere. Both the PixFalcon and PixHawk autopilot units are fitted with a variety of onboard sensors, which are used and logged by the flight control system; gyroscopes and accelerometers measured rotation and acceleration around and along each aircraft axis respectively. Turbulence was sensed by these accelerometers, and to some extent the gyroscopes, however these sensors cannot distinguish plume turbulence from other sources of turbulence such as clouds.

Proper quantification of turbulence requires specialist devices, such as a 5 hole gust probe linked to an accurate, fast-response inertial measurement unit (IMU). These data can then be used to calculate absolute turbulence values using methods such as those presented by Stull [158]. The nature of the operating environment meant these delicate devices would have needed sophisticated protection measures, which was not conducive to minimising the equipment required to sense the plume. Vertical (Z-axis) acceleration is sensed in the aircraft's body-referenced frame by the autopilot IMU and can be considered sufficiently representative of the expected scale of turbulence for plume identification testing. These data could easily be incorporated into an onboard detection system as no additional mass or significant computing power would be involved.

5.1.2 Temperature & Relative Humidity

When a plume reaches maximum altitude it becomes neutrally buoyant because the mean temperature within it is no longer greater than that of the surrounding air. Turbulence within the plume therefore suggests possible heat-related movement of air masses, with warmer pockets of fluid rising to the top before cooling and falling. If the plume is flown through after the initial eruption then it is likely that any measured temperature change would be minimal.

Volcanic emissions from Fuego originate from the subduction zone off the coast of Guatemala in the Pacific Ocean, which could lead to a humid plume upon expulsion into clear air. It is likely that this will vary with distal range from the summit. Should there be a change in either temperature or relative humidity (Temp/RH) upon entering the plume, these sensors may be

added to the plume detection metric with relative ease due to their small form factor, light weight, and low power requirements.

Two types of Temp/RH sensor were tested, both of which are described below and shown mounted to the fuselage of a Skywalker X8 in Figure 5.1. One was available to buy commercially and the second was developed by Dr Rick Thomas from the University of Birmingham [159].

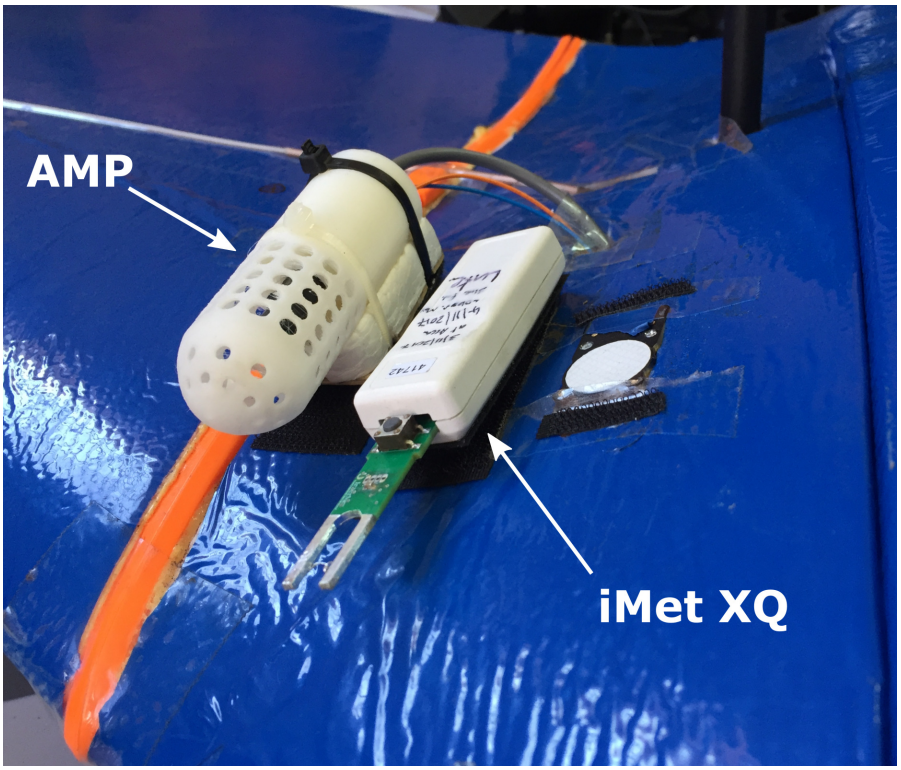

Figure 5.1: Temperature and Relative Humidity sensors. The AMP (left) and iMet (centre) sensors mounted externally on one of the Skywalkers. The unit on the right is not relevant to the sensing of temperature or relative humidity.

iMet-XQ and iMet-XQ2

Manufactured by International Met Systems, these sensors contain a GPS unit, bead thermistor, capacitative RH sensor, and piezoresistive pressure sensor, logging at 60 Hz. The manufacturer's specifications for the XQ are given in Table 5.1. This unit was mounted inside the nose of the aircraft, with the sensor tips in the airflow entering the aircraft, or externally on the forward section of the fuselage, and used on Trips B – D.

The iMet-XQ2 was used for Trip E, and offers better resolution and response time compared to the XQ model, and also has an updated GPS module. The XQ2 was not used for the development of the metric described in this chapter, however its use on subsequent trips provided valuable additional data. The manufacturer's specifications for the XQ2 are given in Table 5.2.

Table 5.1: iMet-XQ Sensor specifications.

Sensed Parameter	**Humidity**	**Temperature**	**Pressure**
Sensor Type	Capacitive	Bead Thermistor	Piezoresistive
Range	0% – 100% RH	-95 – 50°C	10 – 1200 hPa
Response Time	5 s at 1 m/s	2 s	10 ms
Accuracy	± 5% RH	± 0.3°C	± 1.5 hPa
Resolution	0.7% RH	0.01°C	0.02 hPa

Table 5.2: iMet-XQ2 Sensor specifications.

Sensed Parameter	**Humidity**	**Temperature**	**Pressure**
Sensor Type	Capacitive	Bead Thermistor	Piezoresistive
Range	0 – 100% RH	-90 – 50°C	10 – 1200 hPa
Response Time	5.2 s at +5°C	1 s at 1 m/s	10 ms
Accuracy	± 5% RH	± 0.3°C	± 1.5 hPa
Resolution	0.1% RH	0.01°C	0.01 hPa

Avian Meteorological Package

The Avian Meteorological Package (AMP) is a modified version of the Eagle Sensor Package (ESP) described by Thomas *et al.* and Greatwood *et al.* [1, 159]. The ESP evolved to use an Atmel M0 cortex chip on a commercially available micro-controller (Adafruit Feather), and a daughter board was constructed containing a GPS chip, accelerometers, magnetometers, a BMP280 pressure sensor, a GE Fastip sensor, and I^2C connections for the P14 rapid RH sensor, which were all logged at 5 Hz.

The AMP used here is an alternative to the iMet, with greater flexibility for additional sensors and integration with UAV systems. To ensure the sensors were in the best airflow possible, without creating disproportionate amounts of drag, it was mounted on the upper surface of the fuselage near the leading edge. The specifications are given in Table 5.3. Although the AMP is pending validation for sensing of absolute values, the data collected qualitatively shows its potential use.

Table 5.3: Avian Meteorological Package (AMP) Sensor specifications [1].

Sensed Parameter	**Humidity**	**Temperature**	**Pressure**
Sensor ID	IST P14 Rapid	GE Fastip	Bosch BMP280
Sensor Type	Capacitive	Glass Bead Thermistor	MEMS
Range	0% RH to 100% RH	−30°C to +40°C	300 hPa to 1100 hPa
Response Time	<1.5 s	0.1 s	5.5 s
Accuracy	+/- <0.75% RH	+/- 0.2 °C	+/- 1 hPa
Resolution	0.10% RH	1 °C	0.01 hPa

5.1.3 Visual Cameras

Visual identification of proximal plumes is possible using standard red-green-blue (RGB) cameras. Set to record at a resolution of 1920 × 1080 pixels a GoPro Hero 3, fitted in the nose of both the Zephyr and Skywalkers, was used to assess whether the UAV was in the plume on a flight-by-flight basis. Upon approaching the plume, the darker ash cloud took up an increasing proportion of the frame, and the relative enormity of the plume compared to the UAV meant if the aircraft entered the plume the image changed colour completely. This was most clearly seen by comparing the colour of a 'constant' object (such as distant sky or clouds) at a point when the aircraft is known to be in clear air with various points going into and out of the plume.

The GoPro Hero 3 provides a good post-flight qualitative indicator of plume interaction. However, real-time image processing requires significant computational resources, and the clarity of the plume depends on both ash density and time since eruption. While increasing onboard computing power is possible, it minimises the remaining payload available for scientific instrumentation and is therefore undesirable. Visual indication was therefore used instead as a method of verifying the data collected by the accelerometers and Temp/RH sensors.

5.2 In-Plume Detection Metric

The following metric has been developed for real-time plume sensing, using data sets from Trip A to Trip C; t is the flight time in seconds, alt is altitude in meters AMSL, $absH$ is the absolute humidity, $relH$ is the relative humidity, and z is the Z-axis vertical acceleration in m/s^2. $P_{inPlume}$, given in Equation 5.1, represents the probability of the aircraft being in a plume.

$$P_{inPlume} = \begin{cases} C_{inPlume} & C_{inPlume} < 1 \\ 1 & C_{inPlume} \geq 1 \end{cases} \tag{5.1}$$

$$C_{inPlume} = C_{alt}.C_{absH}.(C_{accZ} + C_{relH}) \tag{5.2}$$

C represents the components of the formula, the individual definitions of which are defined in Equations 5.3 - 5.6.

$$C_{alt} = \begin{cases} 0 & alt < 3860 \\ 1 & alt \geq 3860 \end{cases} \tag{5.3}$$

$$C_{absH} = \begin{cases} 0 & absH > 4.95 \\ 1 & absH \leq 4.95 \end{cases} \tag{5.4}$$

$$C_{accZ} = \begin{cases} 0 & i \leq 0 \\ \sqrt{\frac{1}{500}\sum_{n=i}^{500+i}(z_i - \mu)^2)} & i > 0 \end{cases} \tag{5.5}$$

$$\text{where} \quad i = 50(t-10) \quad \text{and} \quad \mu = \frac{1}{500}\sum_{n=i}^{500+i} z_i$$

$$C_{relH} = \frac{relH_t - min\{relH|_j^{7200+j}\}}{50} \tag{5.6}$$

$$\text{where} \quad j = 60(t-120)$$

This initial plume-detection metric was developed during the post-processing of four flight logs, and makes the following assumptions:

- The plume maintains near-constant altitude in the flight area around Fuego, such that a lower altitude limit of 3860 m AMSL can be imposed
- The absolute humidity of the plume is below 4.95 $\mathrm{g\,m^{-3}}$, which it was on all the processed occasions
- The standard deviation of the accelerometer data over the past 10 seconds (500 points at 10 Hz) is a reasonable indicator of turbulence levels
- The difference between the current relative humidity and the minimum value of relative humidity in the last 2 minutes (7200 points at 60 Hz) is a reasonable indicator of local data peaks

5.3 Flight Results

A total of 19 flights reached altitudes over 1900 m AMSL across Trips A – D, with 11 reaching plume altitude or above. Meteorological conditions were best in the morning with rapid deterioration limiting useful operating time and often requiring early return of the aircraft. Four key flights were chosen for analysis here, the details of which are given in Table 5.4. The plume interception points for each flight are shown in Figure 5.2, and the volcanic and environmental details of the flights are given in Table 5.5.

Table 5.4: Details of the flights used to develop the plume sensing metric.

Flight ID	Aircraft	Duration	Max Altitude	Battery Used	Avg Airspeed
A3	Zephyr II	29:45 mins	4089 m	6575 mAh (82%)	20 m/s (GPS)
B8	Skywalker X8	40:34 mins	3920 m	9652 mAh (60%)	20 m/s
B9	Skywalker X8	41:17 mins	4082 m	12458 mAh (78%)	20 m/s
D7	Skywalker X8	37:51 mins	4410 m	10880 mAh (68%)	26 m/s

Table 5.5: Volcanic and atmospheric conditions for the flights used to develop the plume sensing metric. Note that summit altitude is approximately 3800 m AMSL. 'Activity' refers to the volcanic activity type; Strombolian explosions are discrete and Degassing involves a constant release of matter. 'Frequency' refers to the frequency of eruptions/plumes.

Flight ID	Activity	Frequency	Cloud Conditions	Wind Direction
A3	Strombolian	10-15 mins	Low-level stratocumulus, plume clear	North-Easterly
B8	Degassing	Constant	Stratocumulus (up to 2727 m AMSL), plume clear	East-North-East
B9	Degassing	Constant	Clear Sky	East-North-East
D7	Strombolian	1-4 mins	Clear Sky	Southerly

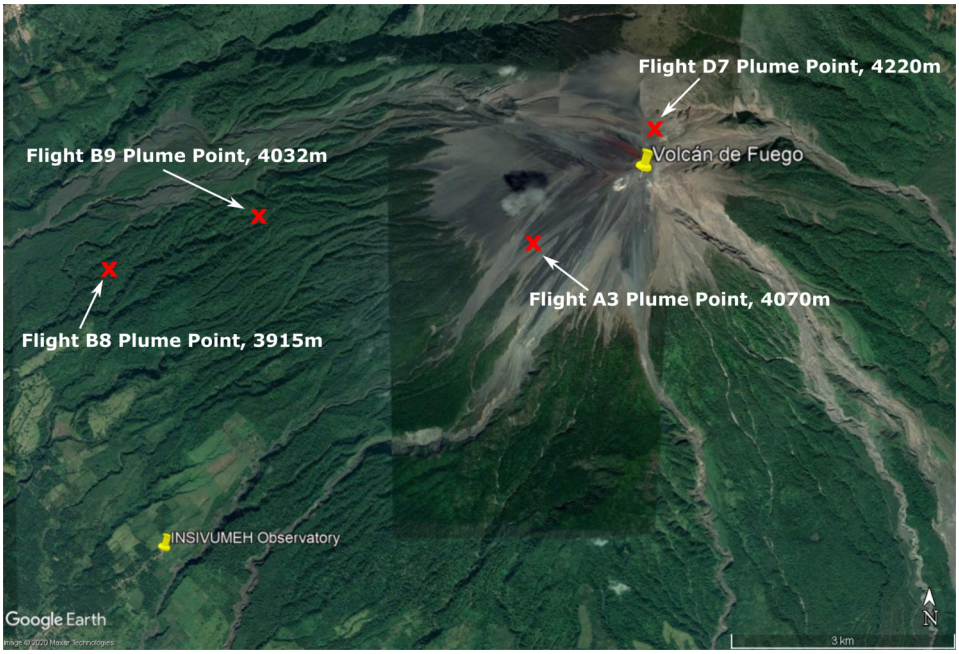

Figure 5.2: Google Earth image showing the plume interception points over the four chosen flights, relative to the observatory and summit of Fuego [10].

5.3.1 Flight A3

The initial flight plan was chosen after a series of Software-In-The-Loop (SITL) simulations. After a manual launch the aircraft entered automated flight and climbed towards the summit for a short distance. It then entered a 300 m diameter climbing spiral to 4000 m AMSL before continuing in a straight climb towards a waypoint at 4100 m AMSL, 500 m from the crater. The descent was direct from the turn point to a point near the GCS, where it then entered a controlled spiralling descent before landing manually.

Climbing in a spiral is inefficient because lift is lost proportionally to the cosine of the roll angle however this flight plan maximised time on the level approach towards the summit and

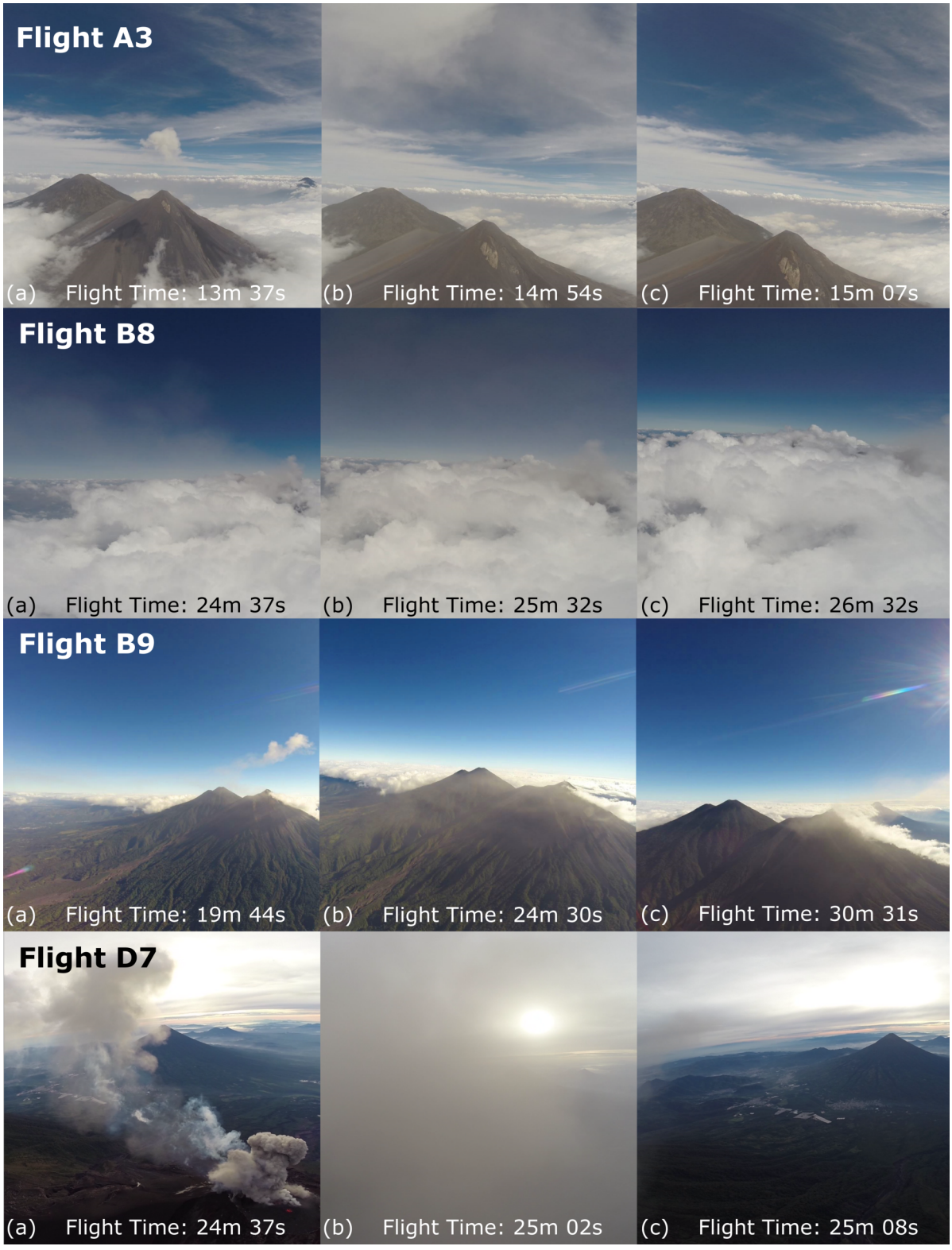

Figure 5.3: Frames from onboard video for Flight A3, B8, and D7 show (a) Before, (b) During, and (c) After plume interception. The frames for flight B9 show (a) Before, (b) Approaching, and (c) During plume interception. Note the differences in colour and change in plume location between frames.

the proximity of the climb to the GCS meant progress could easily be monitored. The volcano erupted approximately 12 minutes into the flight. The alignment of the flight path with the wind (average headwind of 7 m/s) meant the plume was intercepted during the gradual climb from 4000 m to 4100 m AMSL, approximately 1.9 km from the summit.

Figure 5.3 shows frames from onboard forward-facing video. Frame (b) for this flight shows being in, and a little below, the estimated center of the plume. Figure 5.4 shows some of the flight logs from flight A3.

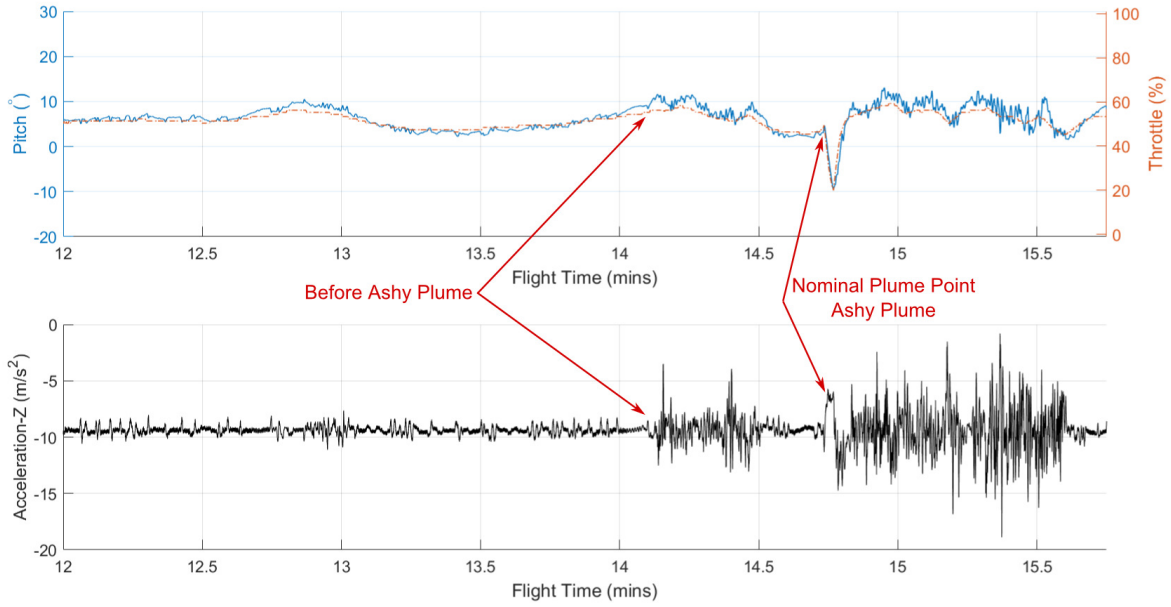

Figure 5.4: Flight logs of the Zephyr II showing flight through the Plume; Pitch (top, blue, solid), Throttle (top, orange, dot-dash), and Vertical Acceleration (bottom, black, solid). A significant increase in activity just after 14 mins of flight indicates initial plume interception.

The timings of the video agree with the logs in Figure 5.4, indicating a plume intersection at around 14 minutes and 45 seconds into the flight. It is clear from both the video and the z-axis acceleration logs that the turbulence experienced in and after the nominal plume point is significantly more than in clear air.

5.3.2 Flight B8

The data presented here are from the first of two important flights of Trip B and indicate intersection with the plume using the autopilot sensors, GoPro camera, Temp/RH mounted as shown in Figure 5.1, and ash collection. This is significant for the development of automated plume detection using UAVs.

The bearing and altitude of the plume were estimated using ground measurements, then the flight plan designed so that a large cross-wind area was covered in order to maximise the chance

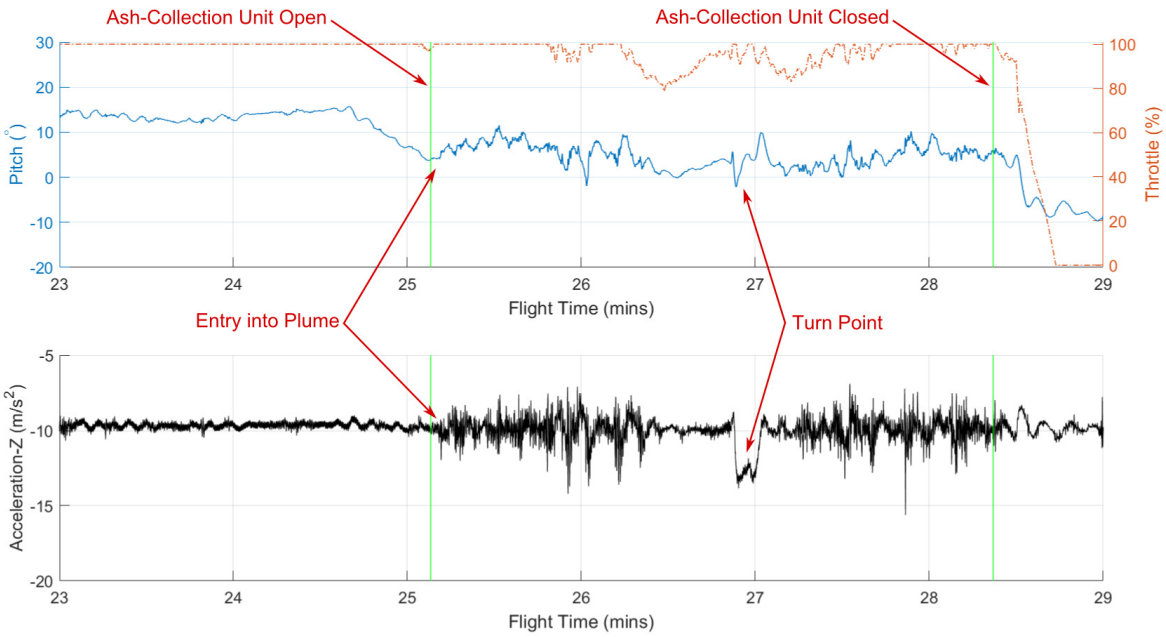

Figure 5.5: Flight logs of a Skywalker showing flight through the Plume; Pitch (top, blue, solid), Throttle (top, orange, dot-dash), Ash-collection unit open/close points (green, vertical, solid), and Vertical Acceleration (bottom, black, solid).

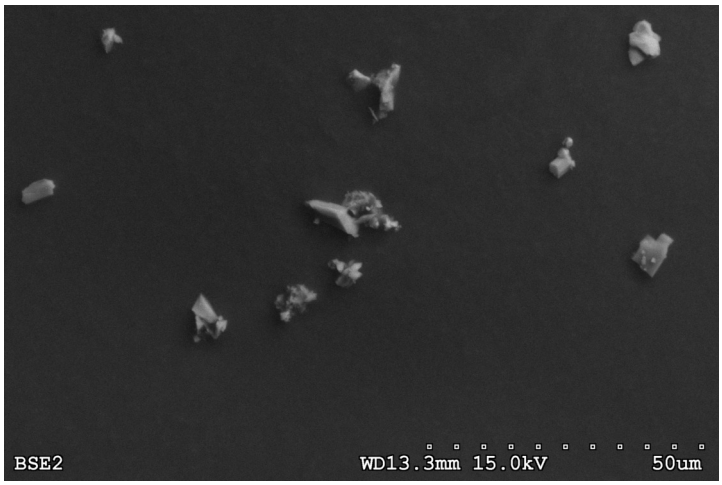

Figure 5.6: SEM stub imaging results, showing ash collected during Flight B8.

of plume interception. The aircraft climbed to altitude in a spiral before a gradual straight-line climb, then traversed across the estimated bearing of the plume (at a distal range of 5.7 km). It then turned and followed the same path in reverse to return to the GCS. This method gave the onboard camera a good view of the plume, making comparison of frames straightforward. Using the FPV video as a guide, the ash-collection unit was opened after the straight-line climb was complete and was closed at the same point upon return, immediately before the descent started.

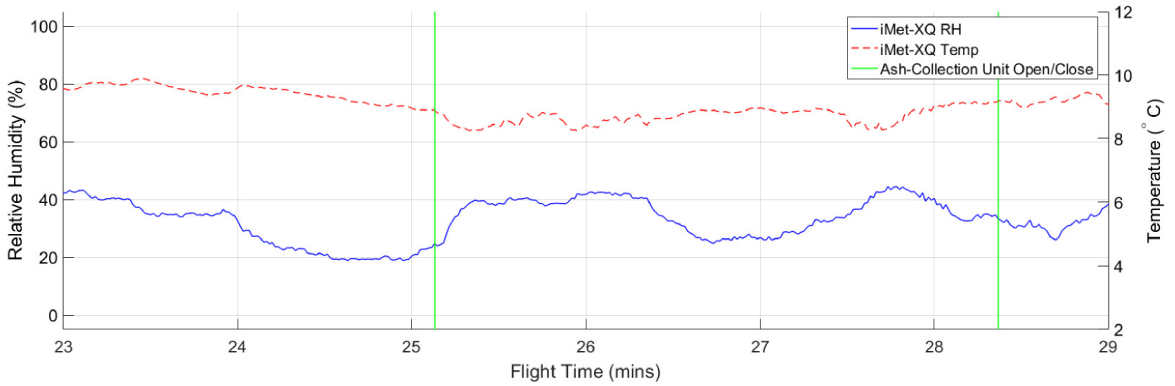

Figure 5.7: Results from the iMet-XQ sensor on Flight B8, showing temperature (red, dashed) and relative humidity (blue, solid). The vertical green lines indicate the ash collector open/close times, as in Figure 5.5.

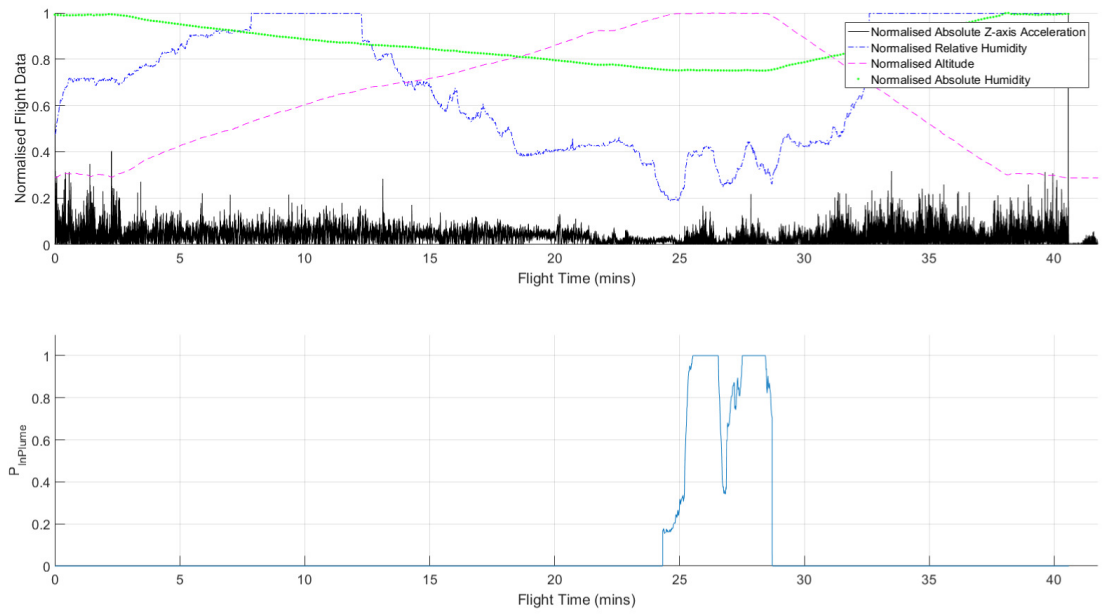

Figure 5.8: Flight performance metrics (top) and $P_{InPlume}$ (bottom), calculated using Equation 5.1. The metric aligns with increased turbulence (black), local relative humidity spikes (blue), and with the video, showing 2 plume interceptions between 25 and 28 minutes flight time.

Figure 5.3 shows three frames from the forward-facing video, corresponding to points before, during, and after plume intersection. Note the differences in colour between (a) and (b), and then again between (b) and (c). In all these frames Fuego's summit was in cloud, located near the intersection of the horizon and right-hand side of the image. Flight logs from Flight B8 are shown in Figure 5.5, where two distinct turbulent sections can be seen. These two sections correspond to the outbound and return paths of the UAV through the plume. Ash of a significant size was

collected on this flight, some of which is shown in Figure 5.6.

The Temp/RH sensor onboard for this flight was the iMet XQ, the results of which are shown in Figure 5.7. These data have been aligned with flight time using GPS time. The climb and descent both passed through cloud, saturating the humidity sensor. $P_{InPlume}$ was calculated in post-processing at time intervals of 50 Hz, the results of which are shown in Figure 5.8, below the plot of selected normalised flight data.

5.3.3 Flight B9

The sensing methods onboard this flight were the autopilot sensors, GoPro Hero 3 camera, and both Temp/RH sensors mounted as shown in Figure 5.1. The wind shear on this day was significant, with larger eruptions pushing the plume to a greater altitude where they were caught by a southerly wind. Lower-level eruptions were instead caught by a different wind layer and travelled on a bearing of approximately 260°, and it was this plume that was targeted and intercepted. According to the flight logs, the wind speed during this section of the flight was approximately 13 m/s. The throttle, pitch, and z-axis acceleration logs are shown in Figure 5.9.

Instead of transecting the plume, the flight plan met the estimated plume bearing and then turned towards the summit before returning to the GCS. The section of the flight that was on a bearing directly towards Fuego went from a distance of 5.7 km to 3.9 km from the crater, at the end of which the plume was flown through. Once again the altitude of the plume proved

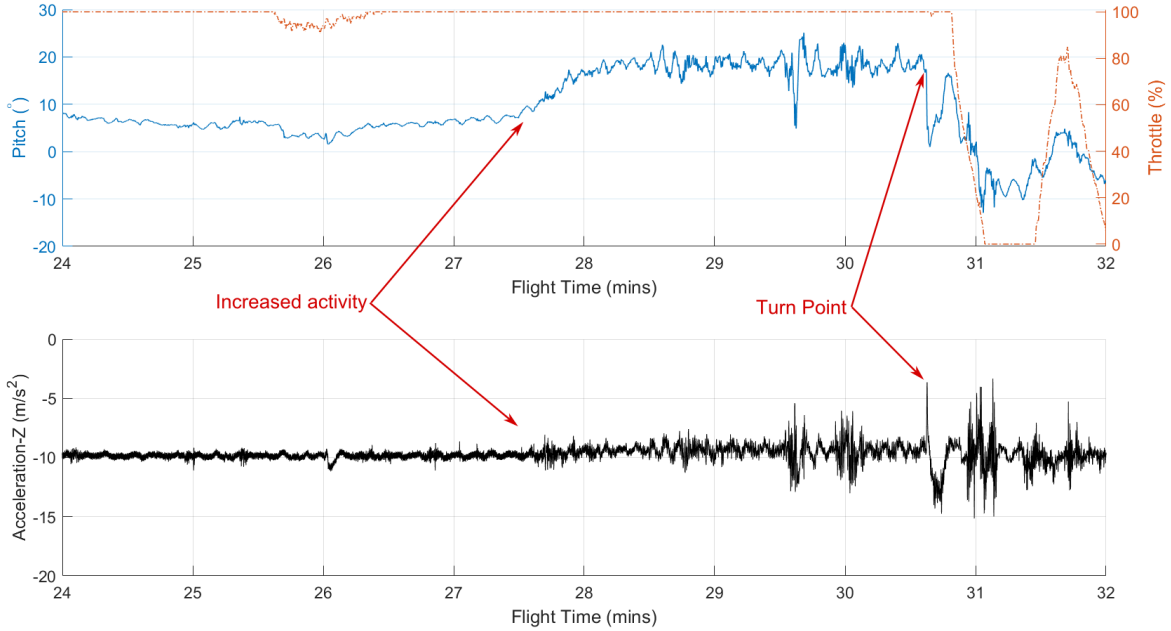

Figure 5.9: Flight logs of a Skywalker showing flight through the Plume; Pitch (top, blue, solid), Throttle (top, orange, dot-dash), and Vertical Acceleration (bottom, black, solid). The summit-approach leg started around 26 minutes, where the throttle reduces to below 100%.

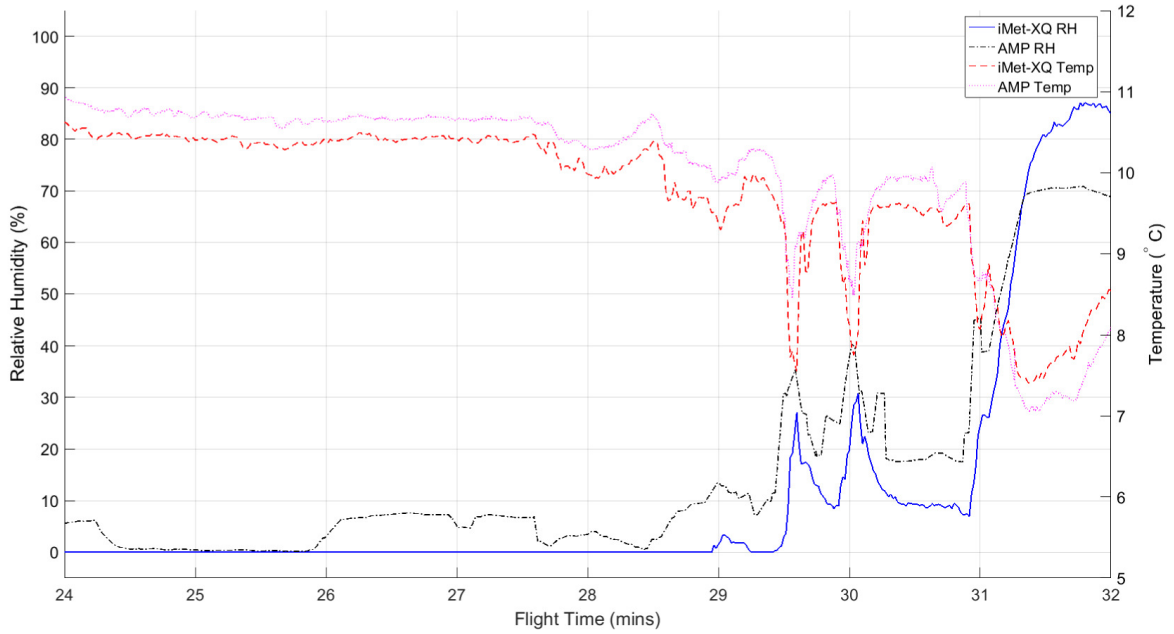

Figure 5.10: Results from both Temp/RH sensors on Flight B9. Note the drop in Temperature as the Relative Humidity peaks.

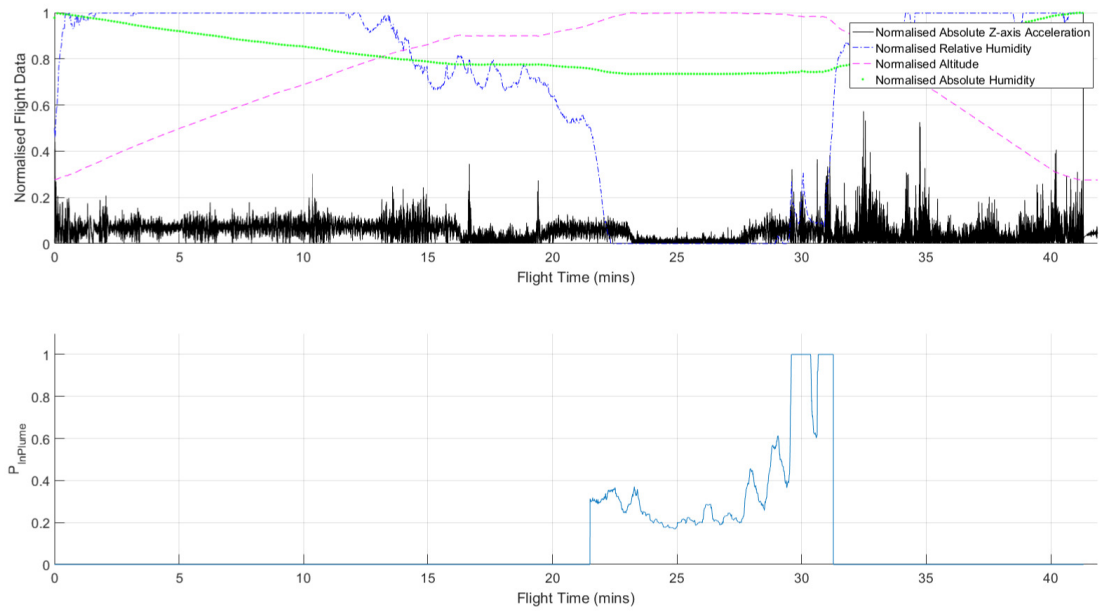

Figure 5.11: Flight performance metrics (top) and $P_{InPlume}$ (bottom).

extremely difficult to estimate. The data show that during this flight the aircraft started above the plume by some estimated 100 m and descended into it as it approached the summit, going from 4080 m AMSL to 4014 m AMSL.

Image (a) in Figure 5.3 (for this flight) shows the lower plume, parallel to the horizon and to the left of Fuego, and the higher plume visible above the summit and to the right. Image (b) was taken before the turning into the wind. The haziness between the aircraft and summits of Fuego (foreground) and Acatenango (background) is the plume. Image (c) was taken just before the aircraft turned away from the wind, and shows more haziness at the level of the aircraft.

The iMet-XQ was fitted alongside the AMP for this flight, the results of which are shown in Figure 5.10. $P_{InPlume}$ was calculated in post-processing at time intervals of 50 Hz, and is shown in the lower part of Figure 5.11.

5.3.4 Flight D7

This flight overflew the crater, loitered a short distance from the summit, and then flew through the plume before returning to the GCS (Figure 5.12). The plume intersection was flown in FBWA mode, where the autopilot maintains wings-level flight but the pilot has control of throttle, pitch, and roll (from level). The sensors onboard were the autopilot sensors, GoPro Hero 3 camera, and both Temp/RH sensors. The AMP was mounted on the wing as in Figure 5.1 but without the white protective shield, and the iMet-XQ was mounted in the nose, protruding from just above the FPV camera. These results are shown in Figure 5.14. The plume was intercepted approximately 0.5 km from the crater centre. A temperature increase of approximately 2°C was recorded on the iMet, and the relative humidity showed a spike of around 10%.

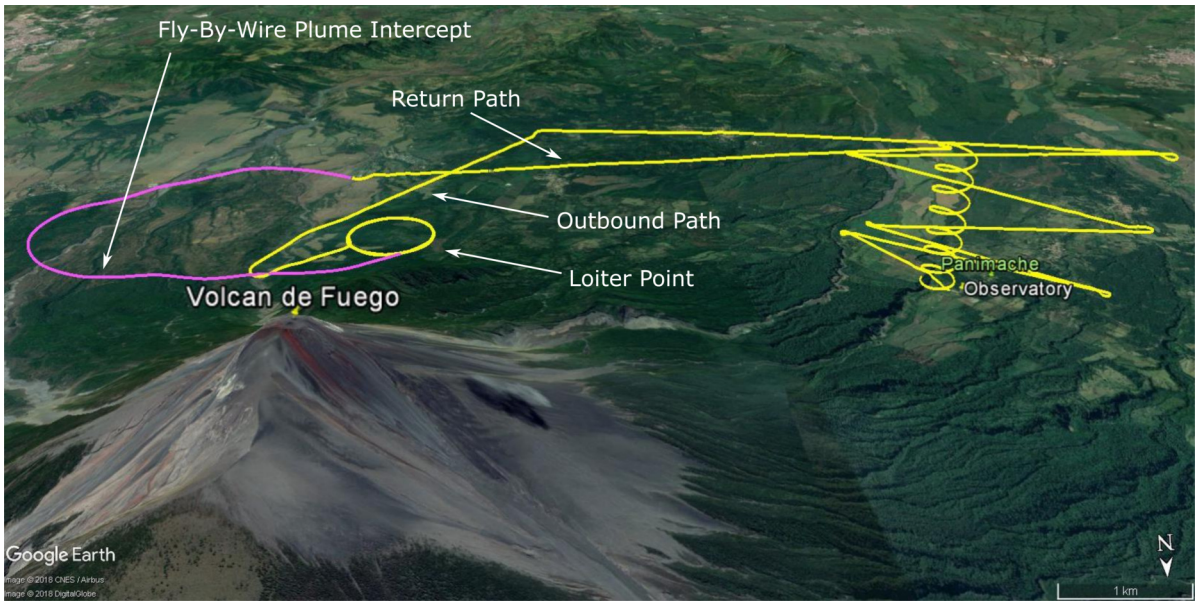

Figure 5.12: Skywalker flight-path from Flight D7 shown on Google Earth, with key tracks and locations labelled [10]. Note the Fly-By-Wire section in pink, where pilot control of the aircraft manoeuvred it through the proximal plume.

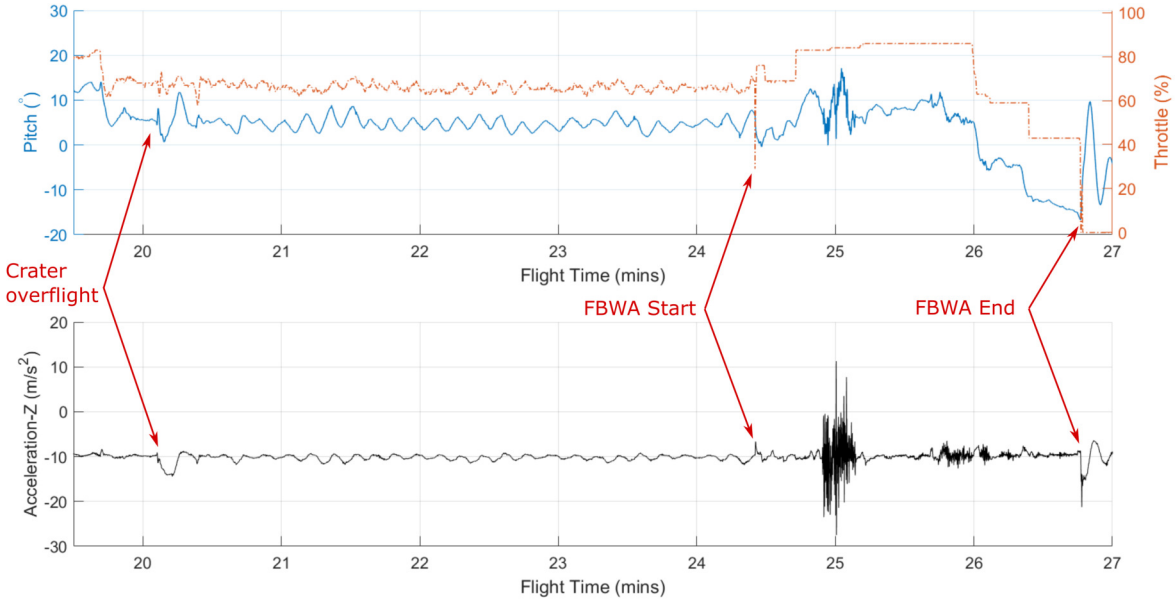

Figure 5.13: Flight logs from Flight D7 showing flight through the Plume; Pitch (top, blue, solid), Throttle (top, orange, dot-dash), and Vertical Acceleration (bottom, black, solid). The aircraft was in Fly-By-Wire mode, with manual control over the throttle.

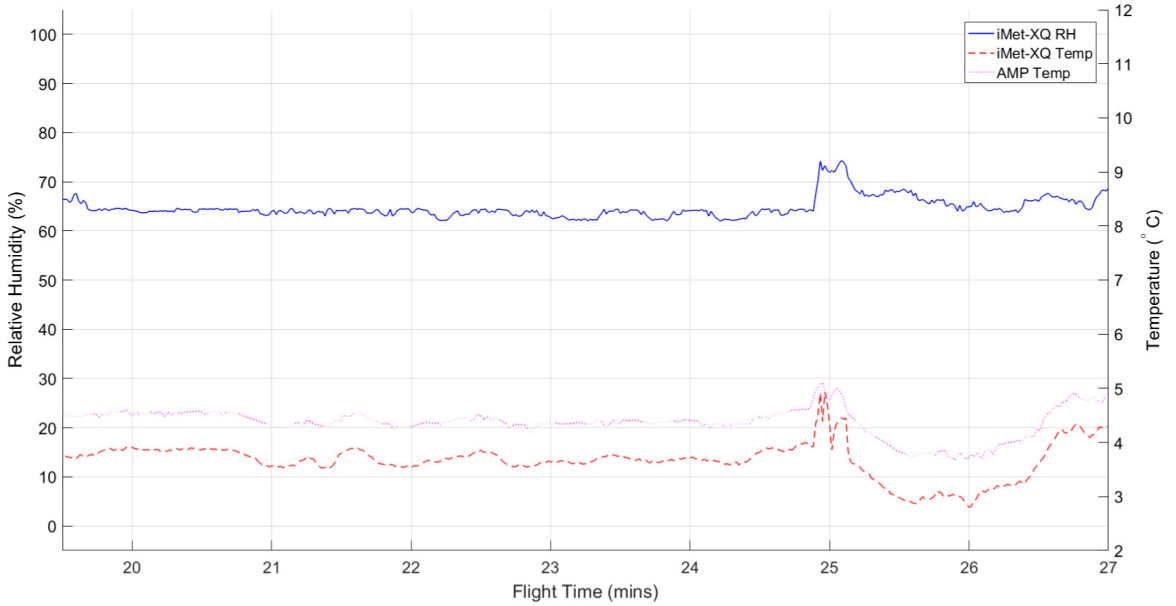

Figure 5.14: Results from both Temp/RH sensors on Flight D7. There is a peak in both Temp and RH in the plume. The AMP RH data are not presented here due to an uncalibrated sensor.

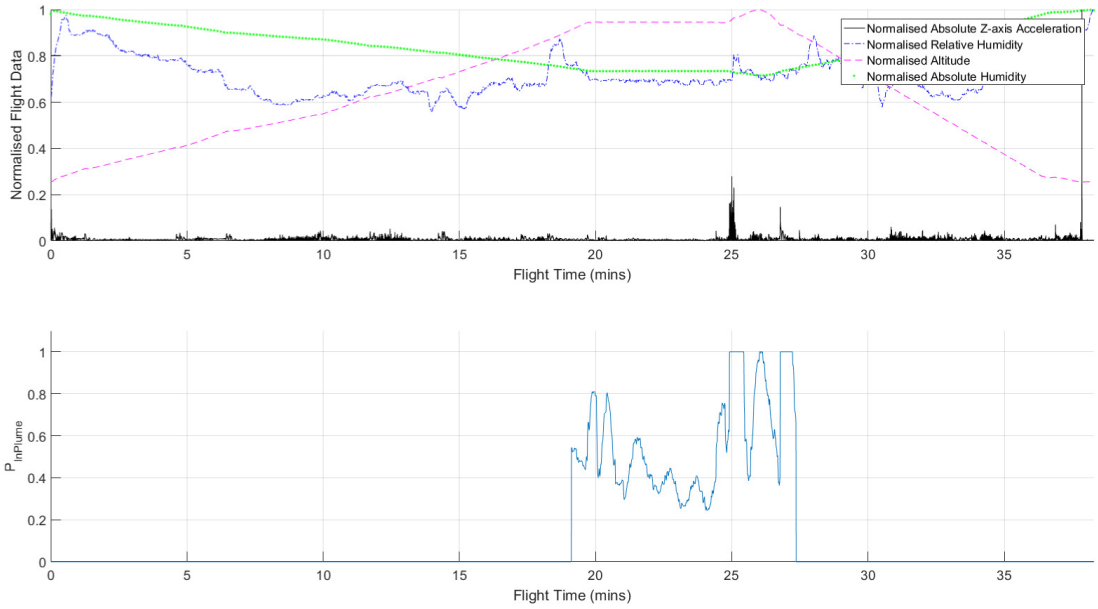

Figure 5.15: Flight performance metrics (top) and $P_{InPlume}$ (bottom).

The aircraft climbed in straight legs to mitigate the inefficiencies of the climbing spiral, then loitered near the summit until a suitable ash cloud was released. FBWA mode was then activated and the aircraft was manoeuvred through the plume with minimal change in pilot control inputs during the plume fly-through (Pink section in Figure 5.12). The aircraft was then returned to automated flight for the return to the GCS. $P_{InPlume}$ was calculated in post-processing at time intervals of 50 Hz, the results of which are shown in the lower part of Figure 5.15.

5.4 Plume Detection & Flight Analysis

This chapter considers the sensing and identification of volcanic plumes using fixed-wing UAVs, focusing on Trips A to D. A volcanic plume is the mixture of gases and ash emitted by an eruption, however for sensing purposes the boundary is hard to define, with gases diffusing at different rates and the ash as the only visible part. This section reviews the data from the onboard sensors showing that the UAVs interacted with plumes on a number of occasions, and assess the accuracy of the hypothesis.

Autopilot Pitch (°), Throttle (%), and Z-axis acceleration (m/s^2) are shown here because they can indicate a variety of different phenomena. The Total Energy Control System (TECS) flight controller works on balancing potential and kinetic energies and responding to errors in predictable ways. The responses in Table 5.6 equate to a required increase or decrease in lift, compensating for a decrease or increase in lift respectively. There are a number of parameters that can effect lift generation, the most effective being airspeed.

Table 5.6: TECS behaviour responses.

Error	Response
High Airspeed	Throttle down, Pitch up
Low Airspeed	Throttle up, Pitch down
Above Target Altitude	Throttle down, Pitch down
Below Target Altitude	Throttle up, Pitch up

Turbulent air was expected upon entering the plume, with a higher density than ambient air. The turbulence felt in the plume is caused by the circulation of hotter air rising to the top of the plume and cooling, before sinking down again. The average temperature in the plume must be equivalent to the ambient air temperature, else buoyancy would not be neutral. The center of the plume is expected to be roughly equivalent to ambient air temperature, while the bottom is expected to be warmer and the top is expected to be cooler.

An increase in the activity and magnitude of Z-axis acceleration indicates turbulence, with increased high-frequency activity in the pitch data too. A discrete increase in air density would see a response from the TECS controller equivalent to being above the target altitude. Changes in windspeed also cause a response, however they tend to be gradual and therefore cause non-discrete responses.

This section will consider each flight in turn, discussing the flight in general and, more specifically, the detection of the volcanic plume during the mission. Data collected by sensors onboard each flight will be considered and compared. It should be noted that the Skywalker X8 flown for Flight B8 and Flight B9 was fitted with the initial motor/speed-controller combination, that was not suited to the mission profile. The cruise throttle was near 100%, leaving little excess thrust for efficient climbing. The motor was replaced before Flight D7, hence the significant change in cruising throttle setting. Also, a notable increase in operational efficiency is apparent when the values in Table 5.4 are compared and considered alongside the dates of the flights.

5.4.1 Flight A3

The camera (Figure 5.2) shows that the aircraft flew through the lower section of the plume front at a flight time of approximately 14.75 minutes, and the accelerometer data (Figure 5.4) showed a corresponding increase in activity. The aircraft hit a patch of turbulent air approximately 30 seconds before reaching the ash-rich plume, likely a non-visible part of the plume that was released in degassing before the main ashy eruption. The accelerometer data indicates an upwards acceleration and a corresponding reduction of pitch and throttle at the Nominal Plume Point, which when combined suggest a sudden increase in lift due to the increased density of the plume.

A high level of turbulence continued over the following minutes of flight; the flight logs in Figure 5.4 showed relatively smooth flight before reaching the plume, however after passing

through the plume the aircraft continues to move through turbulent air for over 60 s (a distance of 1.26 km at 21 m/s). This plume 'tail', or wake, agrees with what is known about plumes through oblique photogrammetry experiments [160].

The turbulence experienced by the Zephyr II in the plume was 1.9 km from the summit and was easily distinguishable in the logs. It was decided after this flight that Temp/RH sensors would be beneficial for further identification of the plume, only adding a small mass to the system. Real-time processing of this turbulence data can be moved onboard, however alone it will not suffice as a method for sensing the plume because turbulence can also be encountered in non-plume environments such as clouds.

5.4.2 Flight B8

Key to this flight was the evidence collected by the ash-collection unit. The altitude and FPV video were used as indicators for when to open the unit, and it was closed at the same point on the return journey. These points are indicated in Figure 5.5 by the vertical solid green lines. The particulates on the SEM stubs were analysed in a laboratory and there was a significant and confirmed presence of ash on both stubs, some of which is shown in Figure 5.6. The volcanological implications of these collections are discussed in part in an upcoming publication by Lucas and Watson *et al.*, from the University of Bristol [Citation Pending].

Regarding the flight log plots in Figure 5.5, there are two distinct periods of increased turbulence during the period of interest. The first corresponds to the initial pass through the plume, and the second corresponds to the return path, after the turn. Peak turbulence was encountered centrally within these sections.

From around 25.75 minutes flight time the throttle starts to fluctuate, which along with a reduced pitch angle suggests excess lift is being generated. Again, this is a suspected result of an increased in-plume density.

Figure 5.7 shows no significant change in temperature during the plume interception, suggesting that the UAV flew relatively close to the center. The relative humidity is approximately 20% either side of the plume, but goes to approximately 40% on each of the plume interceptions. The humidity either side of the plume is the lowest recorded on this flight. The climb and descent on this flight were through cloud, saturating the RH sensor. The UAV was estimated to have climbed out of the atmospheric boundary layer at 24 minutes flight time, approximately 200 m below the altitude of the plume. As shown in Figure 5.7 the boundary layer was at similar RH levels to the plume, which means should the plume have been encountered below the boundary layer then it may have been indistinguishable from ambient air using RH data alone.

5.4.3 Flight B9

In this case, the aircraft started the approach leg above the plume which meant the background features in the video were ground and volcano, not clear sky. Considering the camera alone, it is

not clear that the plume was flown through, however the other sensors used indicated a plume interception towards the end of the summit-approach leg.

Pitch and Z-axis acceleration data in the flight logs (Figure 5.9) suggest an increasing amount of turbulence from approximately 28 minutes flight time until the turn point. At 26 minutes flight time the pitch is at a near-zero value, and the throttle setting was approximately 90%, indicating that the aircraft reached the required altitude at the turn onto summit-approach. The throttle remains at 100% to maintain airspeed, and then the pitch increased to approximately 20 °. The aircraft is attempting to climb over this period (Table 5.6), however the logs show it actually descending until it reaches the turn point.

Temp/RH data (Figure 5.7) show some differences between the iMet and the AMP. Beyond the boundaries of this plot the iMet RH saturates in the cloud on the ascent and descent, and registers 0% during the summit approach leg, the AMP peaks at approximately 85% and is mostly non-zero during the approach leg. One explanation for the lack of saturation on the AMP is the large protective shield fitted for protection of the delicate sensors, as seen in Figure 5.1. There are two peaks in RH, at 29.6 mins and 30.1 mins, which correspond to temperature drops of 2 °C and increased Z-axis acceleration, indicating UAV/Plume interaction. Relative humidity registered at near 0% before the section of turbulence, and slowly increases up to 29.5 mins. At the plume points the RH is approximately 30%, and drops down to 10% again before the aircraft turns at the end of the leg. RH for Flight B9 was similar to Flight B8, at a point of turbulence that was similar both in relative increase and magnitude. The temperature drops by up to 2 °C in the plume, explained by the flight path through the top of the plume where the UAV encountered cooling air.

Combining visual evidence, the gradual increase in acceleration up to the two plume points, and the 20% increase in RH, it seems that the plume was descended into on this leg. It was flown through at an altitude of 4014 m AMSL for approximately 1 minute before the aircraft turned at the end of the leg. That the aircraft descended while trying to climb indicates two things; firstly that the original motor type on the Skywalker was severely limiting in these conditions, and secondly that the air above the plume drew the aircraft down. Effects such as the circulation of matter in the plume could explain this uncommanded movement.

5.4.4 Flight D7

This mission varies from the others presented here most notably because there was a section of piloted flight, albeit with augmentation from the autopilot. Figure 5.3 clearly shows that the plume was flown through. Previous flights intercepted the plume at least 1.8 km from the crater, however this one was approximately 0.6 km from the crater. Prior to the plume interception the UAV overflew the crater and loitered for 4 minutes.

Figure 5.13 shows data from the initial summit overflight until after the piloted section of flight (see Figure 5.12). Note the magnitude of the acceleration data in Figure 5.13; the vertical

accelerations recorded were up to four times greater than anything previously flown through. This can be attributed to the state of the plume at this distance from the summit in these conditions, as it is still visually turbulent and not yet at neutral buoyancy (i.e. it is still rising, albeit at a slower rate than after initial expulsion). The pitch data also indicates turbulence at the plume point, however little else can be gleaned from the pitch or throttle data for this flight due to the aircraft mode. Fully automated flight would have enabled a more thorough analysis of the plume from an aircraft control perspective.

The non-neutral buoyancy observed suggests that there would be an increase in temperature during the fly-through; Figure 5.14 shows a relatively consistent temperature of 3.7 °C during loitering flight and an increase of approximately 1.1 °C during plume fly-through. The AMP temperature registered approximately 0.9 °C higher than the iMet, and only peaked by around 0.5 °C in the plume. These differences could be explained by calibration errors, however the sensors agree that in this instance the in-plume temperature is higher than ambient temperature; approximately 5 °C.

RH data showed similar responses to the temperature, with a consistent 64% during the loiter phase and an increase to 74% in the plume. The iMet has previously shown a 20% RH increase in the plume, but here it is only 10%. One explanation for this is the mounting location of the sensor, as for this flight it was located in the nose rather than on the outside of the fuselage. The iMet data responds with acceptable levels of delay when compared to the flight data turbulence, however further testing of response times at flight speed is needed to confirm the exact point at which RH increases relative to the Z-axis acceleration.

5.4.5 In-Plume Detection Metric

As shown in Figures 5.8, 5.11, and 5.15, $P_{InPlume}$ generates a value that peaks at the identified plume points of the flights. In addition to the previously identified plume points there are some additional unity values shown, such as in Figure 5.15 where there are peaks at around 27 minutes. The metric presented in this paper is a good indicator of UAV/plume interaction, and should be used in real-time for operational decision making when targeting plume fly-throughs using automated systems. Further testing could include the inclusion of CO_2 or SO_2 sensors, which should sense changes in the plume compared to in ambient air. These could be included provided they are capable of sensing the expected change, are suitably light, and have low power requirements.

5.5 Conclusions & Further Work

This chapter has shown that fixed-wing UAVs can be used to collect small ash samples from the plume of Volcán de Fugeo in Guatemala, and that it is possible to automate the detection of volcanic plumes. Flight B8 proved the concept of airborne ash collection using SEM stubs. With

development of a second generation stub deployment mechanism, it is possible to collect multiple airborne samples from a single flight and do so in a repeatable manner. However, further work is required to automate this process for reliability and developed autonomy. Regardless of the level of autonomy involved in collection, these ash-sample data shall be used to better model the effects of volcanic ash on aircraft and be input into active aviation management tools.

Vertical acceleration of the aircraft has been combined with altitude, pressure data, and humidity data to identify when the UAV is in a plume. These data have then been combined to form $P_{InPlume}$, which reached a value of 1 when the aircraft was in a plume over the three sensor-equipped flights presented here. Additional elements such as gas sensors could be added to the calculation of $P_{InPlume}$, provided they add to the robustness of the metric and are suitably small and lightweight. The following chapter addresses the real-time trajectory planning of UAVs, including flight testing around Volcán de Fuego.

CHAPTER 6

REAL-TIME TRAJECTORY PLANNING

The generation of a mission plan that is independent of human input is desirable, providing reliable and consistent routes focused on efficient performance and reducing the likelihood of human error. A real-time trajectory planner might take into account obstacles, moving or stationary, as well as phenomena such as turbulence. The ability to re-plan while in flight should increase mission efficiency and reduce the overall number of aborted missions. In this chapter the development of a Real-Time Trajectory Planner (RTTP) is described, and real-world testing presented. Visual Line of Sight (VLOS) tests were conducted at a UK location before full Beyond Visual Line of Sight (BVLOS) tests around Volcań de Fuego in Guatemala (Figure 6.1).

The challenge of navigating robotic systems through confined spaces means many path planning algorithms consider cluttered environments [97]. Fixed-wing path planning research often focuses on these types of problems, especially concerning route continuity, for VLOS-type operations [118]. Long-range path planning problems are considered by Roberge *et al.* and Nikolos *et al.*, whose problem spaces are filled with terrain and obstacle restrictions to mimic semi-cluttered environments [108, 109]. The environment around Fuego differs from these examples due to the relative sparsity of the problem space, and the significant altitude gain required for the mission. Furthermore, the RTTP developed here is the first of its kind to be flight tested in BVLOS experiments.

Before an RTTP can be flown in the real world with reasonable safety a number of things must first happen. An appropriate algorithm must be chosen, and developed in a manner suitable for route planning. This code must then work on an autopilot companion computer, with appropriate tests to show both hardware and software functionality. The companion computer must then be integrated with the existing UAS, followed by ground and flight testing. A ground station program must also be written so that updates can be received and commands sent when necessary.

Figure 6.1: Sunrise over Volcań de Fuego, from the Ground Control Station in Panimache.

This chapter presents the development of a real-world RTTP system, initially focusing on the version that was flight tested in Trip E. These flight test results are presented and discussed, and then further refinement of the system methodology is given in Section 6.6. Flight testing of the latest RTTP version was due to take place in Guatemala in March 2020, however the COVID-19 pandemic effectively cancelled this trip. The results presented in Section 6.7 are therefore limited to simulation only.

This chapter is based in part on a journal paper titled 'Onboard Real-Time Trajectory Planning for Fixed-Wing Unmanned Aerial Vehicles in Extreme Environments', published by the Author in the MDPI journal Sensors and concerning the RTTP version that was flight tested on Trip E [29].

6.1 Introduction

UAV missions are typically flown according to a predetermined waypoint plan, often resulting in wasted energy and reduced amounts of time available at the target location. Unexpected conditions are usually handled by aborting the flight and refining the mission for the next flight. A reliable method of reaching the target location, while minimising energy use and taking into account the current atmospheric conditions, would increase the time spent collecting useful scientific data. An onboard RTTP allows mission plans to be updated 'on the fly' without the need for a robust high-bandwidth link to the ground station, and future implementations may include sense-and-avoid technology and aircraft performance optimisation. A full solution would detect changes in aircraft performance, the environment, and air traffic, and generate new mission plans to minimise unnecessary energy wastage while avoiding aborting the flight. It is common,

upon arriving at a new operating site, for a significant amount of time to be spent developing suitable and efficient 'conventional' flight plans. With a full-solution RTTP in place, the time taken from arrival at a new site to flying an efficient and successful mission would be reduced significantly.

While numerous motion and path planning algorithms and optimisation methods exist, few have been considered with 3D UAV implementations in mind. There are no published cases of real-time trajectory optimisation taking place onboard a UAV in flight. Optimisation methods typically consist of an algorithm to minimise some 'cost function'; these can have single or multiple objectives and in the context of UAV motion planning they may, for example, include a function of path length and proximity to terrain. Certain limits can also be imposed which form 'hard' constraints on the problem, for example positive separation from an obstacle (i.e. never impacting it), or the continuity of the path being maintained (to ensure it is flyable). The cost function implemented for the work in this paper will be discussed in Section 6.2.3.

As introduced in Chapter 4, NATO defines levels of autonomy for UAVs with the following definitions [157]:

- **Level 1, Remotely Controlled System:** System reactions and behaviour depend on operator input.
- **Level 2, Automated System:** Reactions and behaviour depend on fixed built-in functionality.
- **Level 3, Autonomous Non-Learning System:** Behaviour depends on fixed built-in functionality or upon a fixed set of rules that dictate system behaviour (goal-directed reaction and behaviour).
- **Level 4, Autonomous learning system with the ability to modify rules defining behaviours:** Behaviour depends upon a set of rules that can be modified for continuously improving goal directed reactions and behaviours within an overarching set of inviolate rules/behaviours.

In the context of developing an RTTP, a UAV following a set mission plan can be classified as Level 2 or Level 3. However, generating a mission plan onboard and in real-time could be classified as Level 3 or Level 4. The benefits of real-time flight path generation include dynamic obstacle avoidance and aircraft performance optimisation. In cases where tracking or searching flight is required, the aircraft could adjust its flight pattern in real-time according to the results of the search. Letheren *et al.* have flown multi-rotor UAVs with a route generated in real-time by a bio-inspired algorithm intended to find the source of a forest fire [84]. However, this was limited to 2D VLOS, with the computation of the path carried out on the GCS. Where UAVs are required to operate over longer ranges, extending to BVLOS operations [161], the telemetry link to the aircraft typically has lower bandwidth than VLOS operations. This has implications for the

feasibility of performing the necessary computation on the GCS, providing motivation to move the planning computation onboard the vehicle.

This chapter presents a functioning real-time trajectory planner, run onboard the aircraft, which has been flown in the real world (Trip E). While the vehicle's progress was carefully monitored, it acted entirely under the control of the RTTP, therefore effectively increasing the level of autonomy during testing.

6.2 Methodology

6.2.1 Algorithm Selection

A number of algorithms were explored for implementation onboard UAVs as an RTTP, as discussed in Chapter 2. Based on the literature, the Genetic Algorithm (GA) and Simulated Annealing (SA) methods were selected for initial testing. While both worked well in 2D, the GA proved superior to the SA in early 3D testing. Considering the work of Nikolos *et al.* and Roberge *et al.*, the GA was chosen for testing in the real-world case and is presented in this chapter [108, 109]. Further improvements were made to the algorithm following Trip E, and are presented in Sections 6.6 and 6.7.

The objective of the path planning problem around Fuego is to reliably and robustly plan routes from the INSIVUMEH Observatory (1137 m AMSL) to Fuego's summit area (approximately 4000 m AMSL), with efficiency metrics that are at least comparable to manually developed missions and convergence times that are practical for real-world use. Some of the (early) key problem constraints are listed below:

- Waypoint locations are restricted to within a 12 km radius of Fuego's summit and must not require travel behind the volcano as seen from the GCS location
- Route must not pass within 200 m of the ground, with the exception of the start location
- Tuned for efficient routes over a height gain of approximately 3000 m
- Must avoid obstacles; initially these were placed at arbitrary positions however they will be used to represent real 'No Go' zones in the future
- Must prioritise efficiency, preserving battery capacity for use at and around Fuego's summit

6.2.2 World Space and Trajectory Representation

One of the most fundamental aspects of trajectory planning, particularly for real-world systems, is the incorporation of the environment around the vehicle. Representation of the world space includes terrain, start and end locations, and 'No Go' zones. NASA's Shuttle Radar Topography

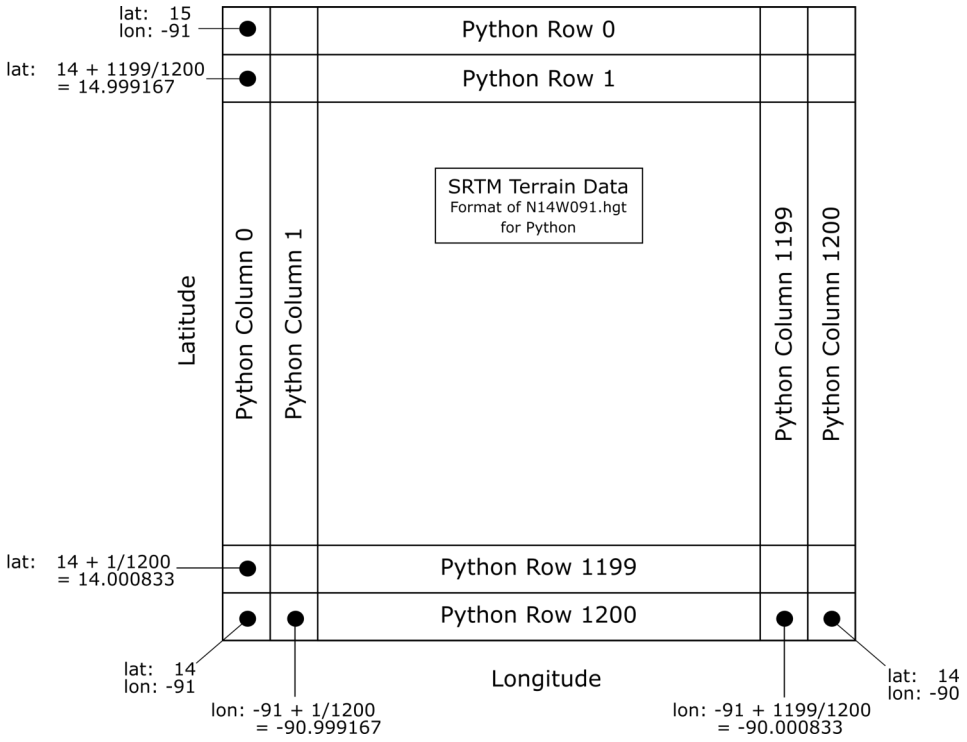

Figure 6.2: Format of the SRTM terrain file used for Guatemalan operations, including Python indexing points.

Mission (SRTM) terrain data was downloaded at 3 arc-second resolution for Guatemalan operations [162], the format of which is described in Figure 6.2 for use in Python. For this position on the globe there is an approximate distance of 90 m between data points, with each point representing an average altitude AMSL of the local and neighbouring 1 arc-second data (30 m resolution). For UAV operations in an environment such as this, where proximity to the ground is generally avoided, an additional safety buffer can be added to the altitude data (200 m in this case), making the resolution sufficient for terrain avoidance.

Point locations are described in the following format:

$$Location = \begin{bmatrix} lat & lon & h \end{bmatrix} \tag{6.1}$$

where h represents the altitude of locations in meters AMSL, referenced to the WGS84 SRTM model. Areas to be avoided were described as 'No Go' zones or 'Obstacles', limited initially to cylindrical shapes of infinite height. Future work involves combining cylinders to form complex shapes, with the addition of altitude limits. The format of these zones follows:

$$NoGo = \begin{bmatrix} lat_1 & lon_1 & r_1 \\ lat_2 & lon_2 & r_2 \\ \dots & \dots & \dots \\ lat_n & lon_n & r_n \end{bmatrix} \tag{6.2}$$

where r_n represents the radius of zone n in meters, centred on lat_n, lon_n (decimal degrees). The trajectory is described as a series of waypoints in the format given in Equation 6.3, with each waypoint represented by a latitude, longitude, and altitude (metres AMSL):

$$Trajectory = \begin{bmatrix} lat_1 & lon_1 & h_1 \\ lat_2 & lon_2 & h_2 \\ \dots & \dots & \dots \\ lat_n & lon_n & h_n \end{bmatrix} \tag{6.3}$$

where h_n represents the altitude of waypoint n in metres AMSL. This form of parametrisation is well suited to large scale problems, and integrates well with the modular planned hardware configuration. A waypoint-style mission can be commanded and monitored from a companion computer and, unlike other methods such as B-spline curve following, requires no modification of the autopilot's navigational system. This builds a level of safety into the practical use case, allowing the companion computer and RTTP to simply be ignored or isolated if the system behaviour is not as expected.

6.2.3 Cost Function

The structure of the cost function was partly inspired by the method of Roberge *et al.*, with elements split into *feasibility criteria* and *optimisation criteria* [108]. The former must be satisfied for a route to be possible and the latter define the characteristics of the route. The general equation is given in Equation 6.4:

$$Cost = \underbrace{C_{Terrain} + C_{Power}}_{\text{Feasibility Criteria}} + \underbrace{C_{Length} + C_{Altitude} + C_{NoGo} + C_{Climb}}_{\text{Optimisation Criteria}} \tag{6.4}$$

Each of the terms given in Equation 6.4 is described below, starting with the term associated with the length of the path:

$$C_{Length} = 1 - \frac{L_{AB}}{L_{traj}} \tag{6.5}$$

therefore

$$C_{Length} \in [0, 1] \tag{6.6}$$

where L_{AB} is the straight line distance from A to B, the start and end points of the route, and L_{traj} is the distance of the route in question. The purpose of this criterion is to minimise the length of the trajectory. This criterion tends to zero as the trajectory length decreases.

Keeping the average altitude low helps reduce unnecessary climbing and therefore unnecessary energy expenditure. This is affected by $C_{Altitude}$:

$$C_{Altitude} = \frac{h_{traj} - h_{ABmin}}{T_{max} - h_{ABmin}} \tag{6.7}$$

set with limits of

$$C_{Altitude} \in [0,1] \tag{6.8}$$

where h_{traj} is the average altitude of the waypoints in the trajectory, T_{max} is the maximum altitude of the terrain data given in the SRTM grid, and $h_{ABmin} = min[h_A, h_B]$, the altitude of the lower of the start and end points. This criterion tends to 1 as average trajectory height tends to T_{max}.

While not permitted to enter 'No Go' zones, flight through them in this case is physically possible and therefore they are included in the cost function as an optimisation criterion, rather than feasibility criteria:

$$C_{NoGo} = \frac{L_{inNoGo}}{\sum_{i=1}^{n} 2 \times rad_i} \tag{6.9}$$

set with limits of

$$C_{NoGo} \in [0,1] \tag{6.10}$$

where L_{inNoGo} is the length of the route inside 'No Go' zones, n is the number of obstacles, and rad_i is the radius of the 'No Go' zone i.

In addition to minimising the average altitude of the route, the cost function penalises over-climbing. C_{Climb} aims to minimise the altitude gain of the trajectory, avoiding over-climbing:

$$C_{Climb} = 1 - \frac{h_B - h_A}{\sum_{i=1}^{n-1} d_i} \tag{6.11}$$

where

$$d_i = \begin{cases} h_{i+1} - h_i, & (h_{i+1} - h_i) > 0 \\ 0, & (h_{i+1} - h_i) <= 0 \end{cases} \tag{6.12}$$

and therefore

$$C_{Climb} \in [0,1] \tag{6.13}$$

where h_A is the altitude of the start location altitude, and h_B is the altitude of the end location.

Two feasibility criteria are used; $C_{Terrain}$ penalises trajectories that pass underground, and C_{Power} penalises trajectories with unrealistic climb angles. These components differ in value to the cost function optimisation criteria by the addition of P when trajectories are not feasible, where P is the number of optimisation criteria. In this case, $P = 4$:

$$C_{Terrain} = \begin{cases} P + \frac{L_{under}}{L_{traj}}, & L_{under} > 0 \\ 0, & L_{under} = 0 \end{cases} \tag{6.14}$$

so

$$C_{Terrain} \in 0 \cup [P, P+1] \tag{6.15}$$

where L_{under} is the length of the route that is underground.

A similar approach was taken to penalising high climb angles, with the length of legs over a given angle summed and if not zero then included:

$$C_{Power} = \begin{cases} P + \frac{L_{overMax}}{L_{traj}}, & L_{overMax} > 0 \\ 0, & L_{overMax} = 0 \end{cases} \tag{6.16}$$

so

$$C_{Power} \in 0 \cup [P, P+1] \tag{6.17}$$

where $L_{overMax}$ is the cumulative length of the legs in the trajectory that exceed a specified climb angle. This value would vary with aircraft type and atmospheric conditions, but was kept constant at 10° for the Skywalkers in this work.

6.2.4 Genetic Algorithm

This section describes the implementation of the GA used for the flight testing work on Trip E. Later developments and refinements are discussed in Section 6.6. The pseudocode for the GA is given in Algorithm 3.

An initial population of waypoints is generated with a Gaussian distribution from the midpoint between the start and end locations, forming a string of $[lat, lon, alt]$ coordinates. A chromosome is formed by arranging the waypoints as shown in Equation 6.3 and evaluating the cost of that trajectory. Each chromosome within the generation is manipulated with crossover or mutation operators, discussed further below. Only the operations that have a positive effect on the fitness of the chromosome are accepted as children, and in this early version they are replaced at the end of the generation. Once the entire generation has been operated on, it is sorted according to the fitness of the children and the lowest quarter is replaced. These new genomes are generated

using a crossover between the top quarter of solutions and randomly generated new routes, and are made chromosomes by evaluating their cost functions. This was found to balance exploration and exploitation of the design space through the optimisation process.

Algorithm 3: Genetic Algorithm pseudocode

```
Result: bestChromosome = generation[0].cost
generate initial population;
i = 1;
while best cost > threshold cost do
    chromosome ← generation[i];
    child ← crossover(chromosome);
    if child.cost < chromosome.cost then
        chromosome ← child;
    end
    child ← mutation(chromosome);
    if child.cost < chromosome.cost then
        chromosome ← child;
    end
    replace chromosome in generation;
    i = i + 1;
    if i = length(generation) then
        i = i - length(generation);
        sort generation by cost;
        worst quarter of the population ←
          random crossovers from top quarter of population;
        sort generation by cost;
    end
end
```

The purpose of this code is to run in real-time, therefore hyper-parameters such as threshold cost and generation size were tuned for the real-world use case. Software-in-the-Loop (SITL) simulations meant tests could be conducted in a safe and repeatable manner, and a threshold cost of 1.5 was found to be suitable for initial flight testing.

6.2.4.1 Chromosome Generation

New trajectories are generated by creating between two and eight intermediary waypoints, joining the start and end locations. For a given trajectory, an iterative waypoint generation process is used where the waypoints have a Gaussian distribution centred on the halfway point between start and end locations, with a standard deviation of $\frac{L_{AB}}{2}$. The altitude is assigned by adding or subtracting an appropriate amount from the altitude of the preceding waypoint, using a flight path angle that is uniformly randomly selected from a prescribed aircraft-specific range.

Large population sizes (>> 10) were found to weight exploration over exploitation, generating

too many new chromosomes and not operating on them enough, given the real-world time constraints. Exploitation was weighted over exploration with population sizes <<10, leading to local minima and sub-optimal results. A population size of 10 was therefore nominally chosen for early development and initial testing, with further analysis taking place during the refinement process.

Section 6.6 presents further development of this code, which led to significant speed-ups. This included testing to evaluate the best threshold cost, population size, and generation replacement method, using the effect on run time and route quality as metrics.

6.2.4.2 Crossover Operation

The crossover operation used in this version of the GA requires two parent routes [108]. A cut point for each parent was randomly chosen and the routes added together. The altitude of the waypoints remained consistent throughout, making the operation similar to a 2D trajectory or path crossover function (Figure 6.3).

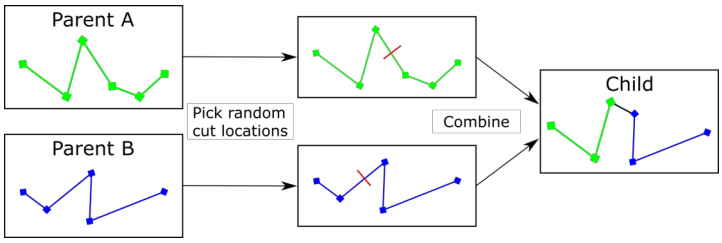

Figure 6.3: The function of a GA crossover operator in a 2D path optimisation example.

6.2.4.3 Mutation Operation

One of three types of mutation was randomly chosen to be attempted: waypoint (WP) addition, deletion, and modification (Figure 6.4). The first two involve the addition or deletion of a WP at a random point between the start and end locations. The modification operator involves effectively combining the two, first selecting a random WP then assigning it a new set of coordinates and altitude using the Gaussian waypoint generator mentioned above.

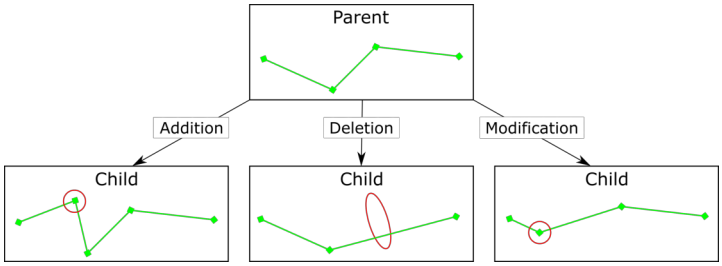

Figure 6.4: Possible functions of a GA mutation operator in a 2D path optimisation example.

6.3 Implementation

The real-world implementation of the RTTP can be split into three main sections: hardware, mission control, and communications; the hardware used is discussed in Chapter 3, and the mission control and communications structures used for Trip E are presented in the following section. Further development of these programs is presented in Section 6.6. Unless otherwise stated, all code was written in Python 2.7.

6.3.1 Mission Control

The mission control program is responsible for running the main loop, as described in Algorithm 4. Once power is turned on and all systems are initiated, the algorithm waits for the aircraft mode to be turned to GUIDED. This is a positive action from the pilot, and control can always be regained by the pilot switching out of GUIDED mode. Conventional operations are flown in AUTO mode, following a pre-set mission plan, and FBWA mode has also been used for pilot-guided first person view (FPV) flight for accurate plume interception.

In addition to the main loop, the early version of the mission control script runs two threads. The first is responsible for handling MAVLink messages, a standard message protocol used by the ArduPilot software, and the second is responsible for running the optimisation algorithm when required. The version of the RTTP that was flown on Trip E generated just one route per flight, as a proof of concept. The generation of new routes during the flight, in order to accommodate changes in conditions, is part of the further development conducted after Trip E (Section 6.6).

6.3.2 Ground Station Update & Command

It is vital to have a line of communication between the aircraft and the ground control station (GCS), where the operators make top-level decisions about the aircraft, mission progress, and RTTP algorithm. There are two key components to this; updates from the aircraft regarding the progress of the RTTP, and sending commands from the GCS to the RTTP. The progress of the aircraft is monitored from an Update & Command program, written using the Python module TKinter (Figure 6.5), with the pilot taking control when appropriate depending on the mission progress and payload requirements.

MAVLink, the existing ArduPlane communications protocol, is used to transmit custom messages between the UAV Mission Control program to the GCS Update & Command program. Both programs establish a link to the MAVLink network using unique component IDs, making it possible to target specific programs or components with messages. While it is possible to create a new message type and integrate it into the protocol, the quantity and regularity of the custom information required was insufficient to justify undertaking this complex task. Instead, existing message types such as ‘`STATUSTEXT`’, ‘`DEBUG_VECT`’, and ‘`NAMED_VALUE_INT`’ were used, loaded with custom information [163].

Algorithm 4: RTTP Mission Control Algorithm

```
Result: Control of RTTP Mission
wait for heartbeat from Autopilot;
wait for heartbeat from GCS Update & Command program;
wait for flag_runLoop =True command from GCS;
while flag_runLoop = True do
    check autopilot heartbeat;
    if flight mode = GUIDED then
        set start and end location;
        set obstacles;
        set flag_runAlgorithm =True;
    end
    if flag_runAlgorithm = True then
        start thread for calculating route;
        set flag_runAlgorithm = False;
    end
    if flag_newMission = True then
        set mission parameters, i.e., route, costs, timetaken;
        set flag_sendToNextWP = True and nextWP = False;
        set flag_newMission = False;
    end
    if flag_sendToNext,WP = True then
        send mavlink command for next WP;
        set flag_checkDistToNextWP = True;
        set flag_sendToNextWP = False;
    end
    if flag_checkDistToNextWP = True then
        if lateral distance to WP < 1.2 × LoiterRad and within 10m of WP altitude then
            if nextWP = lastWP then
                Mission complete;
                set flag_checkDistToNextWP = False;
                set flag_runLoop = False;
            else
                set nextWP = nextWP +1;
                set flag_sendToNextWP = True;
                set flag_checkDistToNextWP = False;
            end
        end
    end
end
```

For example, a 'STATUSTEXT' status message might be sent from the UAV Mission Control to the GCS Update & Command program with the text field containing 'Running algorithm'. This message would signify any time the GA algorithm is running, and would be sent continuously

Figure 6.5: Empty GCS Update & Command program window, written in TKinter (Python), for showing vital information sent from the UAV Mission Control program.

until an acknowledgement message is received from the GCS side. This effectively halts the code until the message is confirmed as both sent and received, a process which could slow down the system however has been found to work adequately in simulation. Acknowledgement messages are sent as a 'COMMAND_ACK' message with a unique integer value for each message, a message type that is usually reserved for acknowledging system commands.

6.4 Flight Testing Results

This section presents results from four RTTP flights and a number of conventional flights. A comparison is made with, and the cost function applied to, the routes of the conventional missions. These results demonstrate that it is possible to fly a fixed-wing UAV mission using an onboard RTTP, and to compare these initial flight tests to conventionally navigated flights in the same area with similar parameters. These results are then discussed in Section 6.5, and further development of the RTTP is presented in Section 6.6.

Plots in this section show just the climbing phase flight path, as this is when the RTTP was used. Time spent monitoring the volcano, and the descent phase of the flight, have been removed for clarity.

6.4.1 RTTP Flights

The four RTTP flights presented here took place on Trip E. Every flight involved calculation of a new and valid route, however some were not flown to completion. One successful and one failed flight is presented for a case with no obstacles, and one successful and one failed flight is presented for a case with two obstacles.

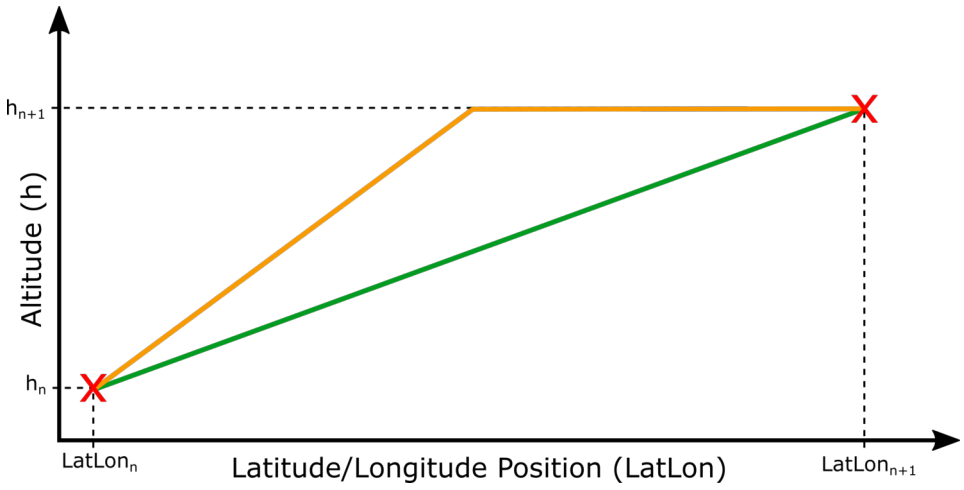

Figure 6.6: Figure showing aircraft climb profile types, between waypoint n and waypoint $n+1$ (represented by two red crosses). The intended climb profile is shown in green, and poor autopilot climb profile shown in orange.

Previously seen predominantly on Trip B in the spiral climb, the autopilot has sometimes been known to climb at the maximum possible angle to reach the altitude required, rather than follow the straight-line flight path linking waypoints (Figure 6.6). Manually generated mission plans can take this into account, however the behaviour demonstrated by the UAV during RTTP flights (in GUIDED mode) showed this same approach to climbing. This behaviour reduces flight efficiency and in some cases has been known to cause extremely high-drag flight configurations.

Initially, the start location, end location, and any obstacles were set via the GCS Update & Command program. The aircraft was then launched and switched to GUIDED mode, circling at that location until a solution was found. Upon receiving a new mission, the UAV flew to the start location before following the calculated route. RTTP parameters such as threshold solution cost and population size were tuned using hardware-in-the-loop (HITL) simulations, and kept constant across all four flights in order to maintain comparability.

The results parameters of the four RTTP solutions are given in Table 6.1. Note that the average time taken per iteration for these solutions range from 2.01 seconds to 4.30 seconds; this is calculated by dividing the total time taken by the number of iterations (i.e. generations).

Table 6.1: Real-Time Trajectory Planner real-world operations results.

Flight	**E17**	**E18**	**E19**	**E23**
Obstacles	None	None	Two	Two
Number of Iterations	2	14	39	73
Time Taken	5.15 s	28.09 s	167.61 s	162.60 s
Worst Cost	74.67	65.70	53.98	58.99
Best Cost	1.196	1.126	1.343	1.345

6.4.1.1 Flight E17

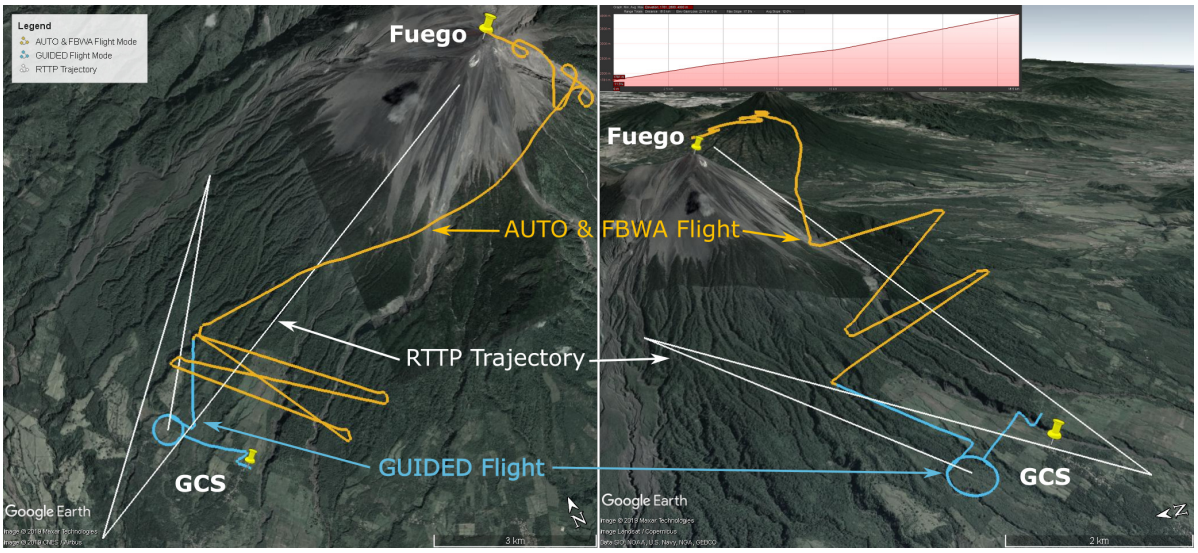

Figure 6.7: Flight E17: top down (left) and 3D (right) maps showing the Real-Time Trajectory Planner solution (white) and the actual flight path (GUIDED: Blue, AUTO/FBWA: Orange) in Google Earth [10]. The elevation profile of the RTTP Trajectory is inset (top right).

This was the first flight to use the RTTP in Guatemala (on the 31st March 2019). A solution was found rapidly and the aircraft sent to the start location. In order to progress to the second waypoint, the aircraft had to be within a threshold radius of the first waypoint. Aircraft parameters meant the UAV circled the waypoint with a radius larger than this threshold, effectively halting progress. This was overridden via the GCS Update & Command Program, and the UAV was sent to the next RTTP waypoint; however, the operational decision was made during the first leg to switch to AUTO mode and continue the flight conventionally. In order to combat this issue, the Mission Control code was modified to increase the threshold distance. Flight E17 is included in the results in part because it represents the first use of an onboard RTTP generating and following a BVLOS flight plan, despite it not reaching the end location.

The parameters of the RTTP solution are given in Table 6.1 and a plot of the trajectory is shown in Figure 6.7. Note the inset gradient of the RTTP trajectory in Figure 6.7, with an average gradient of 7°, maximum gradient of 9.7°, and no excess climbing.

6.4.1.2 Flight E18

This flight was the first successful implementation of Algorithm 4, and also took place on the 31st March 2019. The aircraft reached the target location in 21 min 13 s with 51.1% (LiPo) battery capacity remaining. Flight parameters are presented in Table 6.1, the flight path and RTTP trajectory are shown in Figure 6.8, and a graph of the GA iteration costs is shown in Figure 6.9. For scale, the length of the longest leg in Figure 6.8 is 11.3 km.

Note the inset gradient of the RTTP trajectory in Figure 6.8, with an average gradient of 8.1°. The profile is relatively consistent throughout the trajectory with a maximum gradient of 10.2° and no excess climbing.

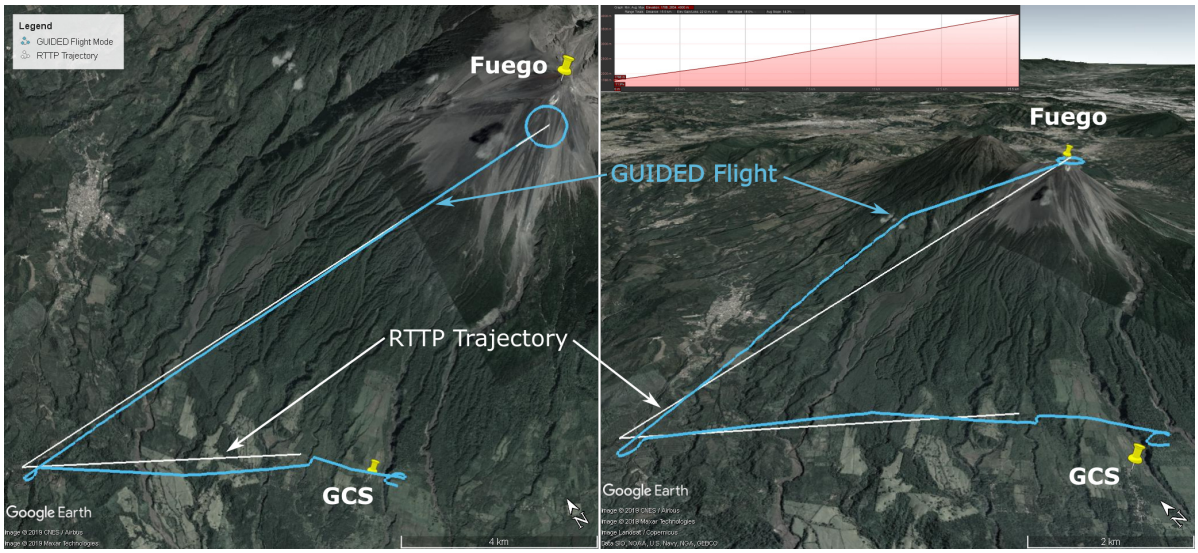

Figure 6.8: Flight E18: top down (left) and 3D (right) maps showing the Real-Time Trajectory Planner solution (white) and the actual flight path (GUIDED: Blue) in Google Earth [10]. The elevation profile of the RTTP Trajectory is inset (top, right).

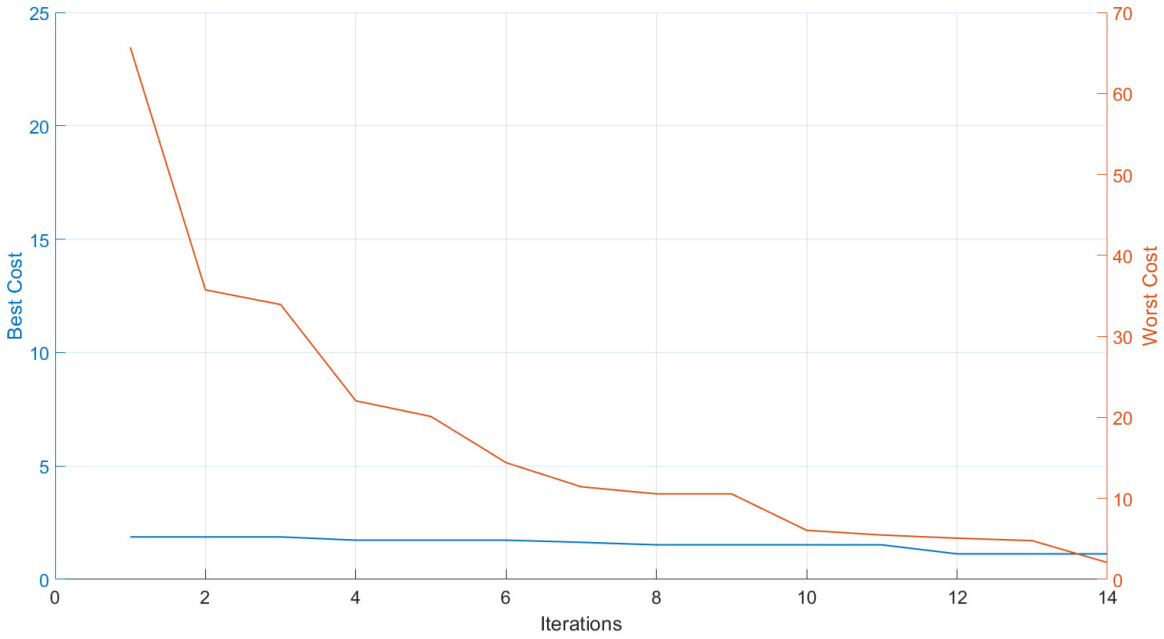

Figure 6.9: Graph showing the best (blue, left) and worst (orange, right) solution costs for each iteration of the GA for Flight E18. Note the different scales on each of the y-axes.

6.4.1.3 Flight E19

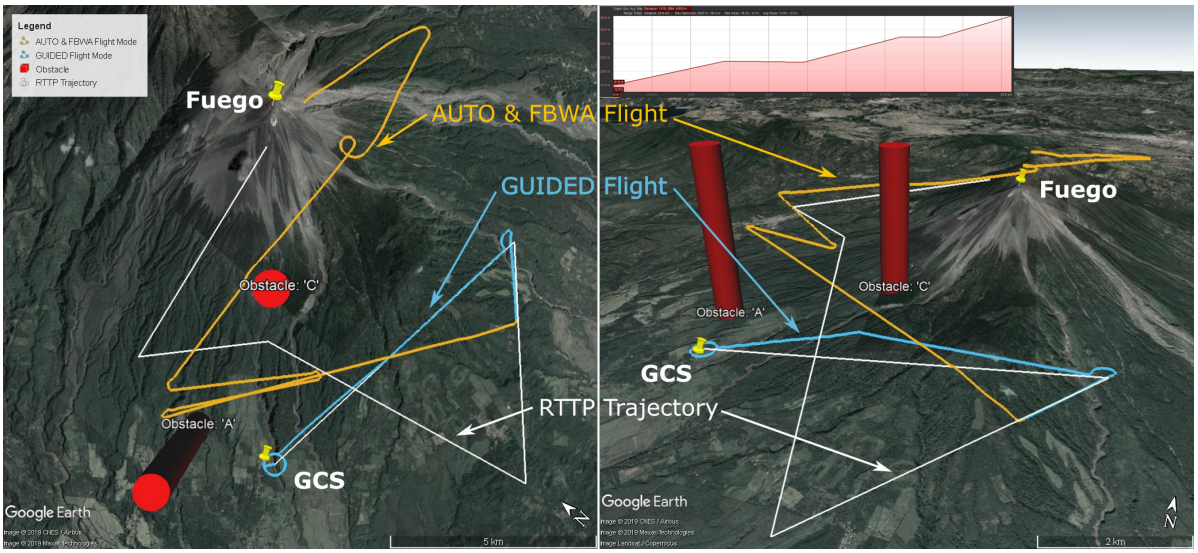

Figure 6.10: Flight E19: top down (left) and 3D (right) maps showing the Real-Time Trajectory Planner solution (white) and the actual flight path (GUIDED: Blue, AUTO/FBWA: Orange) in Google Earth [10]. Obstacles are shown by red cylinders, and the elevation profile of the RTTP Trajectory is inset (top, right).

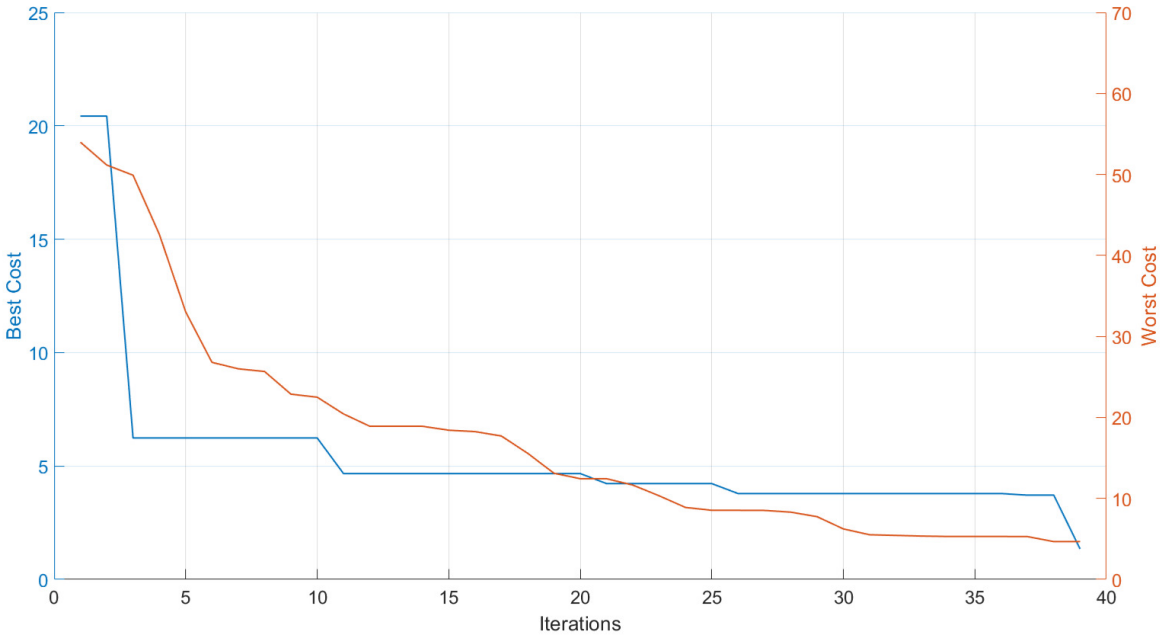

Figure 6.11: Graph showing the best (blue, left) and worst (orange, right) solution costs for each iteration of the GA for Flight E19. Note the different scales on each of the y-axes.

This E19 was the first to use the RTTP with obstacles over Fuego in Guatemala, positioned at arbitrary locations between the GCS and summit to challenge the RTTP. A valid route was generated; however, it was only followed in part before switching to the conventional flight path due to the operational decision to end the RTTP in order to sample a desirable eruption. Route parameters are given in Table 6.1, the flight path and route are shown in Figure 6.10, and a graph of the GA iteration costs is shown in Figure 6.11.

Note the inset gradient of the RTTP trajectory in Figure 6.10, with an average gradient of 7.9°. The profile has a maximum gradient of 9.3° with 38 m of excess climbing.

6.4.1.4 Flight E23

This flight was the first successful implementation of Algorithm 4 with two 'No Go' zones in place, and took place on 2nd April 2019. The aircraft reached the penultimate waypoint, at approximately summit altitude, in 25 min 41 s, with 41.8% (LiPo) battery capacity remaining. The UAV was not allowed to reach the final RTTP waypoint because there was a plume a short distance West of the path between the final two waypoints, and FBWA mode was used to intercept the plume for sampling purposes. Flight parameters are presented in Table 6.1, the flight path and RTTP trajectory are shown in Figure 6.12, and a graph of the GA iteration costs is shown in Figure 6.13.

Note the inset gradient of the RTTP trajectory in Figure 6.12, with an average gradient of 6.2°. The profile starts and ends with a steeper section, has a maximum gradient of 10°, and has no excess climbing.

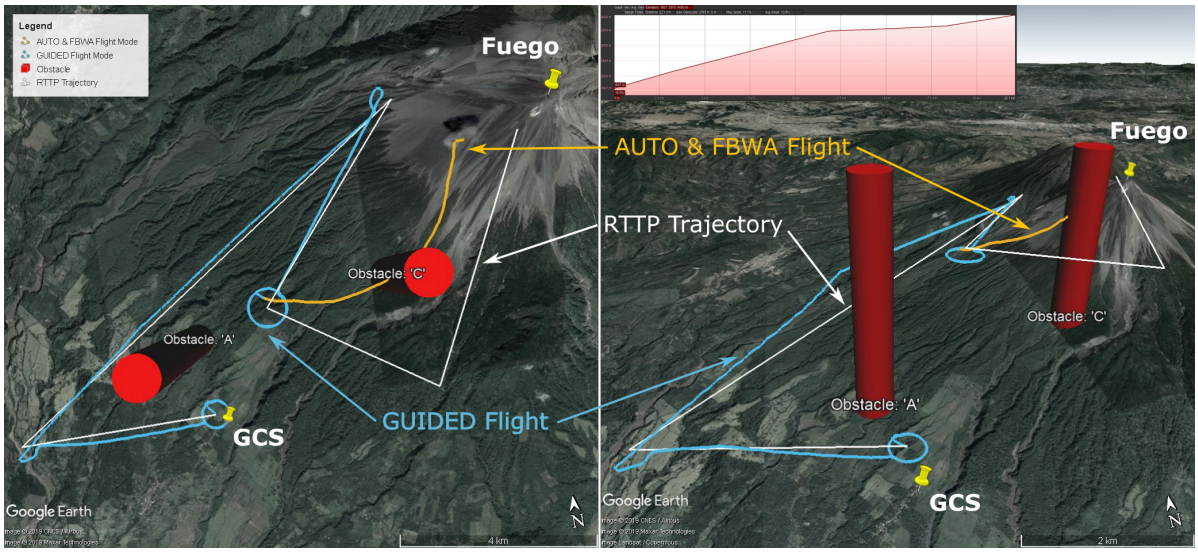

Figure 6.12: Flight E23: top down (left) and 3D (right) maps showing the Real-Time Trajectory Planner solution (white) and the actual flight path (GUIDED: Blue, AUTO/FBWA: Orange) in Google Earth [10]. The elevation profile of the RTTP Trajectory is inset (top, right).

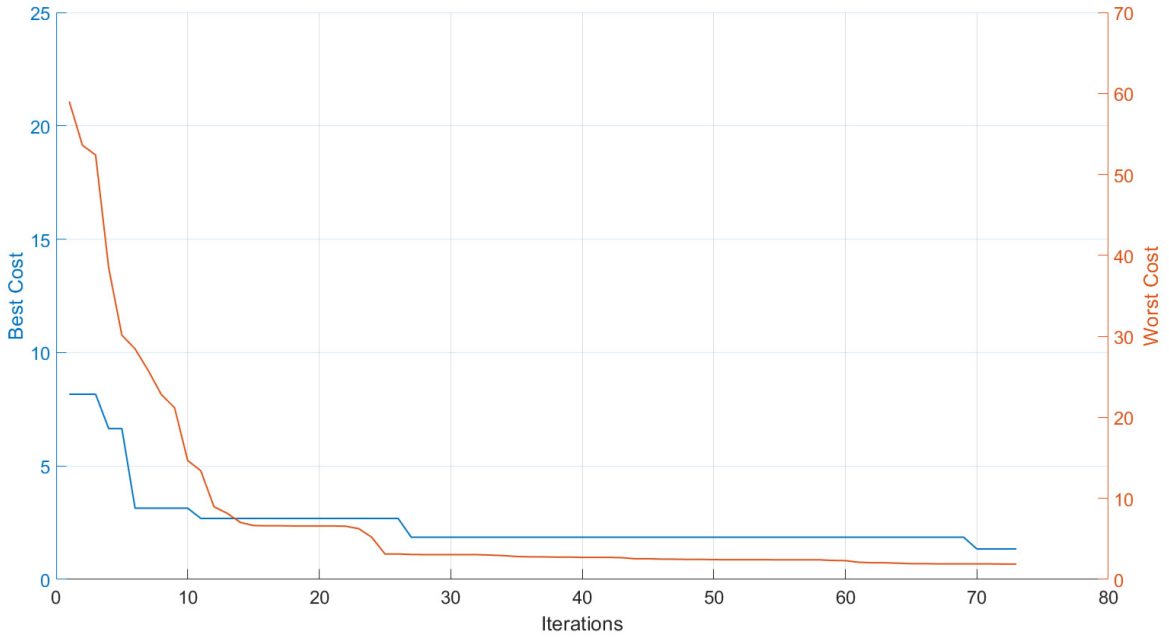

Figure 6.13: Graph showing the best (blue, left) and worst (orange, right) solution costs for each iteration of the GA for Flight E23. Note the different scales on each of the y-axes.

6.4.2 GCS Update & Command

Figure 6.14 shows a typical update box in the GCS Update & Command program. While the system worked well in simulation and for HITL tests, it was slow when used with real-world long range telemetry links. Text and vector messages were received reliably (i.e. status updates and waypoints), however full missions were not successfully received. The route plotting function was therefore of minimal use, and waypoint locations were interpreted and assessed for viability in real-time by plotting the received coordinates on a separate map.

6.4.3 Conventional Flights

RTTP routes can be compared to conventional mission plans, however for the purpose of advancing efficiency only the best version of the manually developed plans is considered. As introduced in Section 4.3.5, these involve straight-leg climbs rising above the GCS before advancing towards the summit at high altitude. The end location may not be the same exact point as in the RTTP routes, however for the sake of comparison the distances between end points in the immediate vicinity of Fuego's summit are considered negligible.

Only flights matching certain criteria are considered here; the aircraft type must be consistent (i.e. Skywalker X8), with similar payload and equipment configurations. The weather naturally varied between flights, and the mass of the aircraft was increased for the RTTP missions by the addition of the Raspberry Pi and associated cables (approximately 50 g).

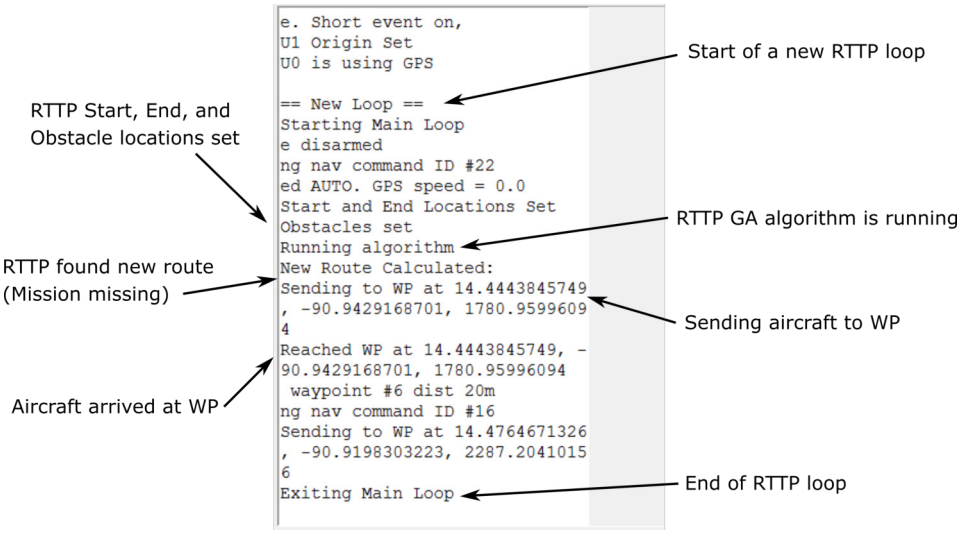

Figure 6.14: A labelled GCS Update & Command Program screen from flight E17.

A conventional straight-leg climb method was developed through trial and error, significantly reducing the energy required to reach both the summit altitude and area compared to initial attempts. A diagram of this route type is shown in Figure 4.9 and an example flight log track is shown in Figure 6.15. Key parameters of these 'standard' mission types are presented in Table 6.2, alongside data from each of the RTTP flights.

Airspeed data show that conventional mission plan flights had a higher average airspeed but also a higher standard deviation compared to RTTP flights (Table 6.2). The lower airspeeds of RTTP flights can be explained by the steep, inefficient climb angle used by the autopilot in these

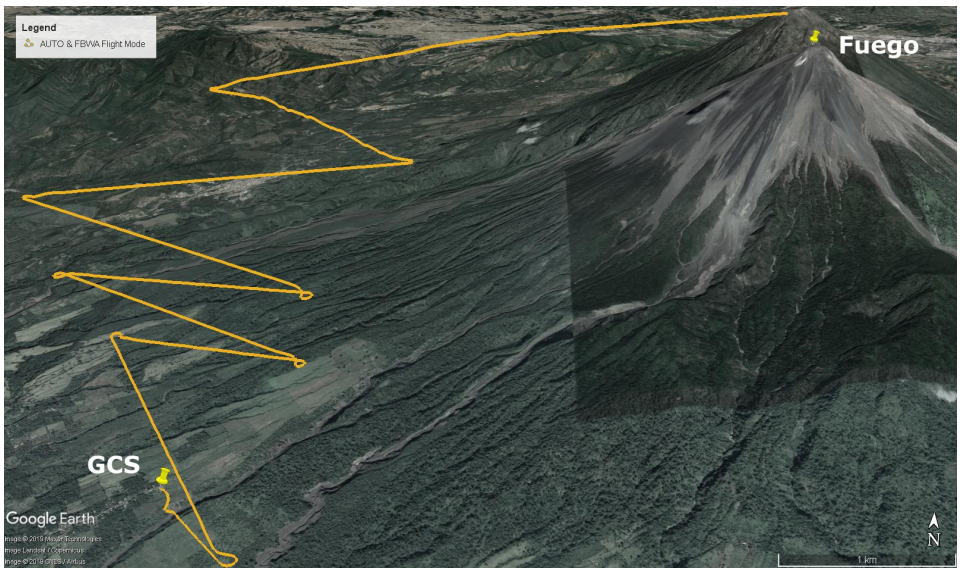

Figure 6.15: Example of a conventional leg climb route, developed over a series of trips.

Table 6.2: Comparison data for different mission types, with conventional mission data presented as Mean / StdDev, $n = 10$.

Parameter	**Conventional Mission**	**E17**	**E18**	**E19**	**E23**
Distance (meters)	25,392 / 2603	18,319	15,301	25,421	21,925
Time (mins:secs)	24:19 / 03:40	n/a	21:13	n/a	25:41
Capacity Used (mAh)	8101 / 709	n/a	7828	n/a	9320
Route Cost (no Obstacles)	2.253 / 0.056	1.196	1.126	n/a	n/a
Route Cost (Obstacles)	2.649 / 0.448	n/a	n/a	1.343	1.390
Airspeed: Mean (m/s)	24.5 / 0.6	21.8	21.2	21.7	22.6
Airspeed: Std Dev (m/s)	4.6 / 0.4	3.8	4.0	3.4	3.9

cases. Comparing airspeed and calculated mean ground speed ($\frac{ground\ distance}{flight\ time}$), the effect of the wind around Fuego is clear; for example, the average ground speed for conventional missions was 17.4 m/s compared to an airspeed of 24.5 m/s.

6.5 Flight Testing Discussion

As mentioned in Section 6.4.1, a behaviour shown by the autopilot in certain instances can cause the UAV to climb at its maximum climb rate until the target altitude has been reached (Figure 6.6). The RTTP flight logs show this behaviour to be present while in GUIDED mode. The effect of this on the flight path is particularly visible in Figure 6.8 by comparing the top-down view and the three-dimensional view. Climbing in this manner typically induces excessive drag and slows the aircraft down, compared to a gradual climb along the flight path which arrives at the target altitude upon reaching the waypoint. An example of a conventionally generated trajectory that takes this behaviour into account can be seen by the orange AUTO/FBWA flight track in Figure 6.7. The real-world efficiency of the RTTP flights presented here is therefore worse than the results expected should this behaviour be addressed.

Flight E17 and Flight E18 represent RTTP paths that do not take into account any obstacles, and are compared to data from conventional missions in Table 6.2. While conventional missions have been carefully designed to climb near the GCS and spend much of the approach to Fuego at summit and plume altitude, the RTTP trajectories are not restricted to these constraints. RTTP routes E17 and E18 average a trajectory length 34% shorter than conventionally planned missions, with cost function values of approximately half and a significantly quicker transit time. The only RTTP route flown to completion here is Flight E18 and the battery capacity remaining at the end location, despite the shortcomings of the autopilot's behaviour in GUIDED mode, represents a 2% saving. With a smoother implementation of the climb by the autopilot, this small saving is likely to increase significantly.

While the obstacles used for Flight E19 and Flight E23 (Figures 6.10 and 6.12 respectively) do not represent anything tangible, the RTTP was tested with them for completeness. Conventional

mission plans (Figure 6.15) have not been designed with these obstacles in mind, hence a true comparison to Flights E19 and E23 can not be made. Table 6.2 shows an average RTTP trajectory length that is slightly shorter (7%) than planned missions, but a slightly longer transit time and 7% more battery consumption. The shorter missions with longer transit time indicate environmental factors such as wind affecting the true ground speed of the aircraft. One area of future RTTP development involves incorporating wind speed and direction, using environmental factors to further increase overall flight efficiency. The cost function for RTTP routes is 40% less than the conventional routes; however, the function was tuned for the RTTP so it cannot be considered a pure form of comparison. The autopilot's climbing behaviour partly explains the comparative inefficiency of Flight E23 and, as is true for all RTTP data presented here, further development and testing is required before significant efficiency savings can be claimed. Some of these improvements are developed and discussed in Sections 6.6 and 6.7.

The primary achievement of these tests is the proof of concept for real-world RTTP implementation, however the performance of the algorithm remains an important aspect of the system's functionality. Comparing computation times between problems with and without obstacles indicates a poor mechanism for coping with obstacles and high values of minimum time per iteration (Table 6.1, 2.01 seconds). While the GA did find valid and suitable routes, the length of time it took to find them means further work must focus on the streamlining of this code.

Figures 6.9, 6.11, and 6.13 show the best and worst costs of each iteration for flights E18, E19, and E23 respectively. The high values for 'worst' costs suggest good exploration of the solution space, however the relatively flat 'best' costs suggest that there has been limited exploitation of the existing solutions. That said, the time taken per iteration limits these runs in the real world; better performance would be gleaned by both speeding up the GA and conducting tests of parameters which control the weighting of exploration and exploitation.

The GCS Update & Command program, shown in Figure 6.14, proved useful for monitoring the RTTP in flight. While the missions did not reliably come through to the ground station, status and waypoint messages were received consistently for use alongside traditional GCS programs such as Mission Planner. The lack of useful plotting tool meant a higher workload for the GCS team, manually locating each waypoint on a separate map in order to keep track of the mission plan. As an initial test system this setup proved sufficient, however future work must involve redesigning the Update & Command program, with the workload of the GCS team in mind.

An RTTP removes the need to manually develop a conventional flight plan. Tuned correctly, and with dynamic assessment of conditions and re-routing, this would avoid the typical trial-and-error approach to flying in new locations. The conventional flight plan that is shown in Figure 6.15 took several trips and over 20 flights to develop, however the RTTP trajectories shown here are comparable according to many of the parameters in Table 6.2. Not only might an RTTP offer energy savings in-flight, but also increased operational efficiency in the more general case when operating at new locations.

6.6 Refinement Methodology

This section presents further analysis and refinement of the RTTP system, following Trip E and the publication of the journal paper titled 'Onboard Real-Time Trajectory Planning for Fixed-Wing Unmanned Aerial Vehicles in Extreme Environments' [29]. The UAV Mission Control and GCS Update & Command programs were also developed further, for use with both the refined RTTP and the Coordinated Plume Interception (CPI) scheme presented in Chapter 7.

6.6.1 UAV Mission Control

The refined UAV Mission Control program developed here is based on the original RTTP Mission Control program, and follows a similar structure to Algorithm 4. A modular and more object-oriented approach was taken to developing it further, in order to accommodate generic schemes that involve following a series of waypoints.

The RTTP was used in Trip E once per flight, and the GA was only initiated once GUIDED mode was engaged. The approach taken in further developing this program means that, depending on the mode of the companion computer, the GA starts to generate routes before take-off and continues to do so in the background throughout the flight. The regularity of this re-calculation can be modified in the program parameters. This means that the loitering phase previously used to calculate a solution before following a route is avoided, thereby saving energy and time.

This script was developed alongside the GCS Update & Command program, and a revised implementation of the MAVLink communications protocol. These changes are presented below.

6.6.2 GCS Update & Command

The version of the GCS Update & Command program used on Trip E performed poorly during real-world testing, and the following key areas of improvement were identified:

- Show map of flight area, including obstacles where appropriate
- Show incoming missions on flight area map
- Show incoming mission and terrain altitudes on an altitude plot
- Plot aircraft location and altitude in real-time
- Show key mission metrics, such as length and maximum gradient
- Require reliable MAVLink communications with companion computer

A user interface such as this, designed for real-world long range UAV operations, requires clear and concise information to be relayed to the operator while maintaining system robustness.

The updated design of the GCS Update & Command program relied on the cumulative experience gained across Trips A – E, and an annotated example is given in Figure 6.16.

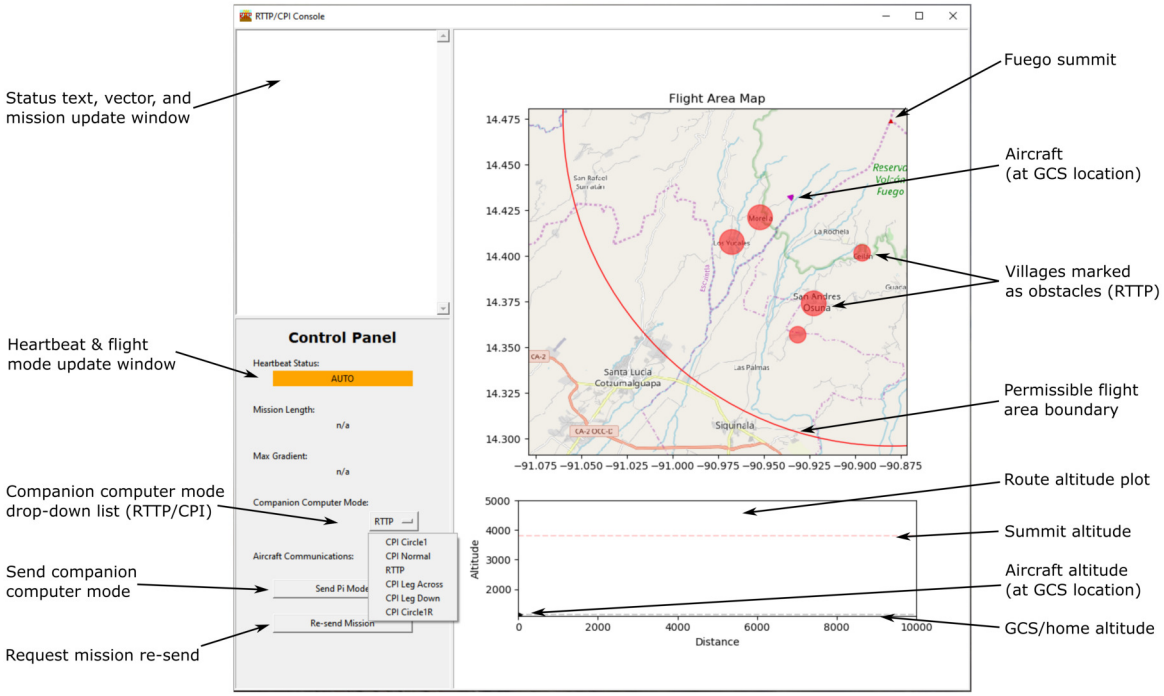

Figure 6.16: Revised GCS Update & Command console, annotated. While update messages continue to be printed in the text window, aircraft position and missions are plotted on the map and altitude plots when available. The control panel features key information about the aircraft and mission, and lets the user define the companion computer mode. This was designed for use with both the RTTP and CPI schemes. Map source: Open Street Map [12].

Table 6.3: Revised GCS Update & Command control panel description.

Item	**Purpose**
Heartbeat Status	Colour coded: Green for GUIDED mode, Orange for any other mode, Red for no heartbeat/link with aircraft
Flight Mode	Indicated by text, inside the coloured Heartbeat Status box
Mission Length	Length of mission in km, ‘n/a’ if no mission yet received
Max Gradient	Maximum mission leg gradient in degrees, ‘n/a’ if no mission yet received
Companion Computer Mode	Drop-down box selection, including RTTP mode and various CPI methods (described further in Chapter 7)
Send Pi Mode	Send the mode selected in ‘Companion Computer Mode’ to the Raspberry Pi companion computer
Re-send Mission	Request the latest mission to be re-sent from the aircraft, for plotting

Text updates from the aircraft show in the status update window in Figure 6.16, some examples including 'Starting Script', 'Calculating route', and 'Going to waypoint [*lat, lon, alt*]'. The messages in this version are revised for better information transfer, and identical consecutive messages cannot be printed to the window. This reduces unnecessary re-printing and therefore keeps the updates easy to read and process. Newly received mission plans are printed to this window too, but also plotted onto the map and altitude plots. This happens automatically upon receipt of a mission, and any new missions erase and replace existing plots. The terrain data used onboard the aircraft for evaluating the RTTP cost function is interrogated here too, and plotted on these plots as a guide for the GCS operator. Markers for aircraft position and altitude are also updated regularly, to better show the progress of the vehicle along the route. The control panel section of Figure 6.16 is described by Table 6.3.

One area of the system that has been revised significantly since Trip E is the use of the MAVLink message protocol. These revisions aim to increase the proportion of messages that are successfully transmitted, and minimise delays in the real-world system. The version used for Trip E involved a custom message ID for every message, which was sent to the sending component upon receipt of the message in question. The sending side would send repeatedly until

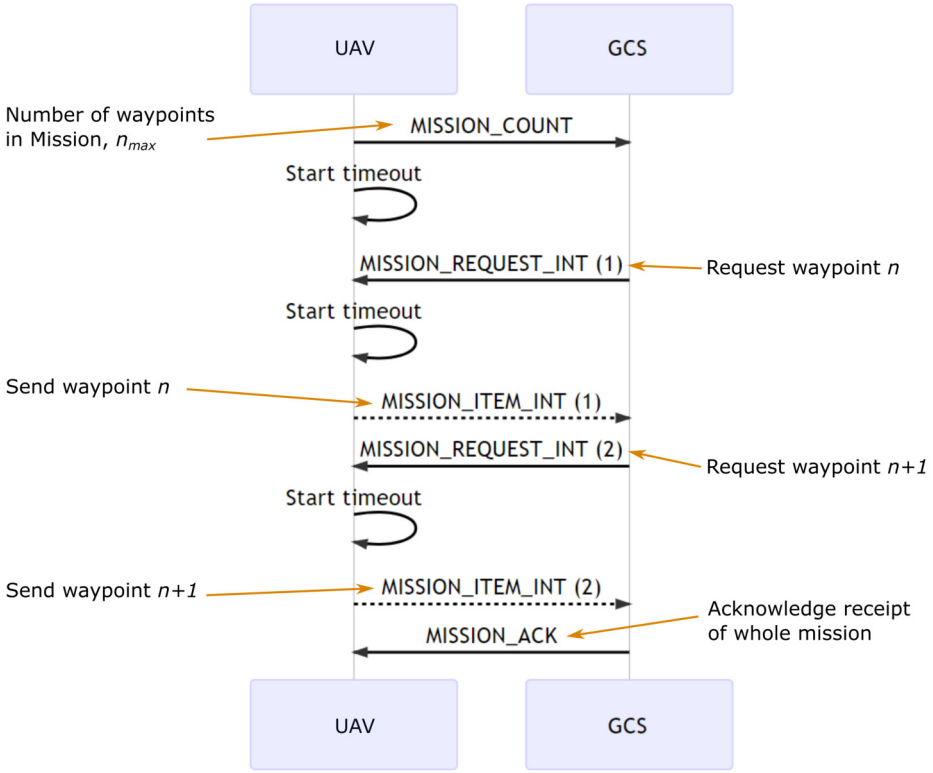

Figure 6.17: Annotated diagram of the process used to send missions from the UAV to the GCS, based on a graphic from mavlink.io [13]

this acknowledgement was received, and this was written into the programs on each component separately. The revisions involved removing duplicated code across all components and increasing robustness, building up a consistent centralised code base that is available to both sides of the link. All MAVLink communications were written into a thread, used by both the UAV Mission Control program and the GCS Update & Command program, making use of a centralised message definition dictionary. Although all MAVLink messages are still sent repeatedly, they now use a 1500 millisecond time-out between send attempts, reducing unnecessary bandwidth use.

The method of sending missions from the UAV to GCS is similar to the approach used for the Trip E, however the implementation changed in line with the centralised MAVLink thread. This approach makes use of the existing mission protocol style, shown in Figure 6.17.

Re-formatting the code from Trip E involved making functions and classes more modular, and better use of threading. The UAV would now typically run three threads; the main one running the central mission manager, one running the MAVLink thread, and a third for RTTP route calculations. Interaction between these threads means that the main mission manager can add messages to the mavlink thread task list without inadvertently halting the progress of the mission manager script. Changes made specifically to the GA are discussed in Section 6.6.4.

6.6.3 Flight Area, Obstacles, and Waypoint Generation

A revised and well defined flight area is used for the new RTTP and CPI schemes. This is partly due to the complexity of the previous (Trip E) version, and because the radius of operations has been expanded to accommodate the CPI scheme. Future trips will use the area described in the following section, pending approval from DGAC and an assessment of safety measures.

The new flight area uses the summit area of Fuego as a reference point, with a radial lateral limit of 20 km imposed in a South-Westerly direction (Figure 6.18). An Eastern limit is drawn by a North-South line from the summit area, and a Northerly limit is drawn by an East-West line from the crater area. An additional small circle with a nominal radius of 2 km is centred on Fuego to allow for a range of proximal flight patterns (not shown in Figure 6.18).

RTTP waypoints were previously generated randomly using a Gaussian distribution around the midpoint between the start and end locations, with a standard deviation tuned such that routes from the GCS to the summit were within the flight area. The revised method instead generates waypoints randomly, with a uniform distribution across the flight zone described above. One of the reasons for keeping flights between the GCS and Fuego was safety; keeping the aircraft near the GCS and over unpopulated areas while at the lower stages of the climb. A combination of developed experience and informed route planning means that a reasonable level of safety can be maintained with this new method, and it offers better routes with less computational time.

Part of the safety built into this new flight zone is the use of obstacles, or No Go zones, at villages and population centres in the area. Shown on the map in Figure 6.18, these obstacles are represented by circular areas of infinite height, with a radius that covers the necessary space.

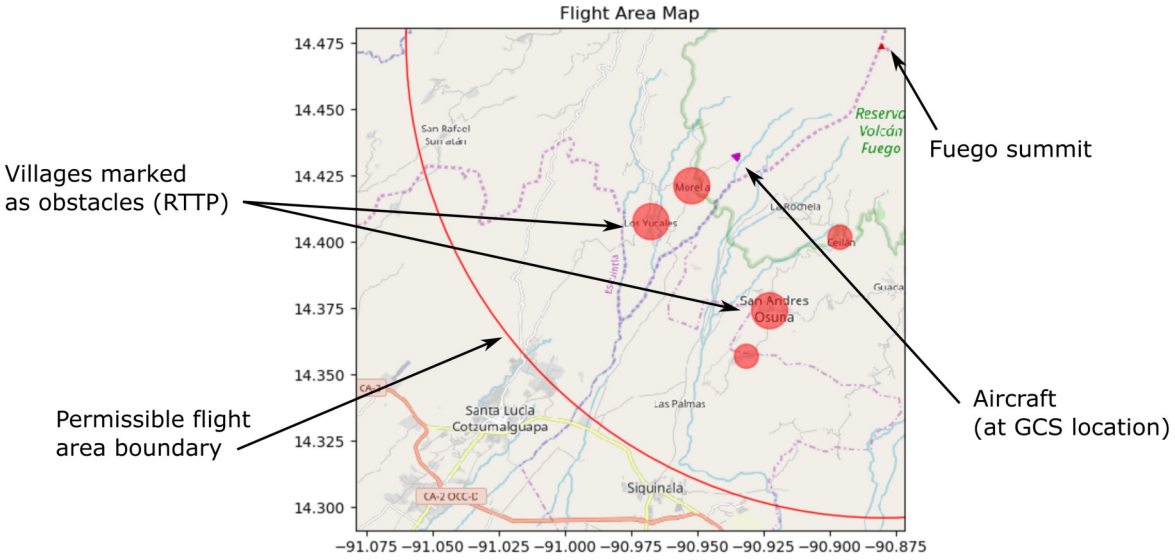

Figure 6.18: Map of flight area around Fuego, as used in the revised GCS Update & Command graphical user interface. This figure is annotated with a 20 km radius of operations and 5 village-based obstacles. Map source: Open Street Map [12].

The RTTP used on Trip E was tested with and without obstacles, however those obstacles did not represent anything tangible in reality. The new obstacles represent real-world No Go zones, and their inclusion is no longer optional for the RTTP. This means that the highlighted areas will always be avoided, helping to build the safety case around conducting these missions. The flight test results suggested that finding routes with obstacles could take an impractical amount of time, however changes made to the GA since Trip E have yielded significant speed-ups and therefore allowed this new approach to be practical.

6.6.4 Genetic Algorithm

In order to further develop the RTTP system, the GA required significant work. While the early version's final results proved satisfactory, the time taken to find the solutions was impractical for real-world use, particularly for the cases where obstacles were used. The length of time required for each iteration meant the GA had to converge rapidly in order to achieve a reasonable run time, however this is not a guaranteed feature of a GA. All development and tests presented in this section took place using a 64-bit Windows 10 operating system with 8 GB of RAM and an Intel Core i5 processor (2.4 GHz). The start and end locations for the GA in this section was kept constant, starting at the INSIVUMEH Observatory and ending at 4000 m AMSL above Fuego's summit.

6.6.4.1 Profiling and Cost Function Speed-Up

The Python profiling tool SnakeViz was used after Trip E to assess the time spent on different areas of the algorithm. This was used on the genetic algorithm function, bypassing the generation and implementation steps, for a typical route problem (with no obstacles) over 10 iterations. The output of the profiler is shown in Figure 6.19 (a); the program took 30.4 seconds to run, with 13.4 seconds of that time spent on the mutation operator and 15.5 seconds spent on the crossover operator. The output of both of these operators calls the cost function, in order to evaluate the new route. The cumulative time spent evaluating the cost function was 29.9 seconds, 98.4% of the total run time. Of this, the Geodesic Inverse function took 28.5 seconds. This is a freely available Python function that was used in this instance to find the distance between two geographical positions.

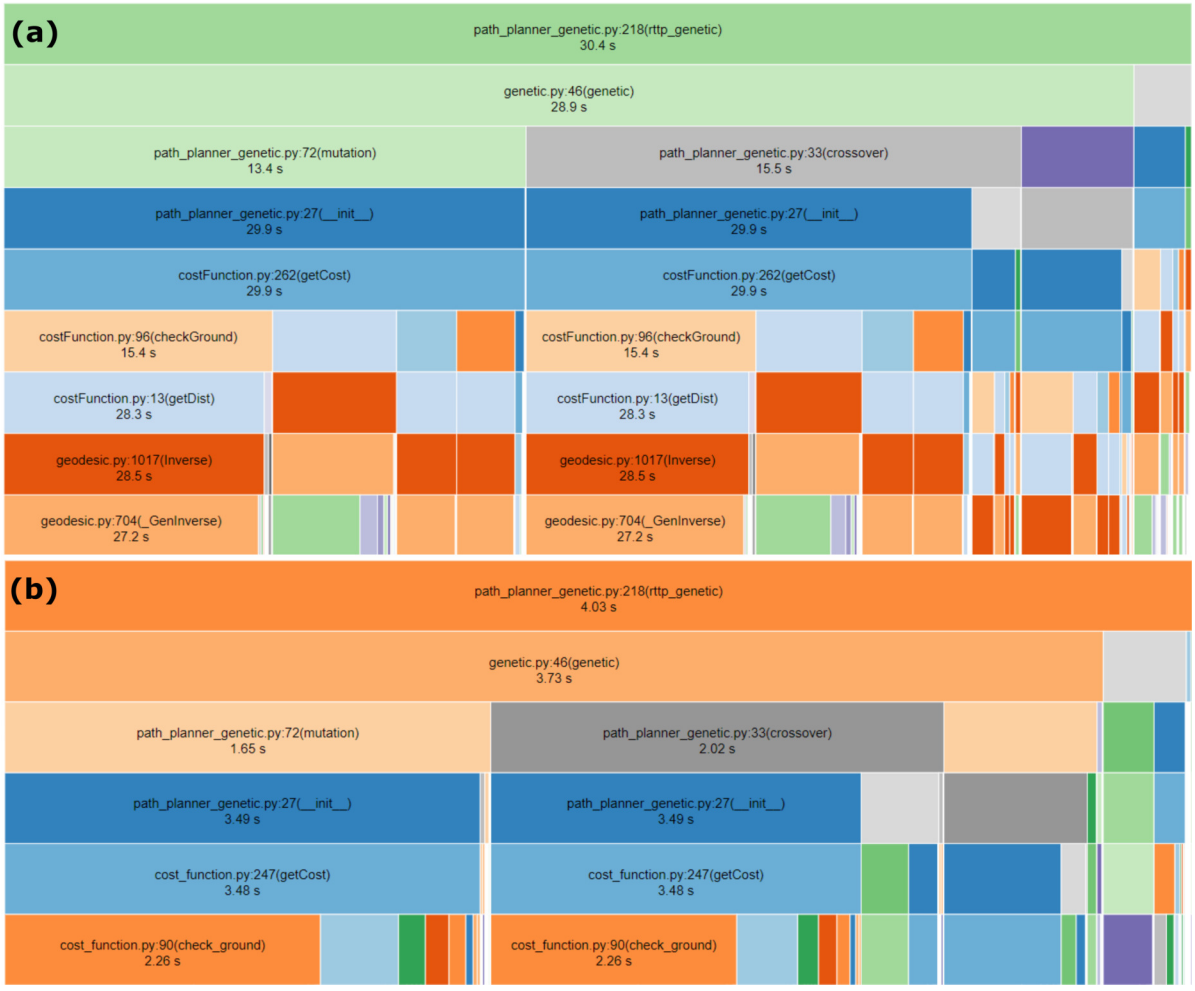

Figure 6.19: Visualisation of the speed-up achieved (a) before and (b) after modifying the cost function, considering a 10 iteration run of the genetic algorithm. This was generated using the SnakeViz profiling module for Python.

To eliminate the bottleneck imposed by the Geodesic Inverse function, the trigonometric Haversine formula was used to calculate distances instead:

$$a = sin^2\left(\frac{\Delta lat}{2}\right) + cos(lat_1) \times cos(lat_2) \times sin^2\left(\frac{\Delta lon}{2}\right) \tag{6.18}$$

$$c = 2 \times atan2(\sqrt{a}, \sqrt{1-a}) \tag{6.19}$$

$$d = R \times c \tag{6.20}$$

where the positions are given by $[lat_1, lon_1]$ and $[lat_2, lon_2]$, R is the radius of the Earth, and d is the distance between positions. This formula gives an approximate distance between two locations with reasonable accuracy. The use of these distances is primarily comparative and these calculations relate to a small area of the Earth, so it is assumed that the error in these approximations is systematic and consistent, and thereby negligible in this context.

Figure 6.19 (b) shows the results of the same 10 iteration test after replacing the Geodesic Inverse function; the overall run time reduced from 30.4 seconds to 4.03 seconds, and the cumulative time spent evaluating the cost function reduced by 88%, from 29.9 seconds to 3.48 seconds. The consequent speed-up of the GA means it has become feasible to use obstacles in a standard application of the RTTP, and the shorter time-per-iteration means that higher-iteration runs are feasible in real-world applications.

6.6.4.2 Threshold Cost

The quality of the routes found for a given cost function depends on the algorithm satisfaction criteria. The threshold cost (τ_c) for Trip E was 1.5, however this could be made stricter with the changes described above. A series of tests were conducted to characterise the effect of τ_c on solution quality, the results of which are given in Figure 6.20. Initial tests indicated that a reasonable value was likely to be in the range [1.0, 1.2], so this range was interrogated five times at intervals of 0.05.

Figures 6.20 (a) and 6.20 (b) show the routes and altitudes of the paths respectively; those with a higher τ_c tend to have more legs with more variation in flight path angle. Note, the maximum flight path angle for any leg shown here is 9.73°, thereby meeting the 10° limit imposed in the RTTP parameters. Figure 6.20 (c) shows the best cost of each iteration in every test, and shows a plateau after approximately 200 iterations for $\tau_c = 1.10, 1.05$, and 1.00. The timings of the runs are shown in Figure 6.20 (d). Some of the lower threshold runs, including all those with a value of 1.00, did not reach convergence and instead hit the 500-iteration limit of the test.

In general, lower τ_c values gave qualitatively and quantitatively better solutions, however took more time and iterations to find them. The adopted solution involves combining various values of τ_c in a time dependent tiered structure. This means that the occasional early convergence of runs with τ_c = 1.05, 1.10 is possible, while retaining the consistently quick convergence features of runs with higher τ_c values. The following tiered threshold function is suggested:

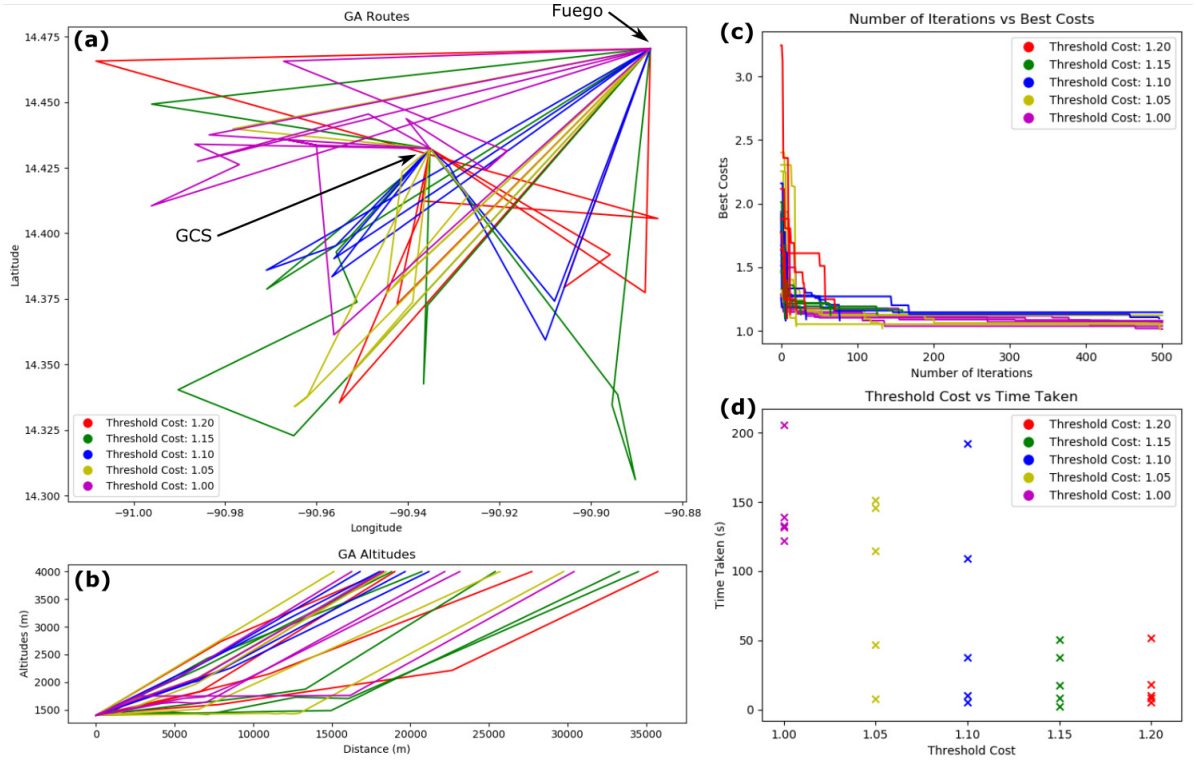

Figure 6.20: Results from a series of tests comparing the effect of different threshold costs (τ_c) on the performance of the GA, with a maximum of 500 iterations. (a) shows the routes on a map (from GCS to Fuego), (b) shows the altitude profiles of these routes (maximum path angle 9.73°), (c) shows the best cost at each iteration for each run, and (d) shows the time taken to complete each run.

$$\tau_c = \begin{cases} 1.1, & 0 \leq t < t_B \\ 1.2, & t_B \leq t < t_C \\ 1.5, & t_C \leq t \end{cases} \tag{6.21}$$

where t is the run time, and the parameters t_B and t_C relate to the acceptance times for tiers B, and C. For development purposes these time limits were set to 40 s and 50 s respectively.

6.6.4.3 Replacement Method

As described in Section 6.2.4, the Trip E version of the GA placed accepted child chromosomes at the end of the generation, rather than the position from which the parent was taken. A test was conducted to challenge this, comparing local and end replacement methods, the results of which are given in Table 6.4. The lower average run time and greater number of runs satisfying Tier A criteria formed the basis for the switch to a local replacement method.

Table 6.4: Results from tests comparing GA generation replacement methods, $n = 19$. Bands A – C relate to the convergence time, incorporating the solutions of the threshold tests refinement.

Parameter	**End**	**Local**
Average Time	39.41 s	26.61 s
Standard Deviation	15.25 s	18.80 s
Number in Tier A	7	13
Number in Tier B	0	0
Number in Tier C	12	6

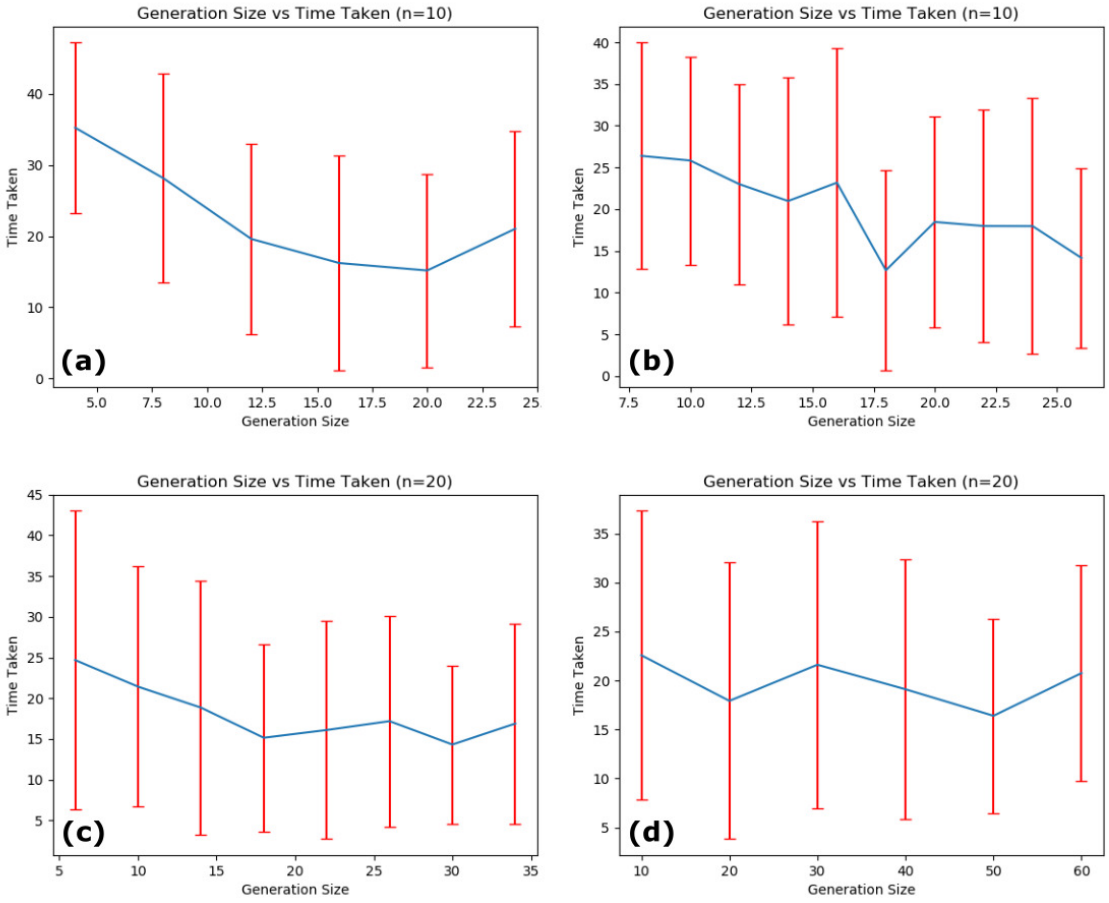

Figure 6.21: Results from a series of tests comparing the effect of generation sizes on the performance of the GA, the blue line representing the average time taken in seconds and the red error bars representing the maximum and minimum data values for those tests.

6.6.4.4 Generation Size

The final significant change made to the GA involved testing to find the best generation size for a typical routing problem around Fuego. The data in Figure 6.21 are presented chronologically and

make use of the threshold function and replacement method results described above. The space was first explored in the range $[4,24]$ with an interval of 4 and $n = 10$ (Figure 6.21 (a)), and this suggested a minimum average time between 12 and 24. The next test therefore further explored the range $[8,26]$ using an interval of 2 and $n = 10$ (Figure 6.21 (b)), however gave inconclusive results. The general trend for these data was a decrease with larger generation sizes, however anomalies at sizes of 16 and 18 suggested further tests were needed with a higher sample number.

The next two tests therefore took 20 runs at each generations size. Figure 6.21 (c) shows data covering the range $[6,34]$ with an interval of 4, and suggests a plateau of results from generation sizes ≥ 18. This is broadly consistent with the findings of previous tests, so a final test took place to try to find the upper bound (Figure 6.21 (d)). This final test explored the range $[10,60]$ with an interval of 10, and showed that generation sizes of 20 and 50 gave the best results. A generation size of 20 was therefore chosen, given the consistent data across all four tests; they all average below 20 seconds with an average maximum run time of approximately 30 seconds.

6.6.5 Climb Behaviour

The climb behaviour witnessed on RTTP flights during Trip E had a negative impact on mission performance, and is a result of the climb controller used by the autopilot when in GUIDED mode. Without modifying the autopilot code, the proposed solution to this involves tuning the ArduPlane TECS climb parameters such that the maximum climb rate and angle are limited to appropriate values for the RTTP. The TECS maximum climb rate parameter was set to 5 m/s throughout Trip E, however climb rates of up to 7.5 m/s were recorded in GUIDED mode. Flight testing is therefore required in order to tune the TECS parameters for appropriate GUIDED mode climb behaviour, and failing that the autopilot code may need to be modified.

Testing the effectiveness of these proposed solutions is beyond the scope of this work, however a solution must be found before any further comparison between the RTTP and conventional flight plans.

6.7 Refinement Results & Discussion

The revised GCS Update & Command console is pictured and annotated in Figure 6.22 and Figure 6.23, for two different RTTP route cases. Figure 6.22 shows the aircraft preparing to follow an RTTP route by climbing to the altitude of the first waypoint. The annotations focus on the text update box; note that some messages are still repeated however this no longer happens consecutively. This means that the update space is less cluttered and the progress of the script is easier to follow in real-time.

Figure 6.23 shows another case where the RTTP is in use, however in this case the aircraft is shown following a route. Note, the status update box has several repeated messages, including the RTTP route. The purple triangle in the map and altitude plot represents the UAV position

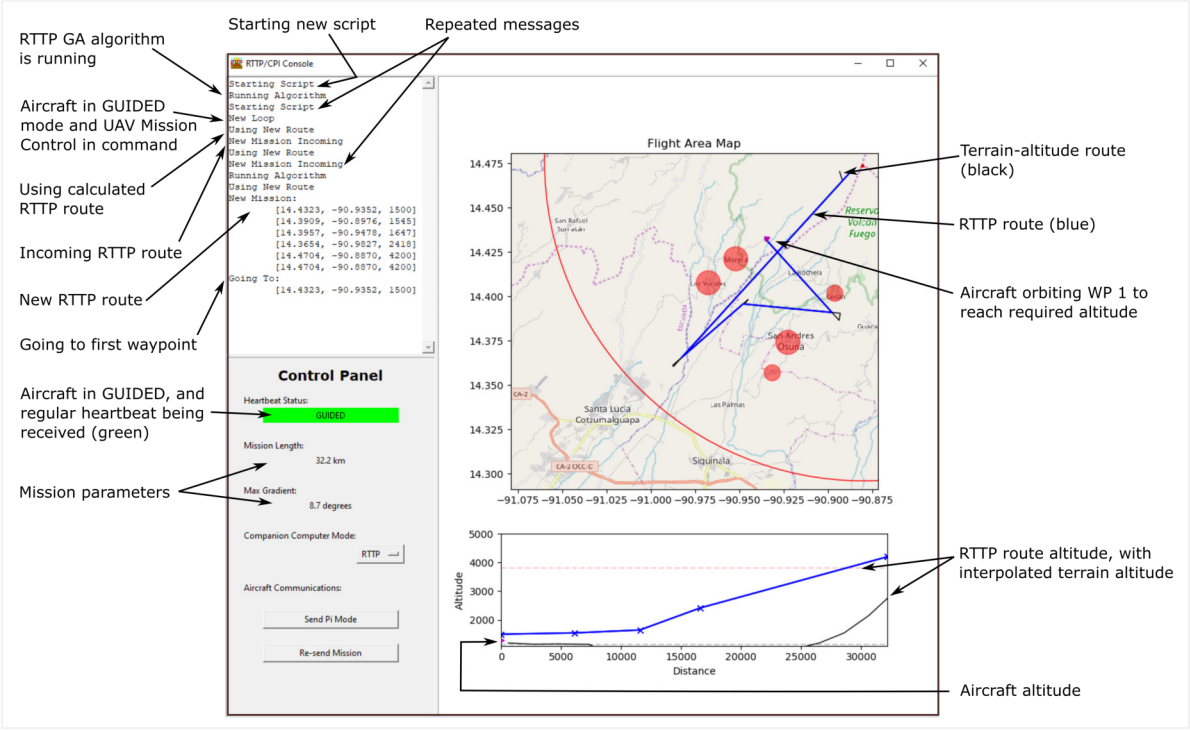

Figure 6.22: Annotated figure of the GCS Update & Command console, showing the aircraft preparing to follow an RTTP route in SITL simulation. Map source: Open Street Map [12].

and altitude respectively. The position of the aircraft marker on the altitude plot shows it starting to climb above the flight path line, recreating the poor climb performance issues seen in missions on Trip E.

Both Figures 6.22 and 6.23 represent significant improvements in the quantity and quality of information available to GCS operators, made possible by redesigning the GCS Update & Command program and the underlying MAVLink communications. Perhaps the most significant change made to the GCS program is the visualisation of the route waypoints and altitudes, however plotting the updated aircraft position and showing key mission parameters also provides useful context to the route. The revised use of the MAVLink communications protocol appears to have made message sending and receiving more robust, using less time by decreasing the number of messages sent and received unnecessarily.

The changes made to the RTTP GA since Trip E have decreased the amount of time spent per iteration. This has in turn meant the algorithm produces relatively low-cost solutions in a realistic real-world run time. In order to compare the effects of these changes, the revised GA must be run on a Raspberry Pi 3B+ in order to simulate the configuration used for the RTTP tests on Trip E. The GA was therefore run 100 times on the Raspberry Pi 3B+; the averaged timing and iteration data are given in Table 6.5 and the threshold cost parameters are given in Table 6.6 with the number of runs that were solved in each of the tiers.

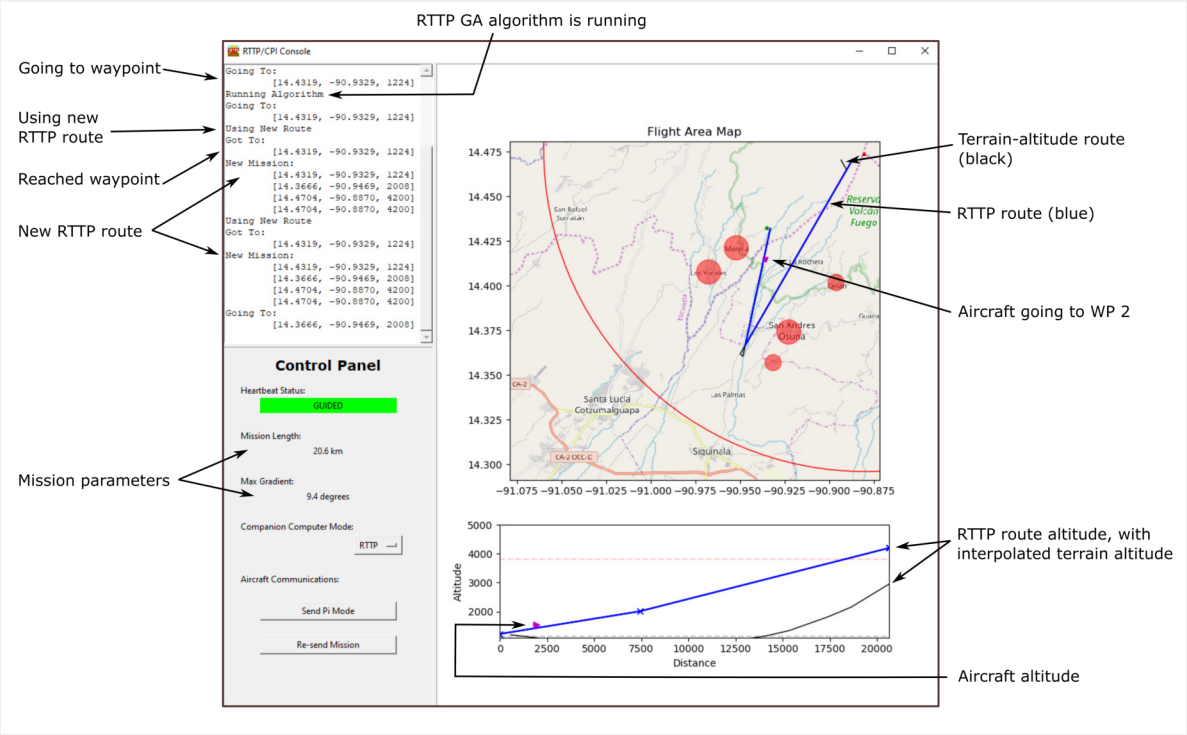

Figure 6.23: Annotated figure of the GCS Update & Command console, showing the aircraft following an RTTP route in SITL simulation. Map source: Open Street Map [12].

The four RTTP runs on Trip E had an average time per iteration of 2.78 s and a standard deviation of 0.90 s, spanning two cases with obstacles and two cases without. The revised data given in Table 6.5 ($n = 100$) indicates a 75% reduction in average time per iteration since Trip E, with a lower standard deviation and more consistent run times. The runs of the revised GA all included the obstacles pictured in Figure 6.16, reinforcing the positive effect of these changes.

Table 6.6 gives data regarding the threshold function for the testing of the revised GA. Note that the threshold function for Trip E was 1.5, and comparatively just 2% of the solutions reached Tier C. In comparison, 53% of the runs converged with a cost of 1.10, satisfying the Tier A threshold, and 45% of the runs converged with a cost of 1.20, satisfying the Tier B threshold. The revised version of the GA is evidently quicker, more reliable, and generates better solutions than

Table 6.5: Data comparing the RTTP GA runs from Trip E ($n = 4$) and the revised version ($n = 100$), both of which ran on a Raspberry Pi 3B+.

Parameter	**Trip E**		**Revised**	
	Mean	Std Dev	Mean	Std Dev
Averaged Time per Iteration (s)	2.78	0.90	0.70	0.19
Run Time (s)	90.86	74.71	28.80	13.23
Number of Iterations	32	27.18	44	22

the version used on Trip E.

Table 6.6: Threshold tier data from the revised GA tests run on the Raspberry Pi 3B+, $n = 100$.

Parameter	**Tier A**	**Tier B**	**Tier C**
Lower time limit	0 s	40 s	50 s
Upper time limit	40 s	50 s	Inf s
Threshold Cost	1.10	1.20	1.50
Number of Runs	53	45	2

6.8 Conclusions & Future work

This chapter has introduced and refined a method for onboard real-time trajectory planning for UAVs, demonstrating the usefulness of an increased level of autonomy. In this case, the systems onboard the aircraft are given the authority to produce and follow a trajectory, with human input required as a fail-safe rather than a positive action. The flight results presented here prove it is possible to use an RTTP to conduct UAV missions, and the refinements presented should build robustness and usability when flight tested.

Flight tests during Trip E generated routes comparable to conventional flight plans, which took significant time to manually develop. According to the data presented in Table 6.2, some efficiency gains can be expected by using the early version of the RTTP but these gains are not significant. Sections 6.6 and 6.7 describe further RTTP development and simulation results respectively, following Trip E and publication of the early work, showing marked improvements. Real-world tests are expected to show significant advances in efficiency compared to both Trip E and conventional mission plans.

The UAV Mission Control script and the GCS Update & Command Console have been presented as methods of controlling and monitoring the RTTP respectively, with significant development after Trip E for smooth and modular use. An important aspect of these revisions was the effective implementation of the MAVLlink communications protocol, centralising the methods and code used between the GCS and the UAV.

The automatic detection of plumes, and onboard real-time trajectory optimisation from GCS to summit, have both been addressed. The next stage in the automation of UAV-based ash sampling around Fuego involves combining the path planner framework and plume sensing ability to form a Coordinated Plume Interception (CPI) scheme. The CPI scheme must find an efficient path to reach the downwind location of the plume after sampling at a proximal location, given an operator-defined spatial or temporal sample interval.

CHAPTER 7

Coordinated Plume Interception

The ability to sample ash from within volcanic plumes opens up a series of formerly unavailable datasets. As previously discussed, a key parameter of interest is the Particle Size Distribution (PSD) of ash in the plume. The second generation ash collection mechanism (Chapter 4, Section 4.1.1) allows multiple ash samples to be collected in a single flight, making it possible to sample consecutive plumes proximally or sample the same plume at both proximal and distal locations. The latter case is of particular interest because the change in ash PSD with time (and, in the presence of wind, distance) is a key metric for ash dispersion models [52].

This chapter presents a method of automating plume re-interception, following the themes of autonomy and UAV operations around Volcán de Fuego in Guatemala. This scheme, named Coordinated Plume Interception (CPI), involves calculating a path in 2D space that starts at a proximal plume sampling location and ends at a downwind plume location in order to intercept and sample the plume at a comparatively distal point. The separation between sample locations may be as large as 15 or 20 km, depending on conditions and the permissible flight zone. This problem has two key requirements; accurately predicting the future location of the plume, and navigating to that location efficiently. Five CPI routing options are presented, including a conventional baseline method, and compared by estimated energy required and compatibility with the flight operations zone over a range of conditions.

A conference paper titled 'Automated Real-Time Volcanic Plume Interception for UAVs', written by the Author, has been accepted to the American Institute of Aeronautics and Astronautics SciTech 2021 conference. The work presented in this chapter was due to be flight tested in Guatemala in March 2020, however the COVID-19 pandemic effectively cancelled this trip. While designed and developed for real-world implementation, the results given in this chapter are therefore limited to simulation only.

7.1 Introduction

One of the common primary aims of a UAV mission around Fuego is to collect ash samples from within the plume. While flight from the observatory to the summit region is usually conducted in AUTO or GUIDED modes, to reduce pilot workload and increase efficiency, many of the most successful ash-collection flights to date have involved use of 'Fly-By-Wire A' (FBWA) mode. Flying the aircraft in this mode, using telemetry data and First Person View (FPV) video as a guide, means the pilot can coordinate plume interception manually, albeit with control augmentation from the autopilot (Figure 7.1).

Figure 7.1: The Author flying a Skywalker X8 through volcanic plumes in FBWA mode using FPV goggles from the Ground Control Station in Panimache, with the support of Matt Pervis (GeoTravel Guatemala) and a ground station crew (not pictured).

In FBWA mode the aircraft holds the wings level unless otherwise commanded by the pilot. Limits are implemented on angles and rates of pitch and roll, however the pilot has full control of the throttle setting between pre-set limits (usually 0% - 100%). It is possible to fly the UAV in MANUAL mode at the range and altitude considered here, however FBWA means pilot workload is reduced while maintaining adequate levels of navigational control for plume sampling.

Pilot involvement in UAS often decreases flight efficiency and allows human error into the system. Automating the plume interception process aims to maximise efficiency and predictability, while reducing the chance and impact of human errors. The repeatability of automated systems means that both mission and aircraft performance can be monitored accurately, making it easier to conduct analysis for system development.

The CPI scheme uses the same navigational control framework as the RTTP in Chapter 6, and has been developed alongside the refinements presented in Sections 6.6 and 6.7 of Chapter 6. Both the RTTP and CPI schemes are designed to run onboard the aircraft, so the modular structure of the UAV Mission Control program, MAVLink messaging protocol implementation, and GCS Update & Command program were developed accordingly for shared use.

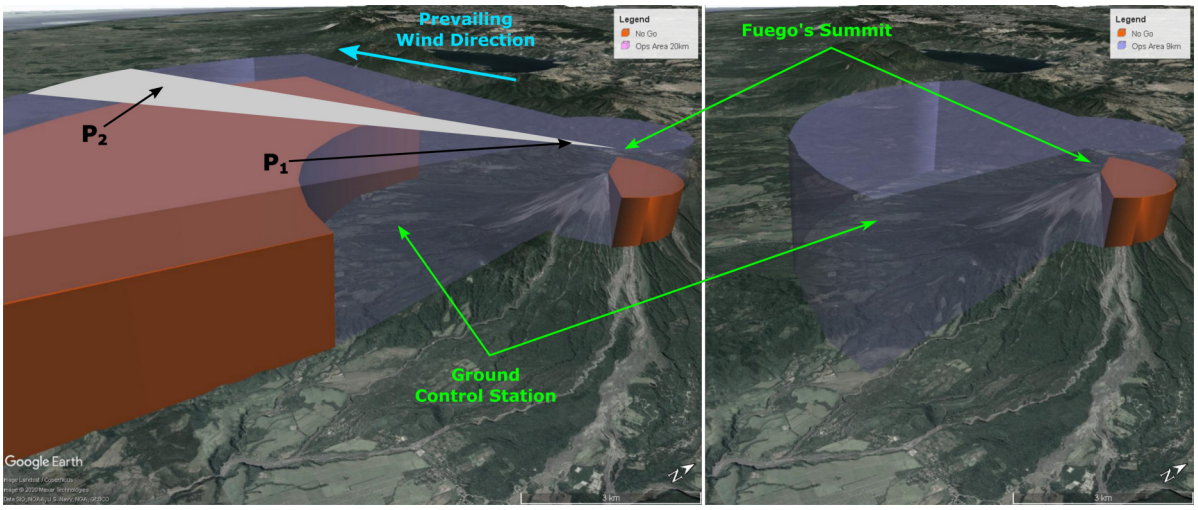

Figure 7.2: Left: Proposed CPI flight operations zone, with a radius of 20 km, showing the prevailing wind direction and consequent prevailing plume area (grey). P_1 and P_2 indicate possible plume sampling positions, between which the CPI must generate a route. Right: Previously used flight operations zone, with a radius of 9 km. UAS flight zones are indicated by purple sections, and No Go zones within the flight zone footprint are shown by orange blocks. The ground station and summit locations are annotated, and maximum flight zone altitude is 5000 m (approximately 18,000 ft) AMSL. Map source: Google Earth [10].

The flight operations area for previous trips has been defined as a 9 km radius cylinder, centred on Fuego with a ceiling of 5000 m (approximately 18,000 ft) AMSL. While it should be possible to use the CPI scheme in this space, higher quality scientific results are expected at longer ranges (e.g. 20 km). This means a new operations area must be negotiated for effective real-world testing; Figure 7.2 compares the proposed (left) and previously used (right) flight areas. Note the lower height limit of the extended areas, equal to Fuego's summit altitude. This is suggested as a safety device, building a buffer between the flight zone and the ground, and limiting the UAV altitude window for airspace management. Figure 7.2 also shows the prevailing wind direction, consequent plume area, and arbitrary plume sampling positions P_1 and P_2.

7.2 Methodology

7.2.1 Problem Development

The CPI problem can be broken down to depend on some key variables, described in Table 7.1. One approach to tracking the plume involves real-time onboard computer vision, however a less complex method is presented in this chapter. Rather than processing images in real-time, perhaps from a dual-camera set-up in order to gauge depth, the method presented here relies on calculating the wind speed and direction after sampling the plume at position P_1. The time at which the plume will be at distance $d_{P_1P_2}$ downwind can then be estimated, and an appropriate UAV path can be planned. The work in this chapter assumes a non-zero wind speed, that the plume does not descend after the initial eruption, that the wind speed is constant across the sample space at the plume altitude, and the plume remains in the permissible flight zone for the duration of the scheme.

Table 7.1: CPI variable definitions.

Variable	**Definition**
V_W	Wind speed, typically between 5 m/s and 15 m/s
θ_W	Wind direction, measured clockwise from North
P_1	Plume sampling position 1, in the inertial frame
P_2	Plume sampling position 2, in the inertial frame
$P_{1,2}$	Plume position, in the wind frame
$d_{P_1P_2}$	Distance between P_1 and P_2, experimentally varied between 10 km and 20 km
$t_{P_1P_2}$	Time taken for plume to travel from P_1 to P_2
V_a	UAV airspeed, considered fixed here (25 m/s)
d_{path}	UAV path length, in the wind frame

The time taken for the plume to travel between position P_1 and P_2 is given by Equation 7.1 and is a key design variable of the CPI route. P_2 is assumed to be inside the flight operations area.

$$t_{P_1P_2} = \frac{d_{P_1P_2}}{V_W} \tag{7.1}$$

This problem is analogous to spending time in a holding pattern, with the added constraint of a final location at a given time; the UAV is required to wait while the plume moves downwind before arriving at point P_2 when $t_{P_1P_2}$ has elapsed. In the inertial (i.e. ground) frame, the UAV path must start at $t = t_0$, position P_1, and end at $t = t_{P_1P_2}$, position P_2. This problem can be simplified by considering it in the wind frame, because the plume then remains stationary at $P_{1,2}$.

While aircraft usually navigate with respect to the inertial frame, between locations defined by latitude and longitude, their movement can be broken down into the relative speed of the aircraft through the air (i.e. wind frame) and the movement of the air/wind frame relative to the

inertial frame. Fixed-wing aircraft such as the Skywalker UAVs used around Fuego rely on a forward airspeed V_a, a velocity that naturally relates to the wind frame.

The distance travelled through the air, or air-distance, can be found by integrating V_a over time. The CPI route length in the wind frame can therefore be defined by Equation 7.2:

$$d_{path} = V_a \times t_{P_1P_2} \tag{7.2}$$

The distance required for the UAV to fly in the wind frame for a given distance $d_{P_1P_2}$ is related to the wind speed, as shown in Figure 7.3. Slower UAV airspeeds naturally mean less distance must be flown, however do not change $t_{P_1P_2}$. Airspeed is best chosen according to the flight performance characteristics of the aircraft, and is considered constant in this method. This means the UAV air-distance (d_{path}, in the wind frame) is fixed by the wind speed V_W and operator-chosen plume sampling separation $d_{P_1P_2}$.

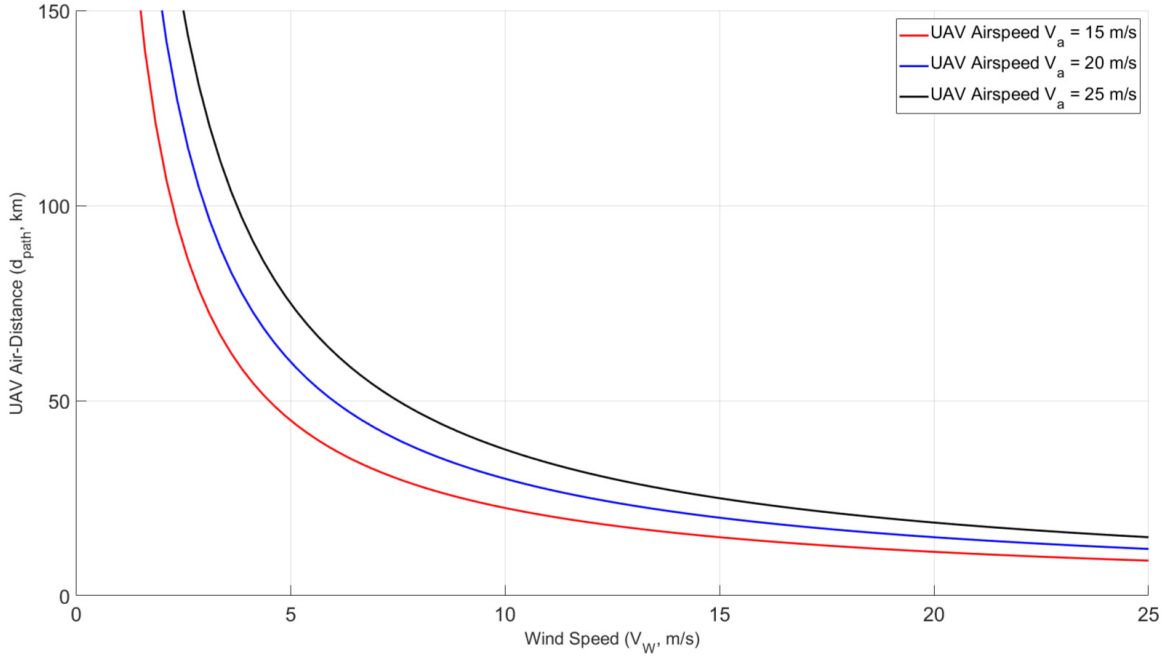

Figure 7.3: Graph showing the relationship between wind speed and UAV air distance for a CPI case with $d_{P_1P_2}$ =15 km and three different UAV airspeeds.

The criteria for a CPI route is therefore that it travels distance d_{path} in the wind axes. A key focus of route generation is to minimise efficiency losses, however practical factors such as staying in the prescribed airspace also impact route design. It is hypothesised that the efficiency of these routes will be affected by time spent in rolling manoeuvres, given that additional lift must be generated for non-zero roll angles (proportional to $cos(\phi)$ where ϕ is the aircraft roll angle).

Approaching the CPI problem in the wind frame simplifies certain aspects of route generation and, once formulated, these routes require transformation into the inertial frame so that they can be followed by conventional earth fixed navigational systems.

The transformation of a single point from the wind frame into the inertial frame is strictly time dependent. If a point in the wind frame starts at coordinates $[0,0]$, the change in position in the inertial frame after time t_{wp} is given by Equation 7.3. This equation applies for a typical wind direction around Fuego of $\theta_W = [0,90]$.

$$\begin{bmatrix} \Delta X_{wp} \\ \Delta Y_{wp} \end{bmatrix} = -t_{wp} V_W \begin{bmatrix} sin\theta_W \\ cos\theta_W \end{bmatrix} \tag{7.3}$$

This can therefore be applied to find the change in inertial coordinates of an object travelling with the wind after time t_{wp} has elapsed. If the point or object has been at $[X_0, Y_0]$ in the inertial frame at $t = t_0$, then its position after t_{wp} is given by Equation 7.4.

$$\begin{bmatrix} X_{wp} \\ Y_{wp} \end{bmatrix} = \begin{bmatrix} X_0 \\ Y_0 \end{bmatrix} + \begin{bmatrix} \Delta X_{wp} \\ \Delta Y_{wp} \end{bmatrix} \tag{7.4}$$

A route is defined by a series of waypoints, and so may be transformed from the wind frame to the inertial frame by subjecting each waypoint to a transformation as described above. This is possible and relevant providing that the aircraft starts at the beginning of the route at t_0 and the time taken to travel between waypoints is known (i.e. Equation 7.1).

7.2.2 Route Options

This section introduces the five approaches considered in this chapter. Figure 7.4 shows diagrams of these approaches, drawn in the inertial and wind frames on the left and right respectively. There follows a brief summary of each route, followed by an assessment of the predicted compatibility between route types, wind conditions, and flight zone limits. Practical use of the routes is discussed in Section 7.2.5, including wind estimation and the impact of flying wind-frame generated routes using a traditional inertial frame navigational controller. Modelling and simulation results for these routes are presented and discussed in Section 7.3.

Option A

Routing option A is the simplest in terms of real-world implementation and involves flying straight to position P_2 and loitering until the plume arrives. This route is inherently based in the inertial frame, treating point P_2 as a conventional waypoint, and is guaranteed to stay within the permissible flight zone given a viable secondary plume sampling position.

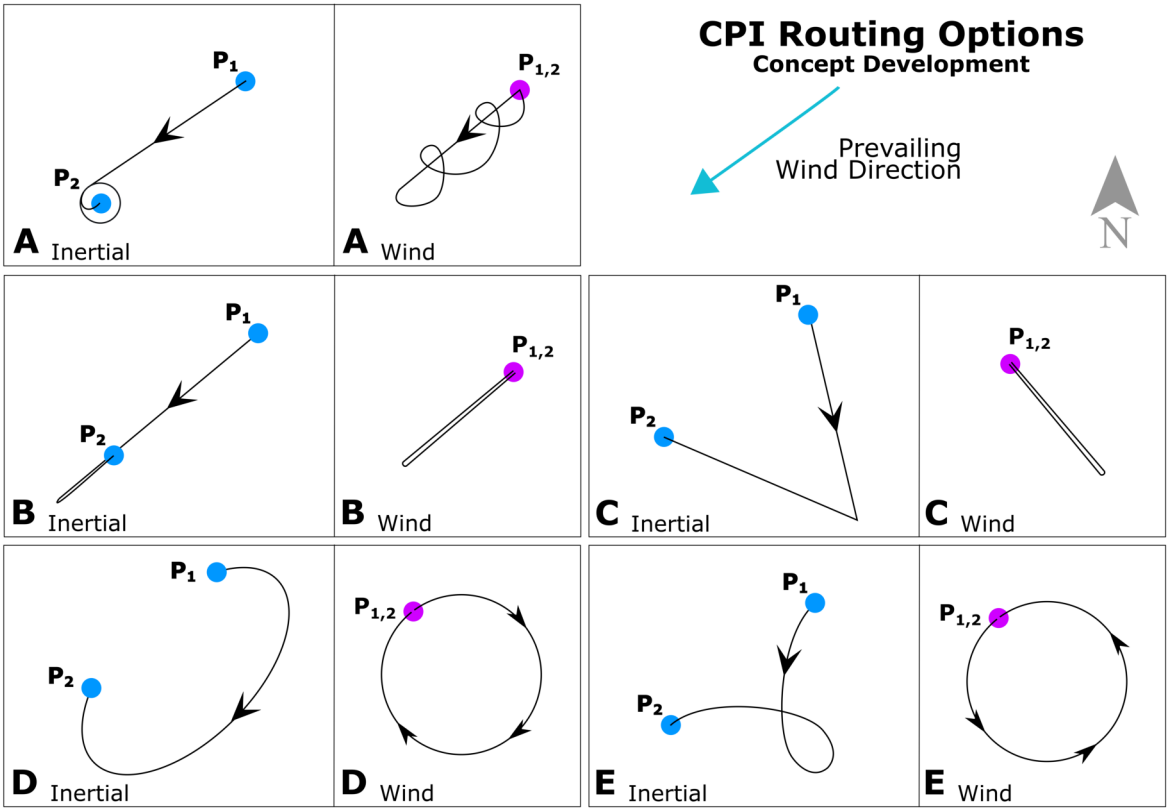

Figure 7.4: The five options considered for CPI, shown in both the inertial (earth) frame and the wind frame. P_1 is the first plume sampling position, and P_2 is the second plume sampling position. Note the prevailing wind direction.

Option B

This option involves flying a straight leg downwind to a turn waypoint, then turning 180° and flying upwind to P_2. This is the first of the five route options that is easiest to develop in the wind axes; the turn waypoint is a distance $\frac{d_{path}}{2}$ downwind in the wind frame. It is also the point of greatest separation from the plume ($P_{1,2}$) which, when converted into inertial coordinates, may fall outside the permissible flight zone radius.

Option C

Route type C is similar in formulation to route B, but involves flying perpendicular to the wind direction rather than parallel to it. This route is best considered first in the wind frame before transformation into the inertial frame, with leg lengths in the wind frame equal to $\frac{d_{path}}{2}$. The direction of the first leg can be ±90° to the wind direction, depending on the conditions and permissible flight zone (Figure 7.5).

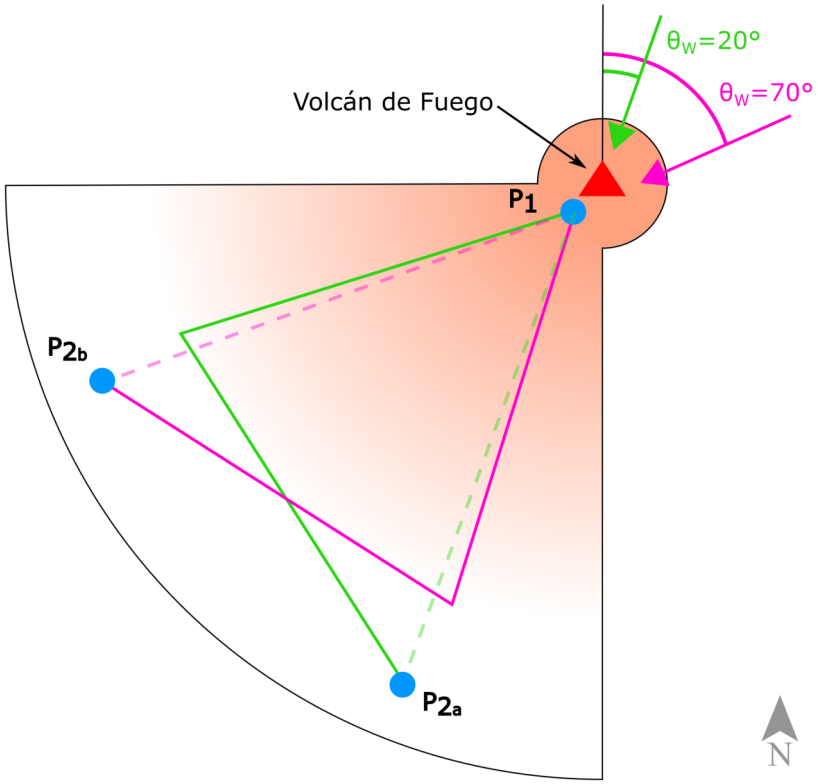

Figure 7.5: Two versions of CPI routing option C in the inertial frame, limited by the quarter-circle flight area shown, for $\theta_W = 20°$ (green) and $\theta_W = 70°$ (pink). Position P_1 is the initial sample location, and positions P_{2a} and P_{2b} represent the secondary sample positions.

Option D

This is the first of two CPI route types that involve flying a circle in the wind axes, and is characterised by starting and ending the route by flying upwind (Figure 7.6, Left). The centre of the circle is ±90° to the wind direction at a distance d_r from plume point $P_{1,2}$, and is given by Equation 7.6.

$$d_r = \frac{d_{path}}{2\pi} \tag{7.5}$$

$$d_r = \frac{d_{P_1P_2}V_a}{2\pi V_W} \tag{7.6}$$

The validity of route option D with respect to the permissible flight zone is compared to the other route options in Section 7.2.3 below; the initial upwind phase of flight is expected to strictly limit the conditions for which this route is viable, given the relatively small flight zone around Fuego.

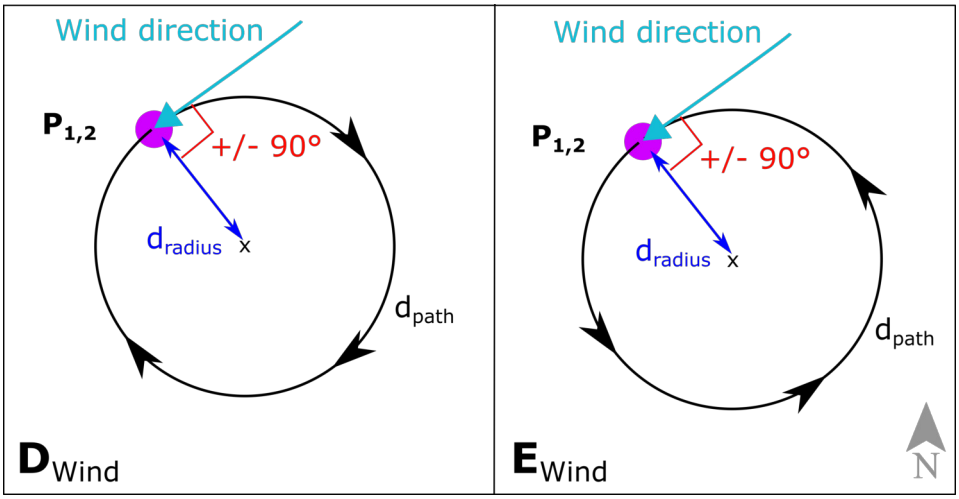

Figure 7.6: Circular-based CPI routing options D (left) and E (right). Note that the angle between the wind direction and vector between the circle centre and point $P_{1,2}$ can be $\pm 90°$ for both cases.

Option E

Finally, route option E also involves flying a circle in the wind axes, however it starts and ends the route by flying downwind (Figure 7.6, Right). The centre of the circle is at the same position as route D, and can also be at $\pm 90°$ to θ_W. This is likely to be one of the most efficient options given that it flies just one loop; the circle radius is large and the consequent loss of lift due to roll should be minimal.

7.2.3 Flight Zone Compatibility

As introduced at the start of this chapter, the flight zone considered for this work is defined by a quarter-circle shape with the circle's centre on Fuego and a radius of 20 km, with an additional circular flight zone (2 km radius) around the summit. The limits of the quarter circle are an East-West line and a North-South line drawn through Fuego's summit; this allows plume sampling up to a limit of $d_{P_1P_2} = 20$ km with wind directions between 0° and 90°.

Dictating the distance distance $d_{P_1P_2}$ limits the viability of each route, depending on wind speed (V_W) and wind direction (θ_W). Considering the wind frame first, the separation between the UAV and plume point $P_{1,2}$ can be split into longitudinal and lateral separation. These are defined, relative to the wind direction, in Figure 7.7. Table 7.2 gives a summary of the maximum longitudinal and lateral separation values for route options B - E. Route A is not considered in this section because, assuming that the plume stays within the flight zone and the distance between P_2 and the zone perimeter is greater than the loiter radius, route A is guaranteed not to breach the perimeter.

Converting routes from the wind frame to the inertial frame depends on the wind conditions and the time interval over which the transformation is applied. A polar approach may be

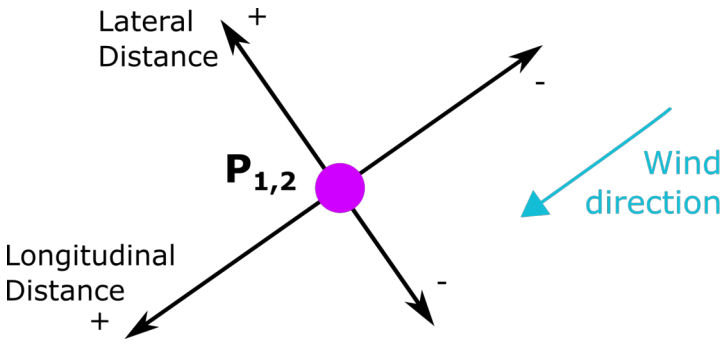

Figure 7.7: UAV separation from the plume can be described in the wind frame in terms of longitudinal and lateral separation from the plume position, $P_{1,2}$. Positive longitudinal separation is defined as distance in the downwind direction, and positive lateral separation is defined in the direction 90° clockwise of downwind.

Table 7.2: Maximum separation distances between the UAV and plume in the wind frame, as defined by Figure 7.7.

Route Option	**Longitudinal**	**Lateral**
A	-	-
B	$\frac{d_{path}}{2}$	-
C	-	$\pm\frac{d_{path}}{2}$
D	$\pm\frac{d_{path}}{2\pi}$	$\pm\frac{d_{path}}{\pi}$
E	-	$\pm\frac{d_{path}}{\pi}$

taken to consider the breaching of flight zone limits; they can either extend radially such that the maximum distance from point P_1 is beyond the quarter-circle limit radius (hereafter labelled R_{max}, nominally 20 km), or they can rotate angularly such that the flight path crosses the horizontal or vertical limits. The small circle around Fuego's summit is denoted by r_{max}, considered here with a value of 2 km.

Each route has between one and three distinct points which threaten breaching the flight zone either radially or rotationally, given certain conditions. The remainder of this section consider each route in turn, forming relationships between V_W, θ_W, and $d_{P_1P_2}$ in the inertial frame.

The grey areas in the associated plots represent valid values for $d_{P_1P_2}$, given V_W and θ_W. Zone limits are given by red lines, radial constraints are represented by blue lines, and rotational constraints are represented by pink lines. The flight zone is symmetric about a line angled at 45° that passes through the circles' center (Fuego's summit), so θ_W is only considered in the range $[0, 45]$°. Should it exceed this range in reality then a mirrored mission plan would be used (i.e. switch from negative to positive lateral separation in the wind frame) and the same restrictions would apply in the opposite direction.

Note that the following section only strictly applies when the point P_1 is the center point of the flight zone circles, i.e. directly above Fuego's crater. However, this section remains applicable

as a guide for the majority of missions which seek to compare ash PSD between proximal and distal locations. This is because proximal ash samples are targeted immediately after eruption and therefore the distance between P_1 and Fuego's crater is typically small.

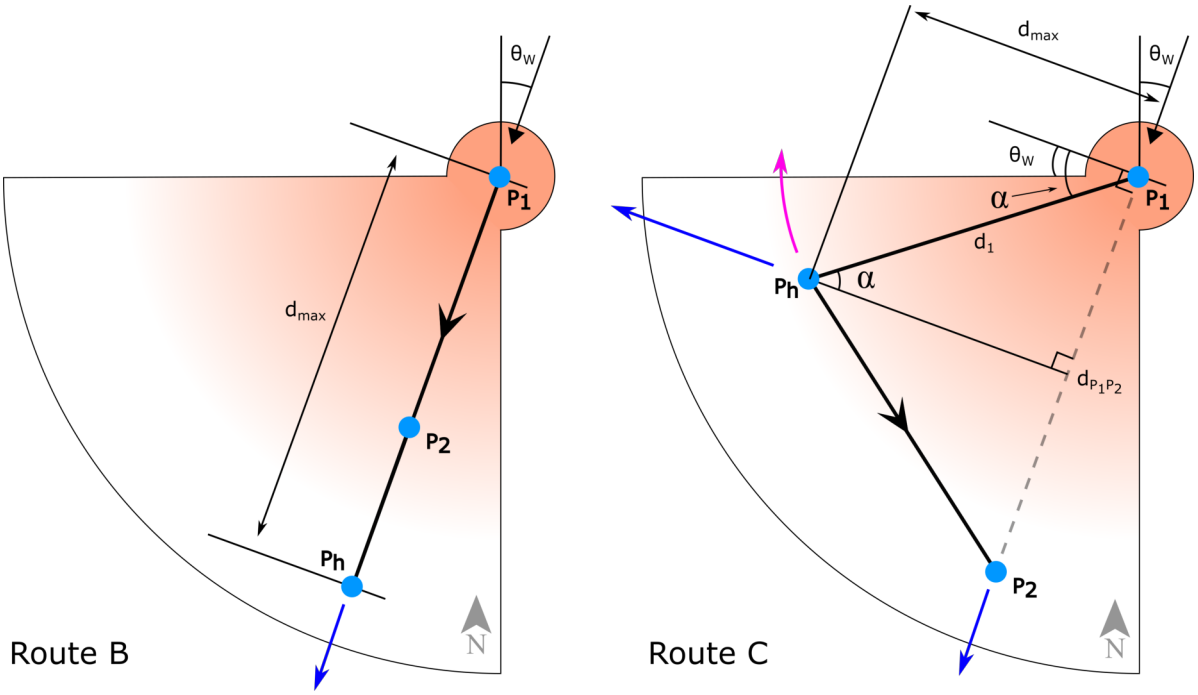

Figure 7.8: Diagram showing flight zones, routes, and key geometric labels for route options B (left) and C (right). Position P_1 is the first plume sampling location and nominally located over Fuego's summit. Position P_2 is the second plume sampling location, a distance $d_{P_1P_2}$ downwind. Blue arrows indicate radial breach points, and pink arrows indicate rotational breach points.

Option B

Route B has no lateral separation from the plume, however extends up to $\frac{d_{path}}{2}$ in the positive longitudinal sense (wind frame). A relationship can be developed between the inertial distance $d_{P_1P_2}$ and the length of the first leg d_{max} by equating the time taken for the UAV and plume to reach the halfway position P_h. The ground speed of the aircraft over this first leg is given by $V_{g_1} = V_a + V_W$, so considering the limit case:

$$\frac{t_{P_1P_2}}{2} = t_{P_h} \tag{7.7}$$

$$\frac{d_{P_1P_2}}{2V_W} = \frac{d_{max}}{V_{g_1}} \tag{7.8}$$

$$d_{P_1P_2} = d_{max}\left(\frac{2V_W}{V_a + V_W}\right) \tag{7.9}$$

The relationship described by Equation 7.9 is plotted by the blue line in Figure 7.9, and the radial limit $R_{max} = 20$ km is shown by the red line. The grey areas in these two plots indicate the allowable values for $d_{P_1P_2}$ given a range of wind speeds (V_W, left) and wind directions (θ_W, right). The plot on the right of is included here for completeness; route B is limited only in the radial sense and is viable for all wind directions between 0° and 90°. This route is best suited to high wind cases; applying Equation 7.9 for typical conditions (i.e. $V_W = 10$ m/s) yields a practical limit of $d_{P_1P_2} = 10$ km which is lower than typically desired.

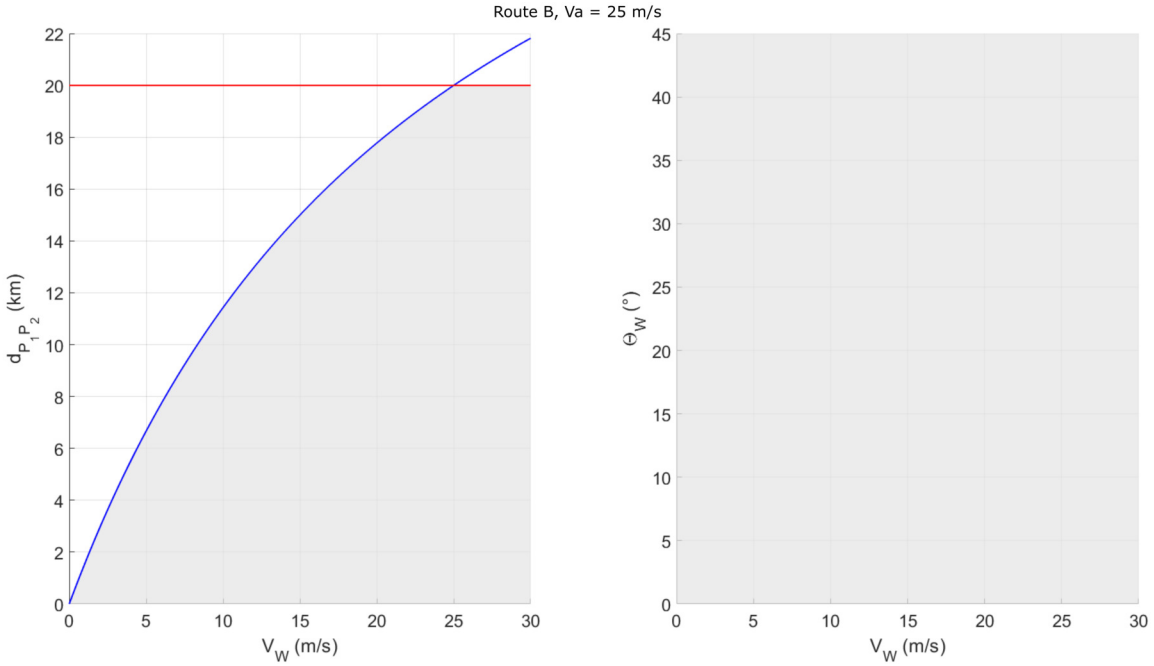

Figure 7.9: Plots showing allowable values of $d_{P_1P_2}$ for route B, given a range of wind speeds (V_W, left) and wind directions (θ_W, right). For a hypothetical case where $V_W > V_a$ the aircraft would fly to a position upwind of P_2 before turning into the wind and appearing to fly backwards in order to arrive at P_2 at the proper time.

Option C

The ways route C might breach the flight zone boundaries are shown by the blue and pink arrows in Figure 7.8, representing radial and rotational breaches respectively. Position P_2 is naturally limited by the distance $d_{P_1P_2}$, and position P_h threatens breaching the zone both radially and rotationally. Relationships for the radial and rotational limits on P_h are given in Equations 7.13 and 7.14 respectively. This route is defined by flying at 90° to the wind vector so:

$$tan(\alpha) = \frac{V_W}{V_a} \tag{7.10}$$

And geometrically:

$$tan(\alpha) = \frac{d_{P_1P_2}}{2} \div d_{max} \tag{7.11}$$

$$tan(\alpha) = \frac{d_{P_1P_2}}{2d_{max}} \tag{7.12}$$

Equating these expressions:

$$d_{P_1P_2} = d_{max}\left(\frac{2V_W}{V_a}\right) \tag{7.13}$$

The rotational limit is reached when $\theta_W = \alpha$, so in the limit case:

$$\theta_W = \tan^{-1}\left(\frac{V_W}{V_a}\right) \tag{7.14}$$

These relationships are plotted in Figure 7.10. By comparison to route B, using the nominal V_W = 10 m/s case, this route has a practical limit for $d_{P_1P_2}$ of approximately 15 km but also imposes rotational limits; the wind bearing must lie between $[0, 22]°$. The symmetry factor means that a wind direction between $[68, 90]°$ is also acceptable.

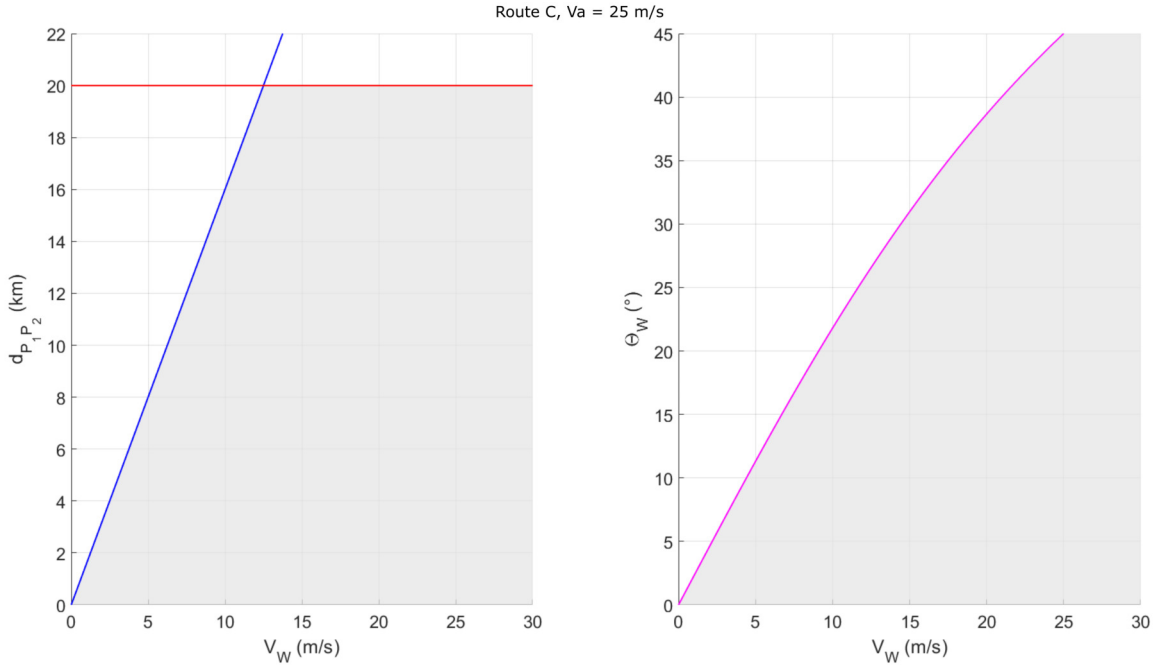

Figure 7.10: Plots showing allowable values of $d_{P_1P_2}$ for route C, given a range of wind speeds (V_W, left) and wind directions (θ_W, right).

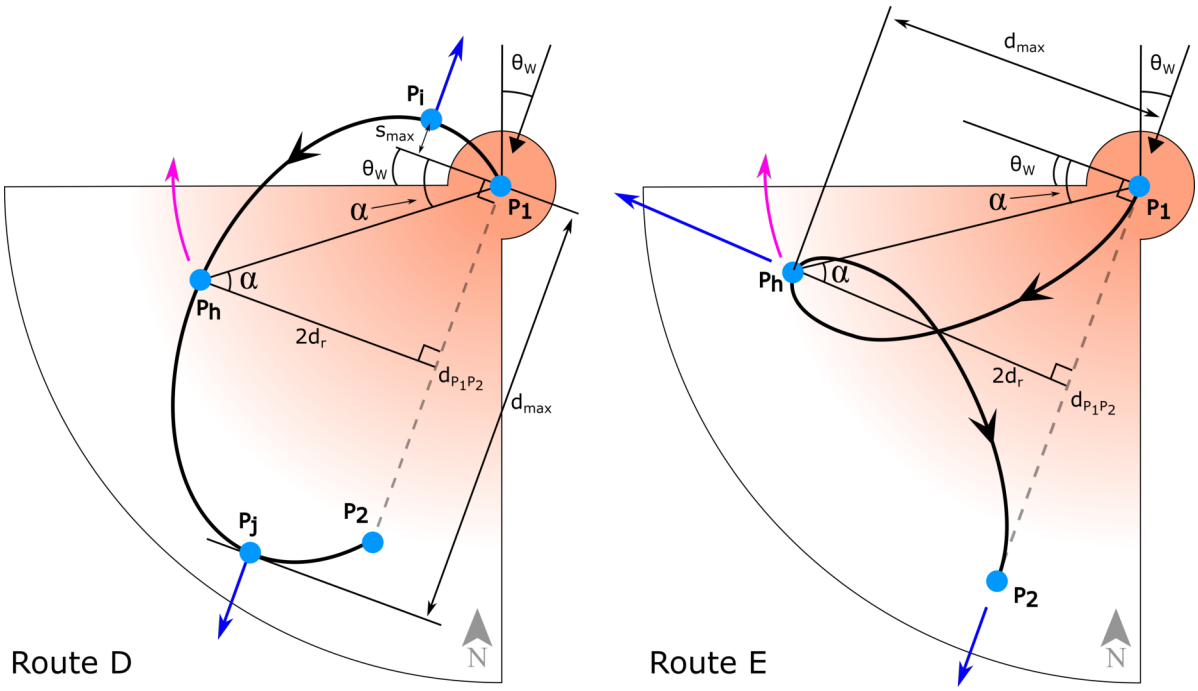

Figure 7.11: Diagram showing flight zones, routes, and key geometric labels for route options D (left) and E (right). Position P_1 is the first plume sampling location and nominally located over Fuego's summit. Position P_2 is the second plume sampling location, a distance $d_{P_1P_2}$ downwind. Blue arrows indicate radial breach points, and pink arrows indicate rotational breach points.

Option D

Route D is characterised by a looping path which starts and ends flying upwind. The blue arrows in Figure 7.11 (left) show that there are two types of radial breach, and the pink arrow approximates the rotational breach case. Position P_j is reached in time $\frac{3}{4}t_{P_1P_2}$ and, before transformation into inertial axes, has a longitudinal separation from point $P_{1,2}$ equal to the circle radius d_r (Equation 7.6). The inertial position of P_j can therefore be described as:

$$d_{max} = d_r + \left(\frac{3}{4}t_{P_1P_2} \times V_W\right) \tag{7.15}$$

$$d_{max} = \frac{d_{P_1P_2}V_a}{2\pi V_W} + \frac{3d_{P_1P_2}}{4V_W}V_W \tag{7.16}$$

$$d_{max} = d_{P_1P_2}\left(\frac{2V_a + 3\pi V_W}{4\pi V_W}\right) \tag{7.17}$$

So the sample separation is limited by:

$$d_{P_1P_2} = d_{max}\left(\frac{4\pi V_W}{2V_a + 3\pi V_W}\right) \tag{7.18}$$

Similarly, position P_i represents the point reached in time $\frac{1}{4}t_{P_1P_2}$, and is considered here relative to the boundary of the small circle (radius r_{max}). The upwind distance between points P_1 and P_i is labelled s_{max}:

$$s_{max} = d_r - \left(\frac{1}{4}t_{P_1P_2} \times V_W\right) \tag{7.19}$$

$$s_{max} = \frac{d_{P_1P_2}V_a}{2\pi V_W} - \frac{d_{P_1P_2}}{4V_W}V_W \tag{7.20}$$

$$s_{max} = d_{P_1P_2}\left(\frac{2V_a - \pi V_W}{4\pi V_W}\right) \tag{7.21}$$

So the sample separation is limited by:

$$d_{P_1P_2} = s_{max}\left(\frac{4\pi V_W}{2V_a - \pi V_W}\right) \tag{7.22}$$

The singularity in Equation 7.22, when $V_W = \frac{2V_a}{\pi}$, signifies the wind speed at which $s_{max} = 0$. Beyond this point the limit imposed by the small circle on point P_i can be ignored because the path will immediately track downwind (i.e. $s_{max} < 0$). The limit imposed by r_{max} is highly restrictive, and means any radial interaction between the flight path and the East-West line limit can be ignored for these preliminary developments.

The rotational limit for route D is approximated here by considering point P_h, the halfway position of the aircraft. As θ_W increases, a point between P_i and P_h will be the first to breach the flight zone limit, however the precise location of this position is not considered here due to its variation with conditions and the likely infeasibility of this route option. The rotational limit for point P_h occurs once again when $\theta_W = \alpha$, however for this route α is given by:

$$\tan(\alpha) = \frac{d_{P_1P_2}}{2} \div 2d_r \tag{7.23}$$

$$\tan(\alpha) = \frac{d_{P_1P_2}}{2} \times \frac{2\pi V_W}{2d_{P_1P_2}V_a} \tag{7.24}$$

$$\tan(\alpha) = \frac{\pi V_W}{2V_a} \tag{7.25}$$

So in the limit case:

$$\theta_W = \tan^{-1}\left(\frac{\pi V_W}{2V_a}\right) \tag{7.26}$$

These relationships are plotted in Figure 7.12, where the dashed lines represent the limits imposed by r_{max} (red) and s_{max} (blue). While $V_W < \frac{2V_a}{\pi}$ the main limit to $d_{P_1P_2}$ is r_{max} (2 km), however when $V_W > \frac{2V_a}{\pi}$ and $s_{max} < 0$ the limiting factor becomes R_{max} on position P_j. The nominal $V_W = 10$ m/s test case yields an allowable maximum distance $d_{P_1P_2} = 2$ km and wind direction $\theta_W = [0,32]°,[58,90]°$.

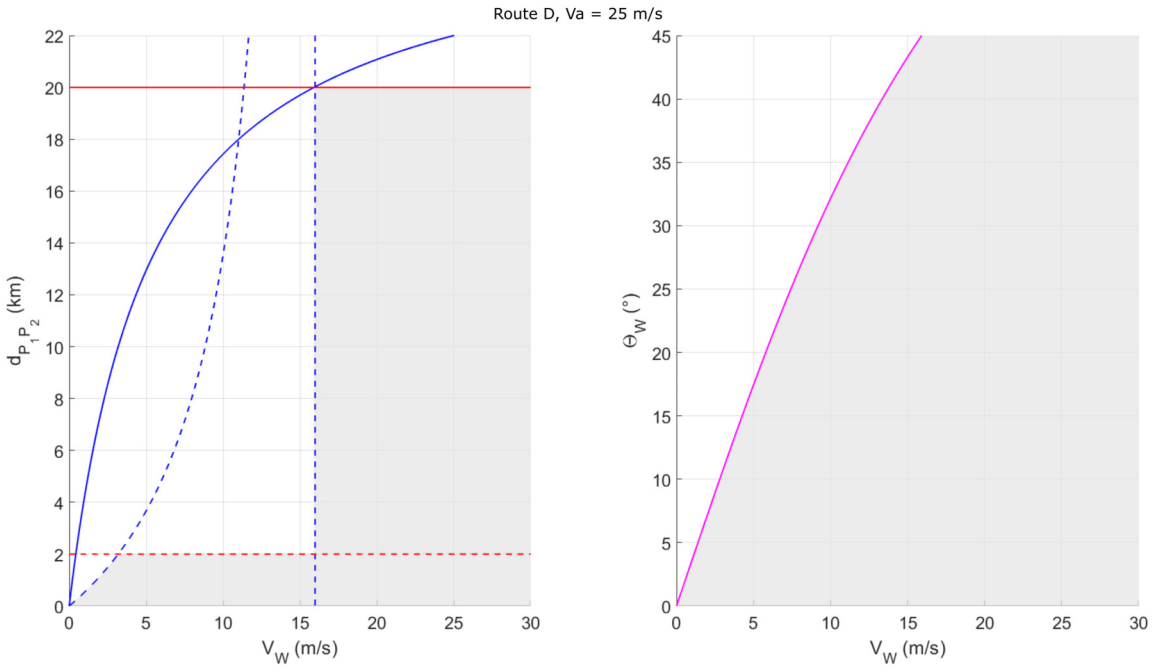

Figure 7.12: Plots showing allowable values of $d_{P_1P_2}$ for route D, given a range of wind speeds (V_W, left) and wind directions (θ_W, right). Dashed lines indicate the limits associated with r_{max} (2 km, red) and s_{max} (blue). When $V_W > \frac{2V_a}{\pi}$ the limit for $d_{P_1P_2}$ becomes R_{max}, set here as 20 km.

Option E

This route is characterised by flying a circle starting and ending in a downwind direction, opposite to route D. In addition to the natural limit imposed on P_2 by R_{max}, point P_h can breach the flight zone both radially and rotationally (Figure 7.11, right). The radial limit case can be written as:

$$d_{max} = 2d_r \tag{7.27}$$

$$d_{max} = 2 \times \frac{d_{P_1P_2} V_a}{2\pi V_W} \tag{7.28}$$

So:

$$d_{P_1P_2} = d_{max}\left(\frac{\pi V_W}{V_a}\right) \tag{7.29}$$

The rotational limit case for point P_h is the same for routes D and E, described by Equation 7.26. Once again, this limit is an approximation because it considers P_h as the first position to breach the flight zone. In reality this is not the case however, unlike route D, the true point of rotational breach is likely to be very close to P_h. This approximation is therefore acceptable for the purposes of comparison. The relationships between V_W, θ_W, and $d_{P_1P_2}$ are plotted in Figure 7.13.

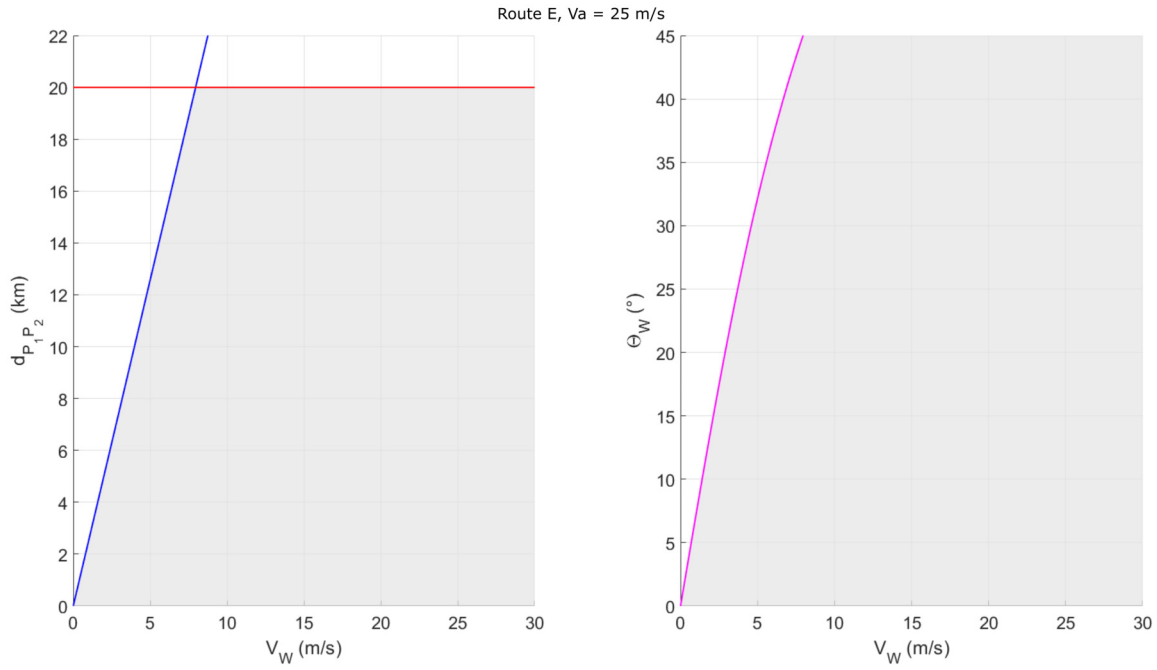

Figure 7.13: Plots showing allowable values of $d_{P_1P_2}$ for route E, given a range of wind speeds (V_W, left) and wind directions (θ_W, right).

For the nominal case where $V_W = 10$ m/s, route E is limited radially by the flight zone limit R_{max} (here set to 20 km) and rotationally in the range $\theta_W = [0,32]°,[58,90]°$.

7.2.4 Efficiency Modelling

The efficiency of each route may be modelled based on the amount of time spent turning, and the magnitude of the associated bank angles [164]. The lift force (L) in straight and level flight is equal to the weight of the aircraft, $W = mg$, where m is the mass of the aircraft in kg and g is acceleration due to gravity (m/s^2). In a level turn, lift is given by $L = nW$, where n is the load factor, $n = \frac{1}{cos(\phi)}$, and ϕ is the roll angle. A generic equation for lift can therefore be described by:

$$L = \frac{mg}{cos(\phi)} \tag{7.30}$$

Roll angle can be related to the turn radius R by equating radial forces on the aircraft, the result of which is:

$$gtan(\phi) = \frac{V_a^2}{R} \tag{7.31}$$

The turn rate ω (with units of rad/s) is given by:

$$\omega = \frac{V_a}{R} \tag{7.32}$$

$$\omega = \frac{gtan(\phi)}{V_a} \tag{7.33}$$

Finally, the time spent turning $t_{turning}$ can be found using Equation 7.34, where $\Delta(hdg)$ is the change in heading in radians:

$$t_{turning} = \frac{\Delta(hdg)}{\omega} \tag{7.34}$$

Energy usage can be approximated here by integrating the power required for thrust over the duration of a CPI route. The thrust and drag forces must be equal and opposite, assuming constant airspeed and altitude, so this power can be found by $P = DV_a$, where D is the drag force on the aircraft. The area under this power curve approximates an energy requirement for the time interval considered. Note that this does not equate directly to the energy required from onboard batteries because there are losses associated with power transmission and environmental factors such as turbulence are not taken into account.

The instantaneous drag during a CPI route can be estimated using the lift approximation given in Equation 7.30 and the experimentally found lift to drag ratio ($L/D = 5.95$). This ratio was found by averaging the straight-line descent phase glide ratio (with no throttle input) over five consecutive flights during Trip E (E3 – E7). The descent airspeed for these flights averaged 28 m/s, 3 m/s higher than the modelled airspeed, however this is considered acceptable for the approximation of a relevant L/D ratio. The results of this energy modelling approach are given in Section 7.3.

7.2.5 Practical Route Calculation

Conventional navigational control methods are based in the inertial/earth frame, requiring real-world waypoints in a $[lat, lon, alt]$ format. This means that route option A is the simplest to implement and, providing that both P_1 and P_2 are in acceptable locations, should be the default method.

Route options B and C involve a single additional waypoint, the inertial location of which can be found by using the transformation process outlined in Section 7.2.1. Options D and E are more involved to implement; the curved paths must be split into straight-leg sections to generate a series of waypoint which can be followed using the UAV Mission Control program and GUIDED flight mode. The appropriate number of waypoints depends on the distance $d_{P_1P_2}$; too few and the path flown will be shorter than planned with high bank angle turns, and too many risks the navigational controller missing waypoints. Splitting the wind frame circle into 12 equally spaced waypoints gives reasonable initial results for typical $d_{P_1P_2}$ values, naturally focusing more waypoints in the areas of higher curvature.

Estimation of the wind speed and direction is possible from onboard a fixed-wing UAV using a variety of methods. Ultrasonic anemometers have been trialled by Hollenbech *et al.* however the results were mixed, finding sensitivities to prop-wash and the mounting position of the sensor on the aircraft [144]. A 3D wind map has been developed by Rodriguez *et al.*, motivated by energy efficient trajectory planning for fixed-wing UAVs [145]. This map is generated by fitting data collected by the UAV and a ground station to a Weibull model [146]. A standard wind data collection method was used onboard the UAV, applying a filter such as the Extended Kalman Filter (EKF) to find the wind speed and direction based on the difference between the expected and true states of the aircraft.

For initial flight testing, the CPI method does not require a full 3D wind map; instead, the wind can be estimated using the onboard EKF. The accuracy of the wind estimation increases with consistent manoeuvres such as circling, so will update naturally during the holding phase before the first plume sample. Initial tests should require two full circles to be flown before the first plume sample may be collected.

7.3 Results

This section describes the development of the CPI scheme using Software-In-The-Loop (SITL), and presents the results of the efficiency modelling described in Section 7.2.4. The scheme has been developed to run on an onboard companion computer without changing the autopilot navigational control code, and was developed alongside the revised versions of the UAV Mission Control Program, MAVLink communications implementation, and GCS Update & Command program (Chapter 6).

7.3.1 Efficiency Modelling

A visual comparison of the roll angle with time for each of the CPI routing options is shown in Figure 7.14. Route A is represented by the black line, starting at $\phi = 0°$ and ending with a sustained roll of $\phi = 14.3°$. Routes B and C share the same roll characteristics and are shown by the red line, peaking to $\phi = 45°$ at time $\frac{t_{P_1P_2}}{2}$. Routes D and E also share roll characteristics and are represented by the blue line, with a constant roll angle of $\phi = 0.6°$.

Table 7.3 presents efficiency related metrics for each of the CPI routes, found using Equations 7.30 to 7.34 and the associated energy approximation method for the nominal test case ($V_a = 25$ m/s, $V_W = 10$ m/s, $d_{P_1P_2} = 15$ km). The grey boxes represent numeric values that are fixed as inputs to the calculations. For example, the loiter radius for route A is set in the aircraft parameters, typically at a value of 250 m for operations around Fuego, and routes B and C must by definition turn just 180° (i.e. half a full turn). The number of turns for route A can be found by rearranging Equation 7.34 to find $\Delta(hdg)$, and dividing by 2π radians.

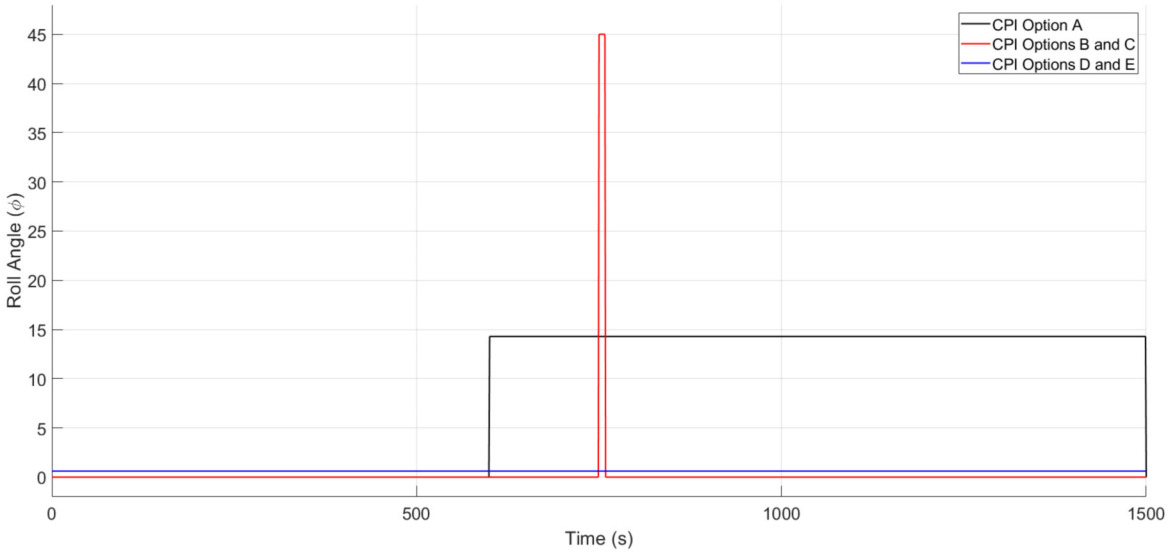

Figure 7.14: Plot showing roll angle with time for each of the CPI routing options, assuming zero turbulence and smooth flight paths.

Table 7.3: Efficiency related metrics for each of the CPI route options, calculated using the nominal test case where $V_a = 25$ m/s, $V_W = 10$ m/s, $d_{P_1P_2} = 15$ km, and $m = 4.5$ kg. The grey cells represent values that are fixed by definition of the route.

	Route A	**Routes B and C**	**Routes D and E**
Number of turns	14.3	0.5	1.0
Turn Type	Loiter	Heading change	Wide
Turn Radius (m)	250.0	64.7	5968.3
Maximum Bank Angle (°)	14.3	45.0	0.6
Rate of Turn (°/s)	5.7	22.5	0.2
Time Turning (s)	900	8	1500
Estimated Energy Required (mAh)	5185.5	5100.6	5088.2

The results in Table 7.3 suggest that, considering route A as the reference point, routes B and C offer a 1.64% energy saving, and routes D and E offer a 1.88% energy saving. Although these savings are small, they are amplified as $d_{P_1P_2}$ increases or V_W decreases. Testing in the real world is likely to separate these efficiency metrics further, highlighting assumptions and approximations which can then be addressed to develop a more representative model. These modelling results suggest that routes D and E are the most efficient of the five options considered here.

Figure 7.15 shows the relationship between estimated energy requirement, plume separation distance $d_{P_1P_2}$, and wind speed V_W for routes D and E. The approximate shape of the surface is the same for all five CPI routing options, with the main differences being the magnitude of the Z-axis data points (i.e. estimated energy required). The most energy intensive scenario involves

low wind speed and high plume sample separation distance, a combination which means $t_{P_1P_2}$ is large and therefore the aircraft must fly further and for longer.

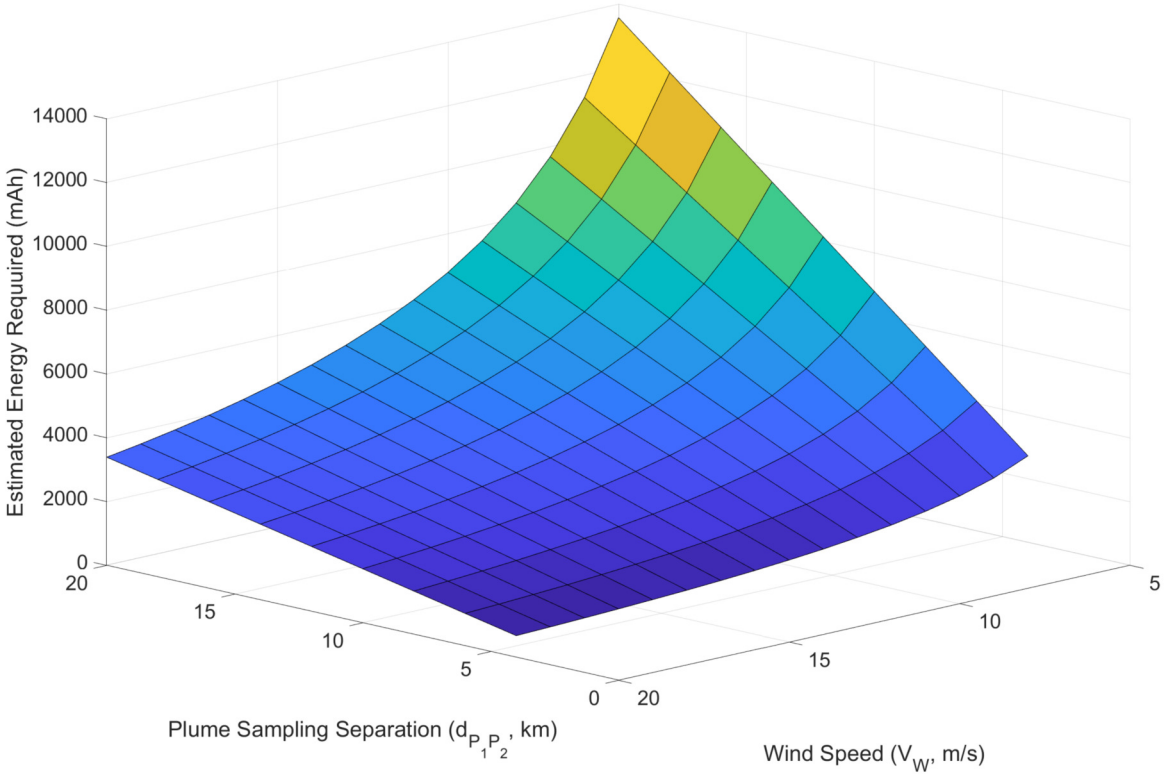

Figure 7.15: Surface plot showing the effect of wind speed (V_W) and plume sampling separation distance ($d_{P_1P_2}$) on estimated energy required (in mAh), for CPI routing options D and E. Data are shown for V_W in the range [5, 20] m/s, and $d_{P_1P_2}$ in the range [5, 20] km.

7.3.2 SITL Testing

The CPI problem was developed alongside the revised UAV Mission Control program so that both the RTTP and CPI schemes may be run from a single program. Each of the CPI options is shown in Figure 7.16 (in the inertial frame), as calculated for a SITL flight case where $\theta_W = 20°$, $V_W = 16$ m/s, $d_{P_1P_2} = 15$ km, and $V_a = 25$ m/s. These conditions were chosen based on the flight zone compatibility results so that every option is visibly possible, with the exception of route B which extends beyond the range of the map into the northern half of the small (2 km) circle around Fuego. A small green star indicates the position of the INSIVUMEH observatory.

The quality of the routes shown in Figure 7.16 depend on the wind and mission parameters. For example, lower wind speeds mean option B extends over the R_{max} (20 km radius) line, and as θ_W approaches 45° options C, D, and E rotate such that they breach the North-South or East-West flight zone boundaries. Option D will likely have waypoints outside the flight zone

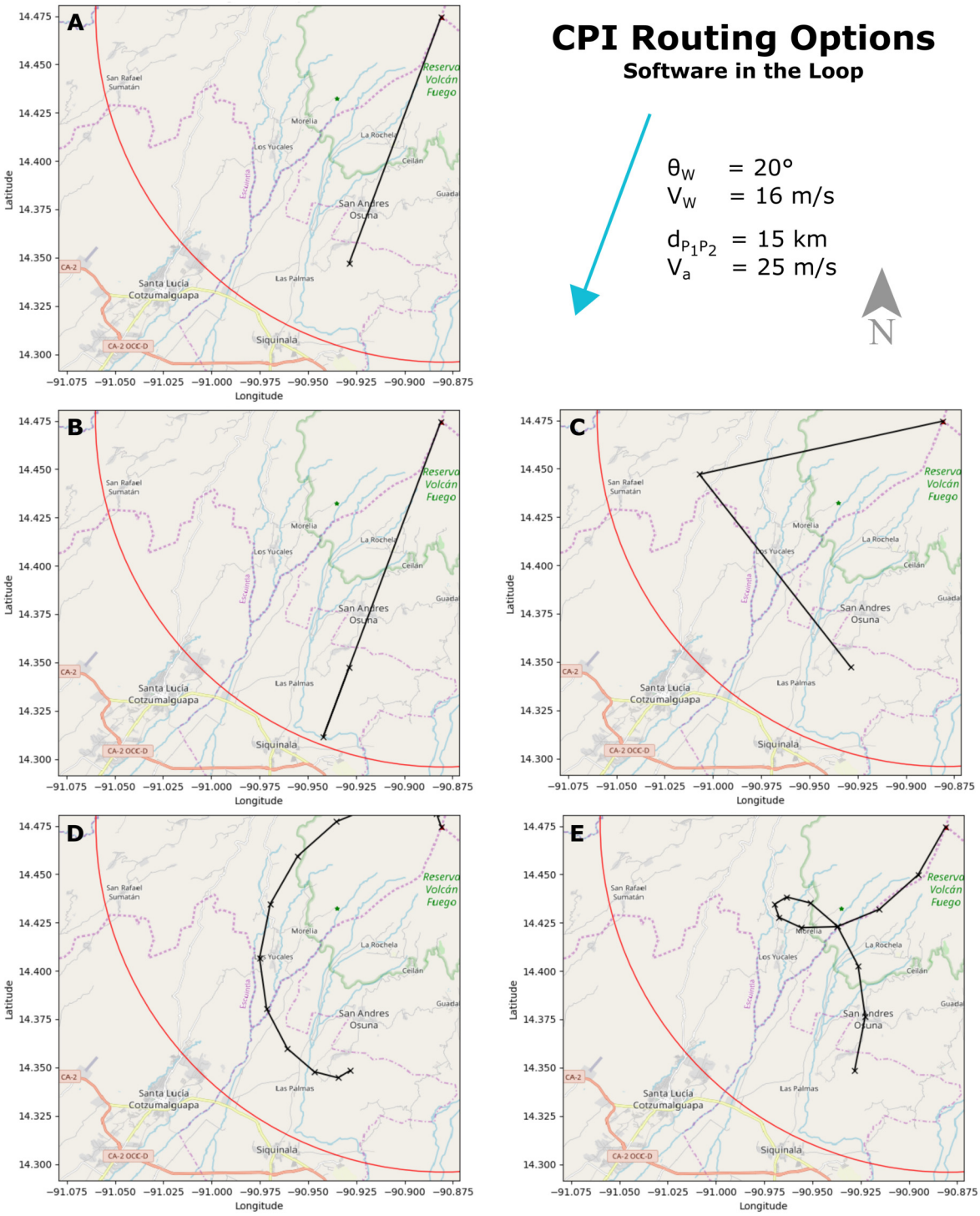

Figure 7.16: CPI routing options in the inertial frame as calculated for SITL, with waypoints indicated by black crosses. The conditions are defined by $\theta_W = 20°$, $V_W = 16$ m/s, $d_{P_1P_2} = 15$ km, and $V_a = 25$ m/s.

pictured, however this may be acceptable in some circumstances. The flight zone considered here includes an area in the immediate vicinity of the crater that is not shown on this map.

7.4 Discussion

A real-world CPI scheme needs to reliably coordinate plume re-interception across a range of conditions while minimising unnecessary energy losses. Of the five routes considered here, there is no clear choice which consistently satisfies these criteria; route A guarantees no breach of the flight zone but is predicted to be less efficient than other options, and routes B – E offer some energy savings but are limited to certain conditions by flight zone compatibility. A tiered approach is therefore recommended, based on the modelling carried out to date, with route type A as the default option. Options B - E can then be ranked according to desirability and feasibility. The best possible route option can be selected on a case by case basis, depending on the conditions.

Flight testing of all route types is required over a range of conditions in order to validate and further develop the models presented here. In reality the wind speed and direction are likely to vary temporally and spatially around Fuego, so robustness to changes in atmospheric conditions must be considered when further developing these CPI routes for real-world use. These changes may be sensed by the onboard EKF, however further work may be required to develop a reliable real-time indicator of the plume position without excessive circling. Another method involves visual sensing using computer vision to compare the predicted and apparent true plume positions. This is unlikely to work for low ash density plumes, and is limited by the 2D nature of images taken onboard, but is a worthwhile area of future work. Once a change in conditions is detected, the flight path must then be altered accordingly.

The loiter pattern at the end of route A means that a reliable local wind speed and direction is likely to be produced by the onboard EKF, however these data may not be representative of the entire space. Visual correction is likely to be challenging at this stage of route A because the plume will only be in view (for a front facing camera) for a short part of each loop.

Route B is highly robust to changes in wind direction. For an overall increase in V_W the plume would be intercepted before reaching position P_2, and for a decrease in V_W the aircraft maybe commanded to continue flying upwind, sampling beyond P_2. Wind speed in this case can be derived from comparing the expected and true ground speeds, however changes to wind direction would be hard to sense with conventional methods. Should a change in θ_W be sensed, the angle of the second leg may be varied in order to intercept the plume.

The length and bearing of the second leg in route C may be modified to accommodate for changing wind conditions. This route is defined by two crosswind legs, so the speed of travel downwind should give an indicator of wind speed. The bearing of the second leg could allow for computer vision to be used to compare the expected and true relative plume positions, however the lack of circling means it may be hard to detect wind direction changes.

Table 7.4: Ordered list of CPI routing options, ranked according to desirability based on energy modelling, satisfaction of flight zone limits, and expected robustness to real-world conditions.

Route Option	**Ranking**	**Reasoning**
E	1	• Joint best efficiency metrics • Satisfies flight zone requirements in most conditions • Likely robustness to variations in wind
C	2	• Good efficiency metrics • Satisfies flight zone requirements in many conditions • Possible robustness to variations in wind
B	3	• Good efficiency metrics • Satisfies flight zone requirements, depending on $d_{P_1P_2}$ • Likely robustness to variations in wind speed
D	4	• Joint best efficiency metrics • Satisfies flight zone requirements in limited conditions • Unlikely robustness to variations in wind
A	5	• Poor efficiency metrics • Guaranteed to stay within flight zone • Possible robustness to variations in wind

A true implementation of routes D and E would involve a smooth path being flown in the wind frame. In reality, this path must be discretised into inertial-frame waypoints (Figure 7.16) meaning some element of efficiency may be lost. The circular element to these paths means it may be possible to approximate changes in wind speed and direction with reasonable accuracy. Route D, if feasible to fly within the permissible operations zone, spends only the last small section of the route facing upwind and towards the plume. This means that computer vision is unlikely to be applicable for correction or validation of the plume position estimate. The tight loop in the inertial version of route E is likely to build quality in the EKF-based wind data estimates and, unlike route D, the last third of the path is angled ideally for correction and validation using computer vision.

Table 7.4 gives a preliminary order of preference for the CPI routes presented in this chapter, based on energy modelling, expected robustness to real-world conditions, and taking into consideration flight zone compatibility.

7.5 Conclusions & Further Work

This chapter has presented a coordinated plume interception scheme for automating ash plume sampling across a large distance (i.e. 15 km). Reliable collection of these samples allows scientific

analysis of the temporal change in ash particle size distribution, an important feature of ash dispersion models that has previously only been estimated using remote sensing techniques. Five routing options have been presented, including a conventional baseline option for comparison.

Load factor based energy modelling has been used to differentiate between routing options. Furthermore, relationships have been developed between wind speed, wind direction, and plume sample separation distance, within the context of the real-world permissible flight zone. The possible robustness of each route to variation in wind conditions has also been discussed. Combining these factors has led to an order of preference between the routes, depending on the conditions and experimentally set plume sample separation distance.

Further work includes flight testing of all five options, across a range of conditions and plume separation distances, in order to validate and develop the modelling presented in this chapter. The schemes may also be developed individually to build robustness to real-world temporal and spatial variation in wind conditions. Two of the routes presented (D and E) are based on a circular flight path in the wind frame that must be discretised into straight legs for inertial-frame waypoint following. Future work therefore also includes investigating a wind-based navigational control system, allowing these routes to be followed in a smoother and more efficient manner.

CHAPTER 8

CONCLUSIONS & FUTURE WORK

This thesis considers the effective long range operation of UAS for volcanic monitoring, focusing on the automation of fixed-wing UAV plume sensing, route planning, and plume re-interception. This work is motivated by volcanological research needs, and the development of airspace management tools, and involves systems development and testing around Volcán de Fuego, Guatemala. Three primary engineering research areas were explored over the course of five trips to Fuego between February 2017 and April 2019, and a parallel secondary research theme ran throughout, consistently targeting UAV-based in-plume ash collection.

8.1 Thesis Summary

Standing at approximately 3800 m Above Mean Sea Level (AMSL) Volcán de Fuego is a stratovolcano in Guatemala, the activity of which limits volcanic monitoring to remotely based techniques. Strombolian eruptions occur (approximately) three to five times per hour, usually expelling ballistics and discrete plumes of ash and gas. The scale and unpredictable timings of these eruptions means it is unsafe to approach the crater for research and monitoring purposes, motivating the development and use of novel methods. Based at an operations point approximately 8 km South-West of the summit (1137 m AMSL), fixed-wing UAVs have been developed to remotely and robustly monitor the summit and collect ash from within the plume. Targeting efficiency and automation, the engineering challenges discussed in this thesis are summarised below.

The theme of method development was common across all the trips to Fuego. Early UAV flights involved exploring methods of basic operation within the experimental environment, from take-off and recovery to flight plan design. Fundamental challenges included naturally low air density due to hot & high conditions, harsh terrain, and meteorological conditions varying both

temporally and spatially during the course of single flights. A successful and robust method of operating was developed over the course of the project, using Skywalker X8 airframes fitted with long range avionics and a Raspberry Pi 3B+ companion computer. A reliable take-off method was developed by combining an elastically powered catapult and a high-thrust motor/electronic speed controller combination. Flight performance parameters on the final trip included flight times of up to 83 minutes and a cumulative altitude gain of 5308 m (17,415 ft), consistently and robustly collecting ash from within plumes.

The first of the primary engineering aims involved the remote sensing and identification of plumes. Automated ash sampling relies on some method of knowing when the aircraft is in a plume; this method ideally involves small, lightweight sensors so that performance losses are minimised and primary payload capacity is maximised. Autopilot data from the first flight through a plume, approximately 2 km downwind of the summit, suggested that a combination of turbulence, temperature, and humidity could indicate plume conditions. Consequent flights were fitted with temperature and humidity sensors in order to quantify those aspects, and the autopilot's vertical axis accelerometer was used as an indicator of turbulence levels. Several plume types were then flown through, including both discrete and continuous plumes, and a metric developed for plume identification. One of these flights had an early version of an ash collection mechanism onboard, and successfully pioneered UAV-based ash sampling from within a plume.

UAV route planning is considered extensively in the literature, with most schemes including some form of simulation but no experimental testing. Path planning as a topic of research began in computer science, focusing on 2D robotics problems, however the field has now expanded to include UAVs in real-world scenarios. The second of the primary engineering aims considered in this thesis explores the development of a Real-Time Trajectory Planner (RTTP) suitable for finding efficient and reactive large scale routes between the launch location and Fuego's summit area. The real-time aspect means new obstacles may be avoided and the route re-calculated mid-flight, and running it onboard effectively increases the level of autonomy. Early tests around Fuego showed promising results, despite relatively high computational costs, with efficiency metrics approximately equivalent to manually developed flight plans. Further analysis and development of the RTTP, and the UAV Mission Control program, show promising results for future testing and use.

A scheme has been developed for the targeted collection of multiple ash samples during a single flight, raising Scanning Electron Microscopy (SEM) stubs into the airflow during plume interception. The Particle Size Distribution (PSD) of ash within these plumes varies temporally post-eruption, and detailed knowledge of these data is useful in the modelling of ash dispersion for aviation management. The third and final primary engineering aim considered presents a method of autonomously re-intercepting plumes in order to collect these samples. This Coordinated Plume Interception (CPI) scheme involves using one of five different routing options, including a baseline

'conventional' option, two straight-leg based methods, and two circular-based methods. These options have been developed and compared within the context of efficiency and compatibility with operational flight zones, and a tiered approach to flight testing is recommended.

8.2 Suggestions of Future Work

While the Skywalker X8 is a capable long-range airframe when fitted with the appropriate avionics, the payload capacity limits the feasible size of scientific sensors. Larger aircraft with the required performance characteristics are commercially available, however their launch and recovery methods prohibit their use in environments such as those found around Fuego. The development of a bespoke vehicle is therefore a worthy area of further work; with a payload capacity of several kilograms, the ability to take-off and land in confined or rough terrain, and performance characteristics suited to long range and high altitude operations.

The plume sensing metric involves combining temperature, humidity, and accelerometer data. For the purposes of development these were collected using sensors, such as the iMet XQ, that restricted data use to post-processing only. Further work in this area involves sourcing and integrating suitable sensors to create a real-time plume sensing module. It may be useful to fit this unit with an accelerometer, however the real-time accelerometer data available from the autopilot may be sufficient. This plume sensing module must then be calibrated and tested, and the metric refined, before including it in the autonomous UAV-based volcano monitoring and plume sampling system.

A genetic algorithm is used for the RTTP presented in this thesis, finding viable and efficient routes from the launch location to Fuego's summit area. Early experimental flight tests are discussed, and a number of changes are made to increase the speed and efficiency of the RTTP implementation. Further work in this area includes flight testing these changes, to assess their real-world benefits, and considering other optimisation techniques such as Particle Swarm Optimisation. Comparing optimisation methods requires a common cost function, the refinement of which is also an area of further work. The incorporation of wind speed and direction in future RTTP methods should also be considered, particularly for problems with large climbs, so that battery capacity may be reserved for use in the destination area.

Five CPI methods are presented and compared mathematically in this thesis. The plume re-interception problem is useful for the real-world collection of plume data, but first needs flight testing to better characterise the routing options. These experiments should test across a range of wind speeds, directions, and sample separations, and consider the robustness of each method to changes in environmental factors. The CPI scheme may benefit from updating the estimated plume position between sampling points; this could involve updating wind estimate data mid-route, perhaps carrying out an intermediate interception to verify the plume position against the expected plume location for that time. Additional energy gains may be found by

developing a 2D navigational controller which is based in the wind frame, allowing smoother flight for those routes which would otherwise require discretisation. Additional future work lies in the integration of the plume sensing module with the CPI method, automating the ash collection mechanism, and mid-scheme plume location verification.

The UAV Mission Control program has been developed to run on a companion computer, for the implementation of the RTTP and CPI without modification of the autopilot. This is a useful safety feature and means testing of the UAV Mission Control, RTTP, and CPI scripts can be done using conventional hardware in the loop simulations with relative ease. Future work includes further development of this high-level control script, as well as the corresponding ground station Update & Command script, for smoother aircraft control and ease of operation.

APPENDIX A

APPENDIX: LIST OF FIXED WING FLIGHTS

This appendix contains a list of all 60 fixed wing flights in Guatemala, spanning Trips A – E. Flight IDs relate to the flight number, resetting at the start of each trip. The aircraft IDs and purposes are summarised below:

Table A.1: Aircraft ID Descriptions

ID	Aircraft	Purpose
BU-518	RiteWing Zephyr II	Long range proof of concept
BU-510	Skywalker X8	Launch and local testing
BU-520	Skywalker X8	Long range volcano monitoring (1/2)
BU-521	Skywalker X8	Long range volcano monitoring (2/2)

Flight Number	Flight ID	Flight Date	Aircraft ID	Battery Type	Motor/ESC Type	Number of SEM Stubs
1	A1	19/02/2017	BU-518	LiPo	OS/Graupner	0
2	A2	20/02/2017	BU-518	LiPo	OS/Graupner	0
3	A3	20/02/2017	BU-518	LiPo	OS/Graupner	0
4	B1	31/10/2017	BU-510	LiPo	Overlander	0
5	B2	31/10/2017	BU-510	LiPo	Overlander	0
6	B3	01/11/2017	BU-510	LiPo	Overlander	0
7	B4	02/11/2017	BU-510	LiPo	Overlander	0
8	B5	04/11/2017	BU-510	LiPo	Overlander	0

9	B6	05/11/2017	BU-520	LiPo	Overlander	0
10	B7	05/11/2017	BU-520	LiPo	Overlander	0
11	B8	06/11/2017	BU-520	LiPo	Overlander	2
12	B9	07/11/2017	BU-520	LiPo	Overlander	2
13	B10	07/11/2017	BU-521	LiPo	Overlander	0
14	B11	07/11/2017	BU-521	LiPo	Overlander	0
15	B12	07/11/2017	BU-520	LiPo	Overlander	2
16	B13	08/11/2017	BU-520	LiPo	Overlander	0
17	B14	08/11/2017	BU-520	LiPo	Overlander	2
18	B15	09/11/2017	BU-520	LiPo	Overlander	2
19	B16	09/11/2017	BU-520	LiPo	Overlander	2
20	B17	10/11/2017	BU-521	LiPo	Overlander	0
21	C1	26/06/2018	BU-520	LiPo	Overlander	0
22	C2	28/06/2018	BU-520	LiPo	Overlander	0
23	C3	28/06/2018	BU-520	LiPo	Overlander	0
24	C4	28/06/2018	BU-520	LiPo	Overlander	0
25	C5	28/06/2018	BU-520	LiPo	Overlander	0
26	C6	28/06/2018	BU-520	LiPo	Overlander	0
27	C7	29/06/2018	BU-521	LiPo	Axi/Jeti	0
28	C8	30/06/2018	BU-521	LiPo	Axi/Jeti	0
29	C9	30/06/2018	BU-521	LiPo	Axi/Jeti	0
30	D1	07/10/2018	BU-521	LiPo	Axi/Jeti	0
31	D2	08/10/2018	BU-521	LiPo	Axi/Jeti	0
32	D3	08/10/2018	BU-521	LiPo	Axi/Jeti	0
33	D4	10/10/2018	BU-521	LiPo	Axi/Jeti	0
34	D5	10/10/2018	BU-521	LiPo	Axi/Jeti	0
35	D6	10/10/2018	BU-521	LiPo	Axi/Jeti	0
36	D7	11/10/2018	BU-521	LiPo	Axi/Jeti	0
37	E1	24/03/2019	BU-521	LiPo	Axi/Jeti	0
38	E2	24/03/2019	BU-521	LiPo	Axi/Jeti	0
39	E3	25/03/2019	BU-521	LiPo	Axi/Jeti	2
40	E4	25/03/2019	BU-520	LiPo	Axi/Jeti	2
41	E5	25/03/2019	BU-521	LiPo	Axi/Jeti	2
42	E6	25/03/2019	BU-520	LiPo	Axi/Jeti	2
43	E7	25/03/2019	BU-520	LiPo	Axi/Jeti	2
44	E8	26/03/2019	BU-520	LiPo	Axi/Jeti	2
45	E9	26/03/2019	BU-520	LiPo	Axi/Jeti	2
46	E10	26/03/2019	BU-521	LiPo	Axi/Jeti	2

47	E11	26/03/2019	BU-521	LiIon	Axi/Jeti	2
48	E12	27/03/2019	BU-521	LiPo	Axi/Jeti	2
49	E13	27/03/2019	BU-521	LiPo	Axi/Jeti	2
50	E14	27/03/2019	BU-521	LiIon	Axi/Jeti	2
51	E15	29/03/2019	BU-521	LiIon	Axi/Jeti	0
52	E16	31/03/2019	BU-521	LiIon	Axi/Jeti	2
53	E17	31/03/2019	BU-521	LiPo	Axi/Jeti	2
54	E18	31/03/2019	BU-521	LiPo	Axi/Jeti	0
55	E19	01/04/2019	BU-521	LiIon	Axi/Jeti	2
56	E20	01/04/2019	BU-521	LiPo	Axi/Jeti	2
57	E21	01/04/2019	BU-521	LiPo	Axi/Jeti	0
58	E22	02/04/2019	BU-521	LiIon	Axi/Jeti	2
59	E23	02/04/2019	BU-521	LiPo	Axi/Jeti	0
60	E24	02/04/2019	BU-521	LiIon	Axi/Jeti	0

References

[1] C. Greatwood, T. S. Richardson, J. Freer, R. M. Thomas, A. Rob Mackenzie, R. Brownlow, D. Lowry, R. E. Fisher, and E. G. Nisbet, "Atmospheric sampling on ascension island using multirotor UAVs," *Sensors (Switzerland)*, vol. 17, no. 6, pp. 1–24, 2017.

[2] HobbyKing, "HobbyKing Bixler 2." [Online]. Available: https://hobbyking.com/en{_}us/h-king-bixler-2-epo-1500mm-pnp.html

[3] Flying Robot, "Skywalker X8." [Online]. Available: https://flyingrobot.co/products/skywalker-x8-wing

[4] Applied Aeronautics, "Albatross Airframe." [Online]. Available: http://store.appliedaeronautics.com/albatross-airframe/

[5] UKRSPEC Systems, "UKRSPEC Systems PD-1." [Online]. Available: https://spetstechnoexport.com/en/product/174https://ukrspecsystems.com/pd-1-3

[6] Electricwingman, "Building an FPV Racing Quadcopter." [Online]. Available: https://www.electricwingman.com/guides/building-fpv-racing-quadcopter.aspx

[7] DJI, "DJI Mavic 2." [Online]. Available: https://www.dji.com/uk/mavic-2

[8] ——, "DJI Inspire 1." [Online]. Available: https://www.dji.com/uk/inspire-1

[9] Vulcan UAV, "Vulcan UAV Raven." [Online]. Available: http://vulcanuav.com/aircraft/

[10] Google Earth, "https://earth.google.com/web/," 2020. [Online]. Available: https://earth.google.com/web/

[11] VWR International, "VWR Store," 2020. [Online]. Available: https://us.vwr.com/stibo/bigweb/std.lang.all/19/24/11941924.jpg

[12] Open Street Map Foundation and Contributors, "Open Street Map." [Online]. Available: https://www.openstreetmap.org/copyright

[13] MAVLink, "Upload a Mission to the Vehicle," 2020. [Online]. Available: https://mavlink.io/en/services/mission.html{#}uploading{_}mission

[14] F. Nex and F. Remondino, "UAV for 3D mapping applications: A review," *Applied Geomatics*, vol. 6, no. 1, pp. 1–15, 2014.

[15] R. Duffy, "Mapping the UK Drone Industry," 2018. [Online]. Available: https://www.nesta.org.uk/blog/uk-drone-industry-map/

[16] N. H. Motlagh, M. Bagaa, and T. Taleb, "UAV-Based IoT Platform: A Crowd Surveillance Use Case," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 128–134, 2017.

[17] S. Doyle, "Drone warfare: the autonomous debate," 2018. [Online]. Available: https://eandt.theiet.org/content/articles/2018/11/drone-warfare-the-autonomous-debate/

[18] C. K. Basu, F. Deacon, J. R. Hutchinson, and A. M. Wilson, "The running kinematics of free-roaming giraffes, measured using a low cost unmanned aerial vehicle (UAV)," *PeerJ*, vol. 2019, no. 2, pp. 1–21, 2019.

[19] J. Jeziorska, "UAS for wetland mapping and hydrological modeling," *Remote Sensing*, vol. 11, no. 17, 2019.

[20] D. Orbea, J. Moposita, W. G. Aguilar, M. Paredes, R. P. Reyes, and L. Montoya, "Vertical take off and landing with fixed rotor," *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies, CHILECON 2017 - Proceedings*, vol. 2017-Janua, pp. 1–6, 2017.

[21] F. Ruggiero, M. A. Trujillo, R. Cano, H. Ascorbe, A. Viguria, C. Perez, V. Lippiello, A. Ollero, and B. Siciliano, "A multilayer control for multirotor UAVs equipped with a servo robot arm," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 4014–4020, 2015.

[22] B. R. Jordan, "Collecting field data in volcanic landscapes using small UAS (sUAS)/drones," *Journal of Volcanology and Geothermal Research*, vol. 385, pp. 231–241, 2019. [Online]. Available: https://doi.org/10.1016/j.jvolgeores.2019.07.006

[23] C. A. Neal, S. R. Brantley, L. Antolik, J. L. Babb, and Etc, "The 2018 rift eruption and summit collapse of Kīlauea Volcano," *Science*, vol. 363, no. January, pp. 367–374, 2019.

[24] D. Pieri, J. A. Diaz, G. Bland, M. Fladeland, Y. Madrigal, E. Corrales, O. Alegria, A. Alan, V. Realmuto, T. E. D. Miles, A. L. I. Abtahi, C. Rica, S. Jose, and C. Rica, "In situ observations and sampling of volcanic emissions with NASA and UCR unmanned aircraft , including a case study at Turrialba Volcano , Costa Rica," 2013.

[25] C. Gomez and B. Kennedy, "Capturing volcanic plumes in 3D with UAV-based photogrammetry at Yasur Volcano – Vanuatu," *Journal of Volcanology and*

Geothermal Research, vol. 350, pp. 84–88, 2018. [Online]. Available: https://doi.org/10.1016/j.jvolgeores.2017.12.007

[26] E. J. Liu, K. Wood, E. Mason, M. Edmonds, A. Aiuppa, G. Giudice, M. Bitetto, V. Francofonte, S. Burrow, T. Richardson, M. Watson, T. D. Pering, T. C. Wilkes, A. J. McGonigle, G. Velasquez, C. Melgarejo, and C. Bucarey, "Dynamics of Outgassing and Plume Transport Revealed by Proximal Unmanned Aerial System (UAS) Measurements at Volcán Villarrica, Chile," *Geochemistry, Geophysics, Geosystems*, vol. 20, no. 2, pp. 730–750, 2019.

[27] B. Schellenberg, T. S. Richardson, R. J. Clarke, M. Watson, J. Freer, A. McConville, and G. Chigna, "BVLOS Operations of Fixed-Wing UAVs for the Collection of Volcanic Ash Above Fuego Volcano, Guatemala," in *AIAA Scitech 2020 Forum*, no. January. Orlando, Florida: American Institute of Aeronautics and Astronautics, jan 2020, pp. 1–11. [Online]. Available: https://arc.aiaa.org/doi/10.2514/6.2020-2204

[28] B. Schellenberg, T. Richardson, M. Watson, C. Greatwood, R. Clarke, R. Thomas, K. Wood, J. Freer, H. Thomas, E. Liu, F. Salama, and G. Chigna, "Remote sensing and identification of volcanic plumes using fixed-wing UAVs over Volcán de Fuego, Guatemala," *Journal of Field Robotics*, vol. 36, no. June, p. rob.21896, 2019. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21896

[29] B. Schellenberg, T. Richardson, A. Richards, R. Clarke, and M. Watson, "On-board real-time trajectory planning for fixed wing unmanned aerial vehicles in extreme environments," *Sensors (Switzerland)*, vol. 19, no. 19, 2019.

[30] British Model Flying Association, "History of Model Flying." [Online]. Available: https://bmfa.org/Info/History-of-Model-Flying

[31] Academy of Model Aeronautics, "Radio Control." [Online]. Available: https://www.modelaircraft.org/radio-control

[32] L. R. Newcome, *Unmanned Aviation: A Brief History of Unmanned Aerial Vehicles*. American Institute of Aeronautics and Astronautics, Inc., 2012.

[33] J. F. Guilmartin, "Unmanned Aerial Vehicle," 2019. [Online]. Available: https://www.britannica.com/technology/transportation-technology

[34] J. Alkobi, "The evolution of Drones: from Military to Hobby and Commercial," 2019. [Online]. Available: https://percepto.co/the-evolution-of-drones-from-military-to-hobby-commercial/

[35] K. B. Culver, “From Battlefield to Newsroom: Ethical Implications of Drone Technology in Journalism,” *Journal of Mass Media Ethics: Exploring Questions of Media Morality*, vol. 29, no. 1, pp. 52–64, 2014.

[36] 60 Minutes Overtime, “Amazon unveils futuristic plan: delivary by drone,” 2013. [Online]. Available: https://www.cbsnews.com/news/amazon-unveils-futuristic-plan-delivery-by-drone/

[37] K. Chiu, “The story of drone pioneer DJI,” 2018. [Online]. Available: https://www.abacusnews.com/whois-whatis/dji-dominates-world-drones/article/2128689

[38] DJI, “DJI releases All-in-One Solution, Read-to-Fly "Phantom" Quadcopter,” 2013. [Online]. Available: https://www.dji.com/newsroom/news/dji-releases-all-in-one-solution-read-to-fly-phantom-quadcopter

[39] A. J. S. McGonigle, A. Aiuppa, G. Giudice, G. Tamburello, A. J. Hodson, and S. Gurrieri, “Unmanned aerial vehicle measurements of volcanic carbon dioxide fluxes,” *Geophysical Research Letters*, vol. 35, no. 6, pp. 3–6, 2008.

[40] A. Sanchez, J. Escareño, O. Garcia, and R. Lozano, *Autonomous Hovering of a Noncyclic Tiltrotor UAV: Modeling, Control and Implementation*. IFAC, 2008, vol. 41, no. 2. [Online]. Available: http://dx.doi.org/10.3182/20080706-5-KR-1001.00138

[41] S. Yu and Y. Kwon, “Development of VTOL Drone for Stable Transit Flight,” *Journal of Computer and Communications*, vol. 05, no. 07, pp. 36–43, 2017.

[42] B. Yuksek, A. Vuruskan, U. Ozdemir, M. A. Yukselen, and G. Inalhan, “Transition Flight Modeling of a Fixed-Wing VTOL UAV,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 84, no. 1-4, pp. 83–105, 2016. [Online]. Available: http://dx.doi.org/10.1007/s10846-015-0325-9

[43] Civil Aviation Authority, “Large Unmanned Aircraft, CAP722.” [Online]. Available: https://www.caa.co.uk/Commercial-industry/Aircraft/Unmanned-aircraft/Large-unmanned-aircraft/https://publicapps.caa.co.uk/modalapplication.aspx?appid=11{&}mode=detail{&}id=415

[44] Drone Safe UK, “The Drone Code.” [Online]. Available: https://dronesafe.uk/drone-code/

[45] E. A. Euteneuer and G. Papageorgiou, “UAS insertion into commercial airspace: Europe and US standards perspective,” *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, pp. 5C5–1–5C5–12, 2011.

[46] F. Tomasello and D. Haddon, “Detect and avoid for unmanned aircraft systems in the total system approach,” *2011 Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles, ESAV’11*, pp. 47–52, 2011.

[47] I. Colomina and P. Molina, "Unmanned aerial systems for photogrammetry and remote sensing: A review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 92, pp. 79–97, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.isprsjprs.2014.02.013

[48] C. Sweatte, R. Storvold, P. Ruel, W. M, K. Tarr, A. la Cour-Harbo, B. Feldberg, J. Kivinen, H. Hólm, M. Raustein, E. Jangren, B. Mulac, J. Adler, D. Davis, L. Cary, and S. Lesenkov, "Implementing Scientific Data Collection across the Arctic Oceanic Region Utilizing Unmanned Aircraft Systems (UAS)," *Arctic Monitoring and Assessment Programme*, p. 5, 2015.

[49] M. Johnson, J. Jung, J. Rios, J. Mercer, J. Homola, T. Prevot, D. Mulfinger, and P. Kopardekar, "Flight test evaluation of an unmanned aircraft system traffic management (UTM) concept for multiple beyond-visual-line-of-sight operations," *12th USA/Europe Air Traffic Management R and D Seminar*, 2017.

[50] D. A. Divis, "With First-Ever FAA Waiver U.S. Closes In On True, Commercial BVLOS Operations," 2019. [Online]. Available: https://insideunmannedsystems.com/with-first-ever-faa-waiver-u-s-closes-in-on-true-commercial-bvlos-operations/

[51] R. J. Clarkson, E. J. E. Majewicz, and P. Mack, "A re-evaluation of the 2010 quantitative understanding of the effects volcanic ash has on gas turbine engines," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 230, no. 12, pp. 2274–2291, 2016. [Online]. Available: http://pig.sagepub.com/content/230/12/2274.abstract

[52] A. Stohl, A. J. Prata, S. Eckhardt, L. Clarisse, A. Durant, S. Henne, N. I. Kristiansen, A. Minikin, U. Schumann, P. Seibert, K. Stebel, H. E. Thomas, T. Thorsteinsson, K. Tørseth, and B. Weinzierl, "Determination of time-and height-resolved volcanic ash emissions and their use for quantitative ash dispersion modeling: The 2010 Eyjafjallajökull eruption," *Atmospheric Chemistry and Physics*, vol. 11, no. 9, pp. 4333–4351, 2011.

[53] I. M. Watson, V. J. Realmuto, W. I. Rose, A. J. Prata, G. J. S. Bluth, Y. Gu, C. E. Bader, and T. Yu, "Thermal infrared remote sensing of volcanic emissions using the moderate resolution imaging spectroradiometer," *Journal of Volcanology and Geothermal Research*, vol. 135, no. 1-2, pp. 75–89, 2004.

[54] H. E. Thomas and I. M. Watson, "Observations of volcanic emissions from space: Current and future perspectives," *Natural Hazards*, vol. 54, no. 2, pp. 323–354, 2010.

[55] A. J. Prata, F. O. R. Volcanic, and A. S. H. Clouds, "Infrared radiative transfer calculations for volcanic ash clouds," vol. 16, no. 11, pp. 1293–1296, 1989.

[56] M. I. Mishchenko, "Light scattering by size-shape distributions of randomly oriented axially symmetric particles of a size comparable to a wavelength." *Applied optics*, vol. 32, no. 24, pp. 4652–4666, 1993.

[57] L. M. Western, M. I. Watson, and P. N. Francis, "Uncertainty in two-channel infrared remote sensing retrievals of a well-characterised volcanic ash cloud," *Bulletin of Volcanology*, vol. 77, no. 8, 2015.

[58] United States Geological Survey, "Pyroclastic Flows," 2016. [Online]. Available: https://volcanoes.usgs.gov/vhp/pyroclastic{_}flows.html

[59] Smithsonian Institute, "https://volcano.si.edu/volcano.cfm?vn=342090&vtab=GeneralInfo," 2002. [Online]. Available: https://volcano.si.edu/volcano.cfm?vn=342090{&}vtab=GeneralInfo

[60] E. I. Gordeev and O. A. Girina, "Volcanoes and their hazard to aviation," *Herald of the Russian Academy of Sciences*, vol. 84, no. 1, pp. 1–8, 2014.

[61] S. S. Wegener, "UAV Autonomous Operations for Airborne Science Missions," in *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, no. September, 2004, pp. 1–10.

[62] J. Everaerts, "The Use of Unmanned Aerial Vehicles (UAVs) For Remote Sensing and Mapping," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 37, no. B1, pp. 1187–1192, 2008.

[63] V. V. Klemas, "Coastal and Environmental Remote Sensing from Unmanned Aerial Vehicles : An Overview," *Journal of Coastal Research*, vol. 31, no. 5, pp. 1260–1267, 2015.

[64] G. Di Stefano, G. Romeo, A. Mazzini, A. Iarocci, S. Hadi, and S. Pelphrey, "The Lusi drone : A multidisciplinary tool to access extreme environments," *Marine and Petroleum Geology*, vol. 30, no. December, pp. 1 – 12, 2017.

[65] V. Ramanathan, M. V. Ramana, G. Roberts, D. Kim, C. Corrigan, C. Chung, and D. Winker, "Warming trends in Asia amplified by brown cloud solar absorption," *Nature*, vol. 448, no. 7153, pp. 575–578, 2007.

[66] R. M. Thomas, K. Lehmann, H. Nguyen, D. L. Jackson, D. Wolfe, and V. Ramanathan, "Measurement of turbulent water vapor fluxes using a lightweight unmanned aerial vehicle system," *Atmospheric Measurement Techniques*, vol. 5, no. 1, pp. 243–257, jan 2012. [Online]. Available: https://www.atmos-meas-tech.net/5/243/2012/

[67] E. M. Wilcox, R. M. Thomas, P. S. Praveen, K. Pistone, F. A.-M. Bender, and V. Ramanathan, "Black carbon solar absorption suppresses turbulence

in the atmospheric boundary layer," *Proceedings of the National Academy of Sciences*, vol. 113, no. 42, pp. 11 794–11 799, 2016. [Online]. Available: http://www.pnas.org/lookup/doi/10.1073/pnas.1525746113

[68] T. F. Villa, F. Gonzalez, B. Miljievic, Z. D. Ristovski, and L. Morawska, "An Overview of Small Unmanned Aerial Vehicles for Air Quality Measurements : Present Applications and Future Prospectives," *Sensors*, vol. 16, pp. 12–20, 2016.

[69] T. F. Villa, F. Salimi, K. Morton, L. Morawska, and F. Gonzalez, "Development and validation of a UAV based system for air pollution measurements," *Sensors (Switzerland)*, vol. 16, no. 12, 2016.

[70] T. J. Schuyler and M. I. Guzman, "Unmanned Aerial Systems for Monitoring Trace Tropospheric Gases," *Atmosphere*, vol. 8, no. 206, 2017.

[71] C. Greatwood, T. Richardson, J. Freer, R. Thomas, R. Brownlow, D. Lowry, R. E. Fisher, and E. G. Nisbet, "Automatic path generation for multirotor descents through varying air masses above Ascension Island," *AIAA Atmospheric Flight Mechanics Conference*, no. January 2016, pp. 1–12, 2016.

[72] B. Altstädter, A. Platis, B. Wehner, A. Scholtz, N. Wildmann, M. Hermann, R. Käthner, H. Baars, J. Bange, and A. Lampert, "ALADINA – an unmanned research aircraft for observing vertical and horizontal distributions of ultrafine particles within the," *Atmospheric Measurement Tec*, vol. 8, pp. 1627–1639, 2015.

[73] M. Ogiso, H. Matsubayashi, T. Yamamoto, M. Murase, F. Kimata, Y. Yamanaka, S. Horikawa, K. Matsuhiro, T. Matsushima, H. Mori, T. Ohkura, S. Yoshikawa, R. Miyajima, H. Inoue, T. Mishima, T. Sonoda, K. Uchida, K. Yamamoto, H. Nakamichi, T. Mori, T. Hashimoto, A. Terada, M. Yoshimoto, R. Kazahaya, H. Shinohara, and R. Tanaka, "Volcanic plume measurements using a UAV for the 2014 Mt. Ontake eruption," *Earth, Planets and Space*, vol. 68, no. 1, p. 49, 2016.

[74] D. Longo, C. Melita, G. Muscato, and G. Giudice, "Measurement and Exploration in Volcanic Environments," in *Handbook of Unmanned Aerial Vehicles*, G. J. Vachtsevanos and K. P. Valavanis, Eds. Springer Science+Business Media Dordrech, 2014, pp. 2667–2692.

[75] S. Amici, M. Turci, F. Giulietti, S. Giammanco, M. F. Buongiorno, A. La Spina, and L. Spampinato, "Volcanic Environments Monitoring By Drones Mud Volcano Case Study," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-1/W2, no. May 2014, pp. 5–10, 2013. [Online]. Available: http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-1-W2/5/2013/

[76] B. R. Jordan, “A bird’s-eye view of geology: The use of micro drones / UAVs in geologic fieldwork and education,” *GSA Today*, vol. 25, no. 7, pp. 50–52, 2015.

[77] J. Stix, J. M. D. Moor, J. Rüdiger, A. Alan, E. Corrales, F. D. Arcy, J. A. Diaz, and M. Liotta, “Journal of Geophysical Research : Solid Earth Using Drones and Miniaturized Instrumentation to Study Degassing at Turrialba and Masaya Volcanoes , Central America,” *Journal of Geophysical Research: Solid Earth*, no. 2013, pp. 1–20, 2018.

[78] C.-j. Lai, J.-k. Liu, W.-c. Hsu, K.-s. Li, M.-c. Wu, and K.-t. Chang, “An Experiment Of Geothermal Exploration With An UAS-TIR In Xiaoyoukeng Area Of Tatun Volcanoes, Taiwan,” in *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, 2018, pp. 7886–7889.

[79] A. Flores, D. Scipion, C. Saito, J. Apaza, and M. Milla, “Unmanned Aircraft System for Andean Volcano monitoring and surveillance,” *2019 IEEE International Symposium on Safety, Security, and Rescue Robotics, SSRR 2019*, pp. 297–302, 2019.

[80] K. Wood, A. Albadra, L. Berthoud, A. Calway, M. Watson, H. Thomas, T. Richardson, E. Liu, and G. Chigna, “Determining the three-dimensional structure of a volcanic plume using Unoccupied Aerial System (UAS) imagery,” *Journal of Volcanology and Geothermal Research*, no. xxxx, p. 106731, 2019. [Online]. Available: https://doi.org/10.1016/j.jvolgeores.2019.106731

[81] J. A. Paredes, C. Saito, M. Abarca, and F. Cuellar, “Study of effects of high-altitude environments on Multicopter and Fixed-Wing UAVs ’ Energy Consumption and Flight Time,” in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, 2017, pp. 1645–1650.

[82] DGAC Guatemala, “Drones - Aeronaves no tripuladas RAN,” 2020. [Online]. Available: http://www.dgac.gob.gt/index.php/drones

[83] G. Montes, B. Letheren, T. Villa, and F. Gonzalez, “Bio-inspired plume tracking algorithm for UAVs,” *Australasian Conference on Robotics and Automation, ACRA*, vol. 02-04-Dece, 2014.

[84] B. Letheren and G. Montes, “Design and Flight Testing of a Bio-Inspired Plume Tracking Algorithm for Unmanned Aerial Vehicles,” *Conference, 2016 IEEE*, pp. 1–9, 2016. [Online]. Available: http://ieeexplore.ieee.org/abstract/document/7500614/

[85] P. Ramanatha, U. R. Nelakuditi, S. Ravishankar, and V. Ranganathan, “UAV based smoke plume detection system controlled via the short message service through the GSM network,” *Proceedings of the International Conference on Inventive Computation Technologies, ICICT 2016*, vol. 2, pp. 1–4, 2016.

[86] R. S. Allison, J. M. Johnston, G. Craig, and S. Jennings, "Airborne optical and thermal remote sensing for wildfire detection and monitoring," *Sensors (Switzerland)*, vol. 16, no. 8, 2016.

[87] C. Seiber, D. Nowlin, B. Landowski, and M. E. Tolentino, "Tracking hazardous aerial plumes using IoT-enabled drone swarms," *IEEE World Forum on Internet of Things, WF-IoT 2018 - Proceedings*, vol. 2018-Janua, pp. 377–382, 2018.

[88] S. Yang, R. W. Talbot, M. B. Frish, L. M. Golston, N. F. Aubut, M. A. Zondlo, C. Gretencord, and J. McSpiritt, "Natural gas fugitive leak detection using an unmanned aerial vehicle: Measurement system description and mass balance approach," *Atmosphere*, vol. 9, no. 10, 2018.

[89] M. Haeffelin, "Exploring microphysical , radiative , dynamic and thermodynamic processes driving fog and low stratus clouds using ground-based Lidar and Radar measurements," vol. 18. EGU General Assembly 2016, 2016, p. 6263. [Online]. Available: https://meetingorganizer.copernicus.org/EGU2016/EGU2016-6263.pdf

[90] J. K. Kuchar and L. C. Yang, "A review of conflict detection and resolution modeling methods," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 4, pp. 179–189, 2000. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=898217

[91] C. Goerzen, Z. Kong, and B. Mettler, "A survey of motion planning algorithms from the perspective of autonomous UAV guidance," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 57, pp. 65–100, 2010.

[92] R. W. Beard and T. W. McLain, *Small Unmanned Aircraft, Theory and Practice*. Princeton: Princeton University Press, 2012.

[93] J. K. Howlet, G. Schulein, and M. H. Mansur, "A Practical Approach to Obstacle Field Route Planning for Unmanned Rotorcraft," in *American Helicopter Society 60th Annual Forum; 7-10 Jun. 2004; Baltimore, MD; United States*, Baltimore, MD, 2004.

[94] O. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," *The International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986. [Online]. Available: http://journals.sagepub.com/doi/10.1177/027836498600500106

[95] Z. Kong, V. Korukanti, and B. Mettler, "Mapping 3D Guidance Performance Using Approximate Optimal Cost-to-go Function," *AIAA Guidance, Navigation, and Control Conference*, no. August, pp. 1–21, 2009. [Online]. Available: http://arc.aiaa.org/doi/10.2514/6.2009-6017

[96] J. T. Betts, “Survey of Numerical Methods for Trajectory Optimization,” *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998. [Online]. Available: http://arc.aiaa.org/doi/10.2514/2.4231

[97] M. Hoy, A. S. Matveev, and A. V. Savkin, “Algorithms for collision-free navigation of mobile robots in complex cluttered environments : a survey,” *Robotica*, vol. 33, no. 2015, pp. 463–497, 2015.

[98] S. Karaman and E. Frazzoli, “Incremental sampling-based algorithms for optimal motion planning,” *Robotics: Science and Systems*, vol. 6, pp. 267–274, 2011.

[99] R. Diankov and J. Kuffner, “Randomized statistical path planning,” in *IEEE International Conference on Intelligent Robots and Systems*, no. May, 2007, pp. 1–6.

[100] B. M. Sathyaraj, L. C. Jain, A. Finn, and S. Drake, “Multiple UAVs path planning algorithms: A comparative study,” *Fuzzy Optimization and Decision Making*, vol. 7, no. 3, pp. 257–267, 2008.

[101] S. S. Ge and Y. J. Cui, “New potential functions for mobile robot path planning,” *IEEE Transactions on Robotics and Automation*, vol. 16, no. 5, pp. 615–620, 2000.

[102] B. K. Patle, G. B. L, A. Pandey, D. R. K. Parhi, and A. Jagadeesh, “A review : On path planning strategies for navigation of mobile robot,” *Defence Technology*, no. xxxx, 2019. [Online]. Available: https://doi.org/10.1016/j.dt.2019.04.011

[103] Z. Zhen, “UAV path planning method based on ant colony optimization,” *2010 Chinese Control and Decision Conference*, pp. 3790–3792, 2010. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5498477

[104] D. Karaboga, “AN IDEA BASED ON HONEY BEE SWARM FOR NUMERICAL OPTIMIZATION,” Tech. Rep., 2005.

[105] L. Davis and M. Steenstrup, *Genetic Algorithms and Simulated Annealing: An Overview*. Morgan Kaufman Publishers, Inc.,Los Altos, CA, 1987, vol. 5, no. 1. [Online]. Available: http://ieeexplore.ieee.org/stampPDF/getPDF.jsp?tp={&}arnumber=17235{&}isnumber=627

[106] H. Martinez-Alfaro and S. Gómez-Garcia, “Mobile robot path planning and tracking using simulated annealing and fuzzy logic control,” *Expert Systems with Applications*, vol. 15, no. 3-4, pp. 421–429, 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417498000554

[107] A. Dogan, “Probabilistic Path Planning For UAVs,” *Journal of Aircraft*, vol. 43, no. 2, pp. 318–335, 2006. [Online]. Available: http://arc.aiaa.org/doi/abs/10.2514/1.15204

[108] V. Roberge, M. Tarbouchi, and G. Labonte, "Comparison of Parallel Genetic Algorithm and Particle Swarm Optimization for Real-Time UAV Path Planning," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 132–141, 2013.

[109] I. K. Nikolos, K. P. Valavanis, N. C. Tsourveloudis, and A. N. Kostaras, "Evolutionary Algorithm Based Offline / Online Path Planner for UAV Navigation," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 33, no. 6, pp. 898–912, 2003.

[110] K. Deb, S. Pratab, S. Agarwal, and T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithm: NGSA-II," *IEEE Transactions on Evolutionary Computing*, vol. 6, no. 2, pp. 182–197, 2002.

[111] S. Mittal and K. Deb, "Three-Dimentional Offline Path Planning for UAVs Using Multiobjective Evolutionary Algorithms," *2007 IEEE Congress on Evolutionary Computation (CEC'2007)*, pp. 3195–3202, 2007.

[112] R. Wise, A. Pongpunwattana, and R. Rysdyk, "Modular Tactical Autonomous Guidance Using Evolution-based Algorithms," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, no. August, 2008, pp. 1–11. [Online]. Available: http://arc.aiaa.org/doi/10.2514/6.2008-6302

[113] S. MahmoudZadeh, A. M. Yazdani, K. Sammut, and D. M. W. Powers, "Online path planning for AUV rendezvous in dynamic cluttered undersea environment using evolutionary algorithms," *Applied Soft Computing Journal*, vol. 70, pp. 929–945, 2018. [Online]. Available: https://doi.org/10.1016/j.asoc.2017.10.025

[114] Y. Xue and J.-Q. Q. Sun, "Solving the Path Planning Problem in Mobile Robotics with the Multi-Objective Evolutionary Algorithm," *Applied Sciences (Switzerland)*, vol. 8, no. 1425, 2018.

[115] B. Salamat and A. M. Tonello, "Stochastic Trajectory Generation Using Particle Swarm Optimization for Quadrotor Unmanned Aerial Vehicles (UAVs)," *Aerospace*, vol. 4, no. 2, p. 27, 2017. [Online]. Available: http://www.mdpi.com/2226-4310/4/2/27

[116] Y. Shi and R. Eberhart, "A Modified Particle Swarm Optimizer," in *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence*, 1998, pp. 69–73.

[117] S. Saremi, S. Mirjalili, and A. Lewis, "Advances in Engineering Software Grasshopper Optimisation Algorithm : Theory and application," *Advances in Engineering Software*, vol. 105, pp. 30–47, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.advengsoft.2017.01.004

[118] J. Wu, H. Wang, N. Li, P. Yao, Y. Huang, Z. Su, and Y. Yu, "Distributed trajectory optimization for multiple solar-powered UAVs target tracking in urban environment by Adaptive Grasshopper Optimization Algorithm," *Aerospace Science and Technology*, vol. 70, pp. 497–510, 2017.

[119] T. Bäck, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford: Oxford University Press, 1996.

[120] M. Dorigo and L. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *Evolutionary Computation, IEEE ...*, vol. 1, no. 1, pp. 53–66, 1997. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=585892

[121] G. Ma, H. Duan, and S. Liu, "Improved Ant Colony Algorithm for Global Optimal Trajectory Planning of UAV under Complex Environment." *International Journal of Computer Science and Applications*, vol. 4, no. 3, pp. 57–68, 2007.

[122] T. Zaza and A. Richards, "Intelligent foresight for UAV routing problems," *2016 UKACC International Conference on Control, UKACC Control 2016*, pp. 1–6, 2016.

[123] R. Storn and K. Price, "Differential Evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces," *Journal of Global Optimization*, pp. 1–12, 1995.

[124] W. Du, S. Yung, S. Leung, Y. Tang, and A. V. Vasilakos, "Differential Evolution with Event-Triggered Impulsive Control," *IEEE Transactions on Cybernetics*, pp. 1–40, 2015.

[125] K. Y. Kok and P. Rajendran, "Differential-Evolution Control Parameter Optimization for Unmanned Aerial Vehicle Path Planning," *PLOS ONE*, pp. 1–12, 2016.

[126] C. A. Floudas, *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*. Oxford University Press, 1995, 1995.

[127] A. Richards and J. P. How, "Aircraft Trajectory Planning With Collision Avoidance Using Mixed Integer Linear Programming," in *Proceedings of the American Control Conference*, 2002, pp. 1936–1941.

[128] J. Bellingham, A. Richards, and J. P. How, "Receding horizon control of autonomous aerial vehicles," *American Control Conference, 2002. Proceedings of the 2002*, vol. 5, pp. 3741–3746, 2002.

[129] A. Richards and O. Turnbull, "Inter-sample avoidance in trajectory optimizers using mixed-integer linear programming," *International Journal of Robust and*

Nonlinear Control, vol. 25, no. 2015, pp. 521–526, 2013. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/rnc.3101/abstracthttp://onlinelibrary.wiley.com/doi/10.1002/rnc.1553/abstracthttp://doi.wiley.com/10.1002/rnc.3101

[130] Y. Kuwata and J. P. How, "Stable trajectory design for highly constrained environments using receding horizon control," *American Control Conference ACC*, vol. 1, pp. 902–907, 2004. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=1383721

[131] F. Borrelli, D. Subramanian, A. Raghunathan, and L. Biegler, "MILP and NLP Techniques for centralized trajectory planning of multiple unmanned air vehicles," *2006 American Control Conference*, no. July 2016, p. 6 pp., 2006. [Online]. Available: http://ieeexplore.ieee.org/document/1657644/

[132] M. Raap, M. Zsifkovits, and S. Pickl, "Trajectory optimization under kinematical constraints for moving target search," *Computers and Operations Research*, vol. 88, pp. 324–331, 2017.

[133] A. J. Eele and A. Richards, "Path-Planning with Avoidance Using Nonlinear Branch-and-Bound Optimization," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 2, pp. 384–394, 2009. [Online]. Available: http://arc.aiaa.org/doi/10.2514/1.40034

[134] R. Dai, U. Lee, S. Hosseini, and M. Mesbahi, "Optimal path planning for solar-powered UAVs based on unit quaternions," *Proceedings of the IEEE Conference on Decision and Control*, pp. 3104–3109, 2012.

[135] A. Richards and J. How, "A Decentralized Algorithm for Robust Constrained Model Predictive Control," *Aerospace*, pp. 4261–4266, 2004.

[136] A. Richards and J. P. How, "Robust distributed model predictive control," *International Journal of Control*, vol. 80, no. 9, pp. 1517–1531, 2007.

[137] N. Bo, X. Li, J. Dai, and J. Tang, "A Hierarchical Optimization Strategy of Trajectory Planning for Multi-UAVs," *2017 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, pp. 294–298, 2017. [Online]. Available: http://ieeexplore.ieee.org/document/8047633/

[138] H. Chao, Y. Cao, and Y. Chen, "Autopilots for small unmanned aerial vehicles: A survey," *International Journal of Control, Automation and Systems*, vol. 8, no. 1, pp. 36–44, 2010.

[139] G. Conte and P. Doherty, "An integrated UAV navigation system based on aerial image matching," *IEEE Aerospace Conference Proceedings*, 2008.

[140] Z. Ai, M. A. Livingston, and I. S. Moskowitz, "Real-time unmanned aerial vehicle 3D environment exploration in a mixed reality environment," *2016 International Conference on Unmanned Aircraft Systems, ICUAS 2016*, pp. 664–670, 2016.

[141] J. C. Mohanta, D. R. Parhi, S. R. Mohanty, and A. Keshari, "A Control Scheme for Navigation and Obstacle Avoidance of Autonomous Flying Agent," *Arabian Journal for Science and Engineering*, vol. 43, no. 3, pp. 1395–1407, 2018.

[142] J. Lee, J. Wang, D. Crandall, S. Sabanovic, and G. Fox, "Real-time, cloud-based object detection for unmanned aerial vehicles," *Proceedings - 2017 1st IEEE International Conference on Robotic Computing, IRC 2017*, pp. 36–43, 2017.

[143] E. Holzbecher, *Environmental Modeling*, 2nd ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. [Online]. Available: http://link.springer.com/10.1007/978-3-642-22042-5

[144] D. Hollenbeck, G. Nunez, L. E. Christensen, and Y. Chen, "Wind Measurement and Estimation with Small Unmanned Aerial Systems (sUAS) Using On-Board Mini Ultrasonic Anemometers," *2018 International Conference on Unmanned Aircraft Systems, ICUAS 2018*, pp. 285–292, 2018.

[145] L. Rodriguez, J. A. Cobano, and A. Ollero, "Wind characterization and mapping using fixed-wing small unmanned aerial systems," *2016 International Conference on Unmanned Aircraft Systems, ICUAS 2016*, pp. 178–184, 2016.

[146] F. J. Liu, P. H. Chen, S. S. Kuo, D. C. Su, T. P. Chang, Y. H. Yu, and T. C. Lin, "Wind characterization analysis incorporating genetic algorithm: A case study in Taiwan Strait," *Energy*, vol. 36, no. 5, pp. 2611–2619, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.energy.2011.02.001

[147] D. B. Wilson, A. H. Göktoğan, and S. Sukkarieh, "UAV rendezvous: From concept to flight test," *Australasian Conference on Robotics and Automation, ACRA*, pp. 3–5, 2012.

[148] T. W. McClain, P. R. Chandler, S. Rasmussen, and M. Pachter, "Cooperative control of UAV Rendezvous," in *Proceedings of the American Control Conference*, 2001.

[149] D. Lee and D. H. Shim, "RRT-based path planning for fixed-wing UAVs with arrival time and approach direction constraints," *2014 International Conference on Unmanned Aircraft Systems, ICUAS 2014 - Conference Proceedings*, pp. 317–328, 2014.

[150] S. Calder, "Guatemala Volcano: Airport Closedf and Travellers Dtranded as Eruption Prompts Major Travel Warning," 2018. [Online]. Available: https://www.independent.co.uk/travel/news-and-advice/

guatemala-volcano-fuego-airport-closed-travel-warnings-volcanic-ash-a8382071.html

[151] J. Oliver, "Standard Atmosphere," 1987. [Online]. Available: https://link.springer.com/referenceworkentry/10.1007{%}2F0-387-30749-4{_}167

[152] N. C. Lautze, J. Taddeucci, D. Andronico, C. Cannata, L. Tornetta, P. Scarlato, B. Houghton, and M. D. Lo Castro, "SEM-based methods for the analysis of basaltic ash from weak explosive activity at Etna in 2006 and the 2007 eruptive crisis at Stromboli," *Physics and Chemistry of the Earth*, vol. 45-46, pp. 113–127, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.pce.2011.02.001

[153] A. V. Vaca, C. R. Arroyo, A. Debut, T. Toulkeridis, L. Cumbal, F. Mato, M. C. D'Howitt, and E. Aguilera, "Characterization of fine-grained material ejected by the cotopaxi volcano employing X-ray diffraction and electron diffraction scattering techniques," *Biology and Medicine*, vol. 8, no. 3, pp. 2015–2017, 2016.

[154] T. J. Jones, K. McNamara, J. Eychenne, A. C. Rust, K. V. Cashman, B. Scheu, and R. Edwards, "Primary and secondary fragmentation of crystal-bearing intermediate magma," *Journal of Volcanology and Geothermal Research*, vol. 327, pp. 70–83, 2016. [Online]. Available: http://dx.doi.org/10.1016/j.jvolgeores.2016.06.022

[155] F. Pardini, M. Queißer, A. Naismith, I. M. Watson, L. Clarisse, and M. R. Burton, "Initial constraints on triggering mechanisms of the eruption of Fuego volcano (Guatemala) from 3 June 2018 using IASI satellite data," *Journal of Volcanology and Geothermal Research*, vol. 376, pp. 54–61, 2019.

[156] M. Rüedi, "Flight Zoomer: Automatic Landings / ILS Approaches," 2020. [Online]. Available: https://flightzoomer.com/manual/hfw{_}automatic-landings-{_}-ils-approaches.html

[157] M. Protti and R. Barzan, "UAV Autonomy – Which level is desirable ? – Which level is acceptable ? Alenia Aeronautica Viewpoint," in *Platform Innovations and System Integration for Unmanned Air, Land and Sea Vehicles (AVT-SCI Joint Symposium)*, 2007, pp. 1–12.

[158] R. B. Stull, *Meteorology For Scientists And Engineers*, 2nd ed., 2005.

[159] R. M. Thomas, A. R. Mackenzie, S. J. Reynolds, J. P. Sadler, F. Cropley, S. Bell, S. J. Dugdale, L. Chapman, A. Quinn, and X. Cai, "Avian sensor packages for meteorological measurements," *Bulletin of the American Meteorological Society*, vol. 99, no. 3, pp. 499–511, 2018.

[160] U.S. Department of Interior, "Oblique Photogrammetry to Monitor Volcanoes," 2013. [Online]. Available: https://eros.usgs.gov/doi-remote-sensing-activities/2013/oblique-photogrammetry-monitor-volcanoes

[161] CAA, "Unmanned Aircraft System Operations in UK Airspace – Guidance," *Cap 722*, no. 5, p. 165, 2015.

[162] NASA, "NASA JPL Shuttle Radar Topography Mission," 2019. [Online]. Available: https://www2.jpl.nasa.gov/srtm/

[163] Various (Open Source), "MAVLINK Common Message Set," 2020. [Online]. Available: https://mavlink.io/en/messages/common.html

[164] D. F. Anderson and S. Eberhardt, "Understanding Flight," in *Understanding Flight*, 2nd ed. McGraw Hil, 2010, ch. 6, Appendi, pp. 163 – 182, 268 – 269.