ABSTRACT

Title of Thesis: DISTRIBUTED SEARCH METHOD FOR TEAMS OF SMALL UNMANNED AIRCRAFT SYSTEMS

Jacob D Moschler, Master of Science, 2018

Thesis directed by: Professor John S. Baras, Department of Electrical and Computer Engineering and ISR

We apply Model Based Systems Engineering (MBSE) methods to develop requirements for unmanned aircraft systems (UAS) use cases across industries and create new path planning algorithms for one group of use cases with similar requirements. We then develop and validate models to estimate cost versus data quality for the aforementioned group of use cases. We use our models in conjunction with the MBSE process to plan and execute flights beyond visual line of sight (BVLOS) to scan large areas of remote jungle using small UAS.

DISTRIBUTED SEARCH METHOD FOR TEAMS OF SMALL UNMANNED AIRCRAFT SYSTEMS

by

Jacob Moschler

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
2018

Advisory Committee:
Professor John S. Baras, Chair
Professor Jeffrey Herrmann
Assistant Professor Huan Xu

Acknowledgements

First and foremost, I thank my wife, Thuan Do for her unwavering support and compassion during my time completing this research.

Sincerest thanks are also in order to Dr. John S. Baras for supporting, advising and mentoring me throughought graduate school. Through wisdom, experience and patience, he provided me with the guidance and foundation to complete the research detailed herein, along with much more research outside the scope of this thesis. Unending administrative and academic support was also provided generously by Kimberly Edwards, John MacCarthy, Mark Austin, Yuchen Zhou, and Leonard Petnga at the University of Maryland Institute for Systems Research.

Thanks also to Matthew Scassero and Anthony Pucciarella at the University of Maryland UAS Test Site for sharing their extensive experience and guiding me to execute UAS flights and turn theory into practice, solving messy, real-world problems, and giving me a very good reason to buy a sailboat.

Additionally, I express special thanks to Dr. Sean Downey for providing the opportunity to conduct UAS flights beyond line of sight in Southern Belize, and providing a safe and rich experience operating UAS in the jungle there.

Lastly, I express deep appreciation for all the LaTeX typesetting help provided by Nicole Cuillerier.

Contents

Acknowledgements ii

List of Figures viii

1 Use Case Analysis 2

1.1 Traditional MBSE Approach 3

1.2 Fire and Emergency Management System Stakeholders 4

1.2.1 Fire / EMS Requirements Workshop 4

1.2.1.1 Recommendations 5

1.2.1.2 Safety 6

1.2.1.3 Data Security 7

1.2.1.4 Test Methods 7

1.2.1.5 Surveillance and Recon abilities 8

1.2.1.6 Airborne Portable Repeater Use Case 8

1.2.1.7 EMS Specific use cases 9

1.2.1.8 Fire Specific use cases 10

1.2.1.9 Public Safety Special Operations use cases . 11
1.2.1.10 Procurement and Support 12
1.2.1.11 Software resources 12
1.2.1.12 Future Work Involving Public Safety UAS . 12
1.2.1.13 Life Preserver Delivery Use Case 14
1.2.1.14 Development Environment 14
1.2.1.15 Current State of the Art of Water Rescue UAS 15
1.2.1.16 Challenges 16
1.2.1.17 Requirements and Industry Input 18
1.2.1.18 Hardware 19
1.2.1.19 Test Flight 21
1.2.1.20 Search 22
1.2.1.21 Drop 22
1.2.1.22 Future Water Rescue Work 23
1.3 Discovering Parallels Across Use Cases by Grouping Requirements 24
1.3.1 Package Delivery and Parallel Use Cases 25
1.3.1.1 Package Delivery Variants 25
1.3.2 Area Search and Parallel Use Cases 26
1.3.3 Conclusion 27

2 Path Planning for Mapping and Sensor Coverage 29
2.1 Area Search Use Cases 29
2.2 Literature Review 30

2.2.1 Approaches to Heuristic Path Planning 30

2.2.2 Existing Sensing Capabilities 31

2.2.3 Approaches to Low-Level Control 32

2.2.4 Approaches to Formal Safety Verification 32

2.3 High Level System Description 33

2.3.1 Use Case Diagram 34

2.3.2 System Architecture 34

2.4 Algorithm for High Level Heuristic Path Planning 37

2.5 Low-Level Control Using Potential Fields 40

2.6 Algorithm to Account for Wind During Large Area Scanning With UAS 45

2.6.1 Algorithm Description 47

3 Models **54**

3.0.1 Sensor Coverage Cost and Schedule Model 55

3.0.2 Inputs 55

3.0.2.1 Outputs 56

3.0.3 Test Flights and UAS Model Validation 57

3.0.4 Weather Model for Southern Belize 59

4 Case Study Using our MBSE Process **61**

4.1 MBSE Approach to a Mapping Problem in Belize 61

4.1.1 Operational Context 62

4.1.2 Motivation 63

4.2 Regulatory Stakeholder Perspective 63

4.2.1 Risks Common to All UAS Use Cases in the NAS . . 64
4.2.2 Public Trust in Automation as Context for UAS Automation 67
4.2.3 Developing safety requirements 69
4.2.4 Process for Documenting Airworthiness Approval . . . 70
4.2.5 Operating in Close Proximity to Large Passenger Aircraft 71
4.2.6 Operating Beyond Line of Sight Near Passenger Aircraft 72
4.2.7 Mitigating Risks Specific to the Belize Use Case . . . 72
4.2.8 Constraints specific to the Belize jungle scanning project 73
4.2.9 Requirements specific to the Belize jungle scanning project 73
4.3 Satisfying of Requirements and Constraints Specific to the Belize Use Case 74
4.3.1 Platform Selection for Belize Use Case 74
4.3.2 Functions of Common Flight Controllers 75
4.4 Trade Space 76
4.4.1 Design Parameters for Large Area Scanning System . 76
4.4.2 Best Practices for Vegetation Scanning 77
4.4.3 Sensor Selection 78
4.4.4 Sensor Testing 79
4.4.5 Processing Sensor Data 80
4.4.6 Tradeoffs 81
4.4.7 Scanning Platform 82

4.4.8 UAS Configuration 82
4.4.9 Range Testing 87
4.4.10 System Operation 87
4.5 Lessons Learned 88
4.6 Conclusion 89

List of Figures

1.1 DJI S-900, University of Maryland 15

1.2 DJI S-1000 using landing gear to drop a PFD, Associated Press 16

1.3 Phoenix 60 UAS with Dragonview Sensor, UMD 19

1.4 Communication of MOB1 Locator, Oceansignal.com 20

1.5 PFD Drop Mechanism, designed and built by the author . . . 21

1.6 Normal and infrared images of rescue personnel on a boat during a cold day, taken by the author 22

1.7 Our UAS dropping a self-inflating life preserver 23

1.8 Photo of a NASA UAS piloted by the author carrying experimental sensors above a fire during simultaneous overflight of VIIRS-NPP satellite 26

2.1 Use Case Diagram for UAS 34

2.2 Block Definition Diagram of Aircraft 35

2.3 Integrating existing air traffic detection systems within the concept of synergistic planning 36

2.4 High Level Heuristic Planning Pseudocode 37

2.5 Heuristic Patrol Algorithm 38

2.6 Visualization of Dynamic Surface 44

2.7 Visualization of Dynamic Surface 44

2.8 Heuristic Algorithm to Cover Large Areas with Multiple UAS or Many Flights with One UAS 48

3.1 Our model for sensor coverage and flight times of various UAS and Sensor configurations, built around a simplistic geometric sensor model by Pix4d 57

3.2 Flight paths during model validation and sensor test flights over 1000 acres of jungle, captured by the author 58

3.3 Recorded flight path of sysid flight to update model parameters for this UAS platform 59

3.4 We developed this weather model to estimate the percentage of flight days cancelled for weather. El Nio data courtesy of NOAA. 59

4.1 Operational Context of a UAS Scanning System 62

4.2 Airspace Classes 64

4.3 Birdstrike damage to the Second Stage Stators in the Core of a CFM56 Passenger Jet Engine, Taken by the Author 65

4.4 Hazard caused by failure of a UAS power component, taken by the author from a chase UAS 67

4.5 The author serves as a subject matter expert during an airworthiness review of a Lattitude HQ-60 fixed-wing hybrid VTOL UAS 71
4.6 Table of Requirements and Constraints 76
4.7 77
4.8 Micasense Rededge front view. B. Side view. C. Downwelling Light Sensor. D. Dedigated GPS 79
4.9 Wavelengths of the five cameras within the Micasense Rededge Multispectral sensor, image courtesy of Micasense 79
4.10 A single capture of raw sensor data with corresponding wavelengths marked below 80
4.11 Corresponding data from the visible spectrum 80
4.12 Post-processing Micasense Rededge data collected by the author 81
4.13 Flight paths during model validation and sensor test flights over 1000 acres of jungle, captured by the author 82
4.14 Firefly6 Pro Hybrid UAS in hover mode, taken by the author 83
4.15 Equipment loaded for the final leg of transportation into the jungle 88

Introduction

We propose and evaluate the following four key contributions to model based systems engineering (MBSE):

1. A process for grouping use cases with common requirements through stakeholder interviews across industries.

2. Algorithms to provide path planning for a group of use cases that involve scanning a large area using multiple UAS or many flights of a single UAS while taking wind into account.

3. Models to estimate time and cost to scan a large area with multiple UAS or many flights of a single UAS given sensor specifications and flight dynamics.

4. Lessons learned while using MBSE processes including tradeoff analysis for a project in which we flew small UAS beyond visual line of sight (BVLOS) to scan a large area of remote jungle in Central America.

Chapter 1

Use Case Analysis

We investigate a variety of use cases for UAS (unmanned aircraft systems) across multiple industries. We hold discussions with industry stakeholders and search for use cases common across industry sectors. We perform a variety of case studies and partner with industry and government to develop and operate UAS for numerous test cases. We first follow a traditional systems engineering approach to develop use cases and requirements through discussions with stakeholders. We propose a methodology of grouping use cases developed during conversations with stakeholders based on similarities between requirements discovered in unrelated projects. Lastly, we propose metrics for measuring this contribution to MBSE processes and evaluate its potential costs and benefits.

1.1 Traditional MBSE Approach

Availability of inexpensive MEMS gyros and embedded systems along with open source flight control software products have paved the way for widespread adaptation of inexpensive UAS to serve all industry sectors. We conducted meetings and interviews with stakeholders in construction, anthropology, agriculture, forestry, public safety, and weather forecasting. Some stakeholders already employed manned aircraft and/or utilized remote sensors to collect data. We held lengthy discussions with engineers at NOAA which allowed us to develop inexpensive UAS to improve satellite calibration for the GOES-R/GOES-16 satellite program. Other stakeholders we met with from the fields of forestry and anthropology had previously used mobile phones held out the window of a Cessna to collect aerial data. Tiffany Shorten, manager of a nursery in Maryland, explained to us that using UAS to measure and count plants could cut Maryland plant nursery industry management effort by 40%. Dr. Sean Downey, an anthropologist studying Mayan culture in Belize,] needed to collect quantitative data on land use and jungle recovery from rural farming methods. US Navy engineers wanted to use UAS as communications relays to greatly extend the operating range of their existing EOD (explosive ordinance disposal) ground robots. Water rescue teams in Charles County, MD wanted the ability to deliver life preservers and also to quickly search large areas of the southern Potomac River for the bodies of drowning victims.

1.2 Fire and Emergency Management System Stakeholders

In recent years, UAS have been increasingly pursued as a tool to improve the efficiency and safety of fire and emergency medical service providers in the United States. UAS use cases in this industry vary widely, to include wildfire monitoring, and search and rescue. We begin exploring the requirements of this industry by meeting with regional experts in the field. We then conducted multiple experiments to develop UAS for water rescue disaster and automobile accident reconstruction use cases.

1.2.1 Fire / EMS Requirements Workshop

On April 28th, 2015, Fire and EMS leaders in the National Capital Region collaborated with academic and industry experts in UAS technology comprised of engineers, policymakers and safety experts to conduct a summit and workshop organized by the University of Maryland UAS test site. The purpose of this workshop was to develop UAS requirements that will support the development of standards, policies, and best practices in the national public safety community. We present the results of our Fire and EMS workshop herein.

Academics and engineers are quickly developing UAS to meet Fire and EMS needs. However, many systems are still lacking capabilities that are needed for Fire and EMS use cases. The UMD UAS Test Site worked with the Chief of Charles County Emergency Services and the Battalion Chief of Special Operations for the Prince Georges County Fire/EMS to bring together Fire and EMS experts from the National Capital Region and the research expertise of the University of Maryland to help develop and define UAS requirements that seek to solve real-world Fire and EMS challenges. Researchers and UAS Subject Matter Experts listened to the needs of the Fire and EMS community and facilitated discussions focused on evaluating gaps in current Fire/EMS capabilities and identifying areas where UAS could provide the greatest support and impact. Fire, EMS, and Special Operations and Communications cases were discussed and are further described below. Experts in our working groups outlined specific concerns about operating UAS within their agencies. Training programs for many UAS platforms currently being adopted are lacking or non-existent. Potential operators expressed concern that communication with UAS could be lost and fail-safes and return-to-home modes may not function properly every time. Future UAS operators also voiced concerns over security. They stressed that video and data should be encrypted whenever possible and should only be analyzed by the agency operating the UAS that collected the data. Stakeholders indicated that payload and flight time constraints are prohibitive to many missions and that flight time should be maximized. They expressed concern about operation in various weather conditions as well as overall system durability.

1.2.1.1 Recommendations

Use cases and requirements were listed after brainstorming sessions between industry experts, academics and Fire/EMS leaders. Use cases were split among the categories of Fire, EMS, and Special Operations and Communications. The following list divides UAS into five platform-types suitable to fulfill the various requirements discussed herein:

- Low-cost, disposable UAS for inspecting contaminated areas
- Short-range; used on scene
- Long-term loiter (in some cases with tether)
- Wide area search; beyond line of sight
- Long-range, high altitude

The following list of requirements highlights needs that were commonly expressed by all groups:

- Low maintenance needs with hot swap replacement parts
- Durable and all-weather systems
- Quick deployment/response time
- Simple operating procedures
- Encrypted real time data collection and live feed capabilities
- Single platform design with swappable multi-mission accessories/devices.

1.2.1.2 Safety

Industry experts and engineers also provided safety recommendations for Fire and EMS use of UAS. Intrinsic safety (limiting sparks and ignition sources in hazardous areas) was emphasized by each working group independently.

Training resources for crews must be regulated to include safe operation in disasters, interagency UAS cooperation, and airspace management and compliance with local and national laws.

1.2.1.3 Data Security

During the workshop, emergency response leaders strongly advised all data collected by UAS should be safeguarded. Video and data transmissions should be encrypted whenever possible and data should be analyzed and interpreted within the agency operating the UAS rather than a third party.

1.2.1.4 Test Methods

Currently, not enough data is available to first responders and policymakers regarding the capabilities of existing platforms. Informed decisions cannot be made by policymakers regarding the risk/benefit of different size classes of UAS without understanding the capabilities. A comprehensive set of performance metrics and test methods for UAS capabilities should be created based upon previous standards development work and first responder input. Performance metrics and associated test methods are generally useful for several different kinds of emergency responder applications and a good start toward a comprehensive suite of standard test methods to measure safety and performance of such systems. Current aspects of performance that are critical to the first responder community include, but are not limited to the following:

- Lost link behavior
- Lost power behavior
- Station-keeping near a target (horizontal and vertical)
- Visual acuity
- Endurance
- Impact Forces (drop test)
- Communications protocol
- Vehicle performance when exposed to high heat, heavy smoke and water

1.2.1.5 Surveillance and Recon abilities

The surveillance and reconnaissance use case was listed prominently by all working groups (Fire, EMS, and Special Operations and Communications). This use case requires real-time monitoring and streaming live video data. Thermal data to recognize humans and analyze fires is also highly desired. Vehicles may operate in this use case during initial scene evaluation, investigating and surveying dangerous environments and threat matrix analysis.

1.2.1.6 Airborne Portable Repeater Use Case

Both Fire and EMS working groups emphasized a use case of portable repeaters for wireless data, mobile phone service and local radio communication. Mobile phone networks can be degraded or destroyed entirely during public safety crises. Even in normal situations, high rise buildings often cause communication radios to operate poorly. This use case may not occur often, but systems meeting this need will be of very high value when the need inevitably does arise. Systems with this capability would need to operate for extended periods of time. In some cases, these systems may only need to loiter in one location, therefore they may operate with a tether.

1.2.1.7 EMS Specific use cases

In addition to the aforementioned surveillance and airborne portable repeater use cases, the experts in our EMS working group developed five key ways in which UAS could be used to enhance the existing abilities of EMS teams. These uses are as follows:

- Critical Event Pre-Planning for an Incident Action Plan (IAP)
- Damage assessment after a natural disaster or contamination event
- Indoor surveillance for safety assurance
- Search and recovery of missing persons

Discussion of these use cases led to a list of general requirements for UAS operated by EMS. Requirements are listed as follows, not in order of priority or emphasis.

- Integrate UAS with existing Mobile Data Terminal and GIS data systems
- Minimize the need to transport extra hardware when operating UAS

- Include easy-to-use grid search function for UAS to search an area
- Include simple go to target function
- Allow moderate camera sensor zoom from an altitude of 500 feet
- Prevent UAS from creating any spark (required for some hazmat operations)

1.2.1.8 Fire Specific use cases

Firefighting experts described a variety of use cases in addition to the surveillance and airborne portable repeater use cases. Firefighting use cases are as follows:

- Autonomous investigation of a non-emergency location and returning to base
- Deployment of a five pound payload to a specified location
- Traffic control and resource allocation, including traffic situational awareness, locating chokepoints, routing traffic flow and choosing signals to change

These use cases were developed into requirements through our discussions with engineers and firefighting experts. The requirements are listed as follows, not in order of priority or emphasis.

- Package a simple system in one pelican case
- Autonomously deploy, perform a simple mission and return home with one button press

- Allow multiple, modular payloads
- Allow flight beyond line of sight
- Fly in all weather conditions and smoke
- Refuel or replace battery quickly on site
- Provide ultra-stable control for adverse weather
- Allow control by a single operator-observer

1.2.1.9 Public Safety Special Operations use cases

A Special Operations working group focused on operations not always covered by standard Fire and EMS units. The Special Ops team developed the following use cases:

- Search of a remote area, distant from the insertion point
- Surveillance of a disaster area providing GIS data and inspecting behind rubble
- General enhancement of existing ground-centric operations
- Radiological and chemical sensing
- Investigation of dangerous environments
- Provide initial scene evaluation into a threat matrix

1.2.1.10 Procurement and Support

Fire and EMS leaders expressed a need for service contracts for on-site maintenance of UAS. Our teams of experts also recommended federal grant programs as initial sources of funding. UAS purchased from these funds would need to be on the AEL (Approved Equipment List).

1.2.1.11 Software resources

Our experts identified several ways in which software could complement the use of UAS by Fire and EMS personnel. This software may be developed and supplied separately from the UAS as long as it is properly integrated and supported.

- Risk-based analysis of what platforms and software tools should be used
- Integration of UAS with existing computer-aided dispatch systems
- Capability to identify friendly UAS, ground units and personnel

1.2.1.12 Future Work Involving Public Safety UAS

The working groups at our event finished with requests for action from regulators and other agencies. Policy makers and the general public must be educated about the aforementioned use cases of UAS. A successful public awareness initiative should convey the public health and safety benefits of UAS for firefighting and emergency response in contrast with the more controversial UAS use cases that have been highlighted by the media.

A model Fire and EMS policy for operating UAS must be generated to give agencies a starting point to ensure compliance with regulations and best practices. This model policy must be repeatable and general enough to be quickly implemented by a variety of Fire and EMS agencies. A national repository of best practices, policies and procedures should be established for Fire and EMS agencies to reduce overhead needed to develop policies for operating UAS. This policy repository may also include law enforcement UAS policies and best practices. Recurrent training requirements for system operation must be set by regulators. Training to include crew resource management role management for supervisors, administrators and other roles should be implemented.

The experts at our workshop also felt that the FAA could support Fire/EMS use of UAS by creating mechanisms to prioritize airspace access by Fire and EMS agencies. This mechanism should dynamically inform operators of airspace restrictions in near-real-time. Regulators should also generate a mechanism to initiate a shutdown of all nonessential commercial and hobby UAS so that Fire and EMS may operate freely and safely during emergencies. Lastly, we proposed that the FAA should create a channel for Fire and EMS operators to request an emergency Certificate of Waiver or Authorization (COA) with an activation time of fifteen minutes or less, with provisions for immediate flights during situations of duress and imminent threat. The FAA has since created channels for exactly such scenarios.

In conclusion, we presented regulators and agency leaders with immediate opportunities to improve public safety by working together to enact the above recommendations as quickly and thoroughly as possible. We also provide UAS designers and vendors with the aforementioned use cases and requirements. UAS technology has since been adopted by Fire and EMS agencies across the US.

1.2.1.13 Life Preserver Delivery Use Case

An estimated 360,000 people die of unintentional drownings each year [1]. Many drownings occur at public beaches, however ocean going vessels also present a danger when passengers and crew fall overboard. This is a particularly sensitive problem for cruise ships and offshore yacht racing. Out of 27 passengers and crew who fell overboard from cruise ships in 2015, only 7 were rescued alive [2]. Thus an immediate demand exists for reliable solution that could provide aerial search and rescue in these situations. Unmanned aerial systems are gaining popularity as a tool to aid drowning victims by quickly delivering personal floatation devices (PFDs) [3]. Water rescue UAS are most commonly flown by a beach lifeguard using manual control, however current state of the art hardware and software allows lifesaving systems to be fully automated. Furthermore, falling prices of COTS UAS components create an ideal environment for widespread use of autonomous water rescue aircraft.

1.2.1.14 Development Environment

Water rescue UAS are quickly gaining traction in open-source development communities thanks to the common ground open-source developers find with greater-good, humanitarian systems [**?**]. Public support for water rescue adaptations of UAS technology is also conducive to successful crowdsourcing campaigns via Kickstarter.com and gofundme.com.

1.2.1.15 Current State of the Art of Water Rescue UAS

Current water rescue UAS are typically manually-operated at busy beaches to reach drowning victims more quickly than lifeguards can swim to victims or launch a rescue boat. Lifeguards perform their standard rescue process, using a second rescuer to send a life preserver as a stopgap response while the first lifeguard is in transit. Rescuers have used popular multirotor UAS such as the DJI S-1000 and S-900 (figure 1.1) in successful water rescue operations.

, 

FIGURE 1.1: DJI S-900, University of Maryland

First-person-view (FPV) cameras onboard the aircraft allow the operator and observers to search for victims via handheld video monitors. Downward-facing cameras allow the operator to manually position the aircraft so that the floatation device will fall within reach for a successful rescue. The retractable landing gear of the S-1000 and S-900 allows the operator to command a PFD drop via a simple physical modification that attaches rescue flotation rings to the aircraft landing gear. When the operator raises the landing gear, the rings are dropped below. Thus the landing gear becomes a COTS drop mechanism, as shown in figure 1.2.

,

FIGURE 1.2: DJI S-1000 using landing gear to drop a PFD, Associated Press

1.2.1.16 Challenges

Multiple challenges exist, preventing COTS UAS from executing water rescue missions without careful consideration to address various problems. The most important factor when considering water rescue is that it only takes a person 20-60 seconds to drown [4]. In contrast, most COTS UAS take 10-20 minutes to set up and launch. Professional grade systems operated by large organizations may take even longer due to complex procedures and flight crew requirements that arise from concerns about operator error.

Fully autonomous water rescue systems could drastically improve launch times. However, many improvements would need to be made in cost, speed and complexity to make autonomous systems reliable enough for rescue operations. Covered COTS launch pads with built-in battery chargers are available that would allow aircraft to remain fully charged and ready to automatically launch at a moments notice [5]. Most UAS do have a standby state suitable for awaiting a launch command in a covered launch pad, but typical UAS require frequent gyro and compass calibration. These calibrations involve entering calibration mode and rotating the aircraft on each axis for around 30 seconds. Thus more research into COTS MEMS gyro and compass calibration is required before a set it and forget it UAS will be available for water rescue.

Launching water rescue UAS from a boat adds complexity to the standard failsafe parameter of a home point where the UAS is programmed to return in case of lost link or low battery. IR beacons or other precision landing zone indicators can be added, but this adds another point of failure. Another solution for this problem could be to design the UAS itself to be the flotation device and cause it to land in the water near the victim thus eliminating the need to fly back to the launch point. Waterproof COTS UAS might be suitable for such applications.

Another challenge with water rescue systems is locating the victim. Even small ocean swells can obscure a victim from view. When a person falls off a boat, standard man overboard procedure is to immediately assign multiple crew members to maintain visual contact with the victim. Autonomously locating the victim adds to this challenge, as computer vision may have to pick out the victim among floating debris and other flotation devices thrown from the vessel in the standard man overboard procedure.

Environmental conditions present perhaps the most significant challenge, where strong winds, rain, heavy seas and darkness can make rescue and operation of UAS more difficult. Manufacturers of the UAS we tested in this scenario recommend that the systems only be operated in winds below 10 m/s. However, our preliminary tests of similar aircraft have indicated that the systems should be able to launch, traverse and hover in winds in excess of 18 m/s. However, some floatation devices could add more surface area and severely reduce aircraft performance in high winds. Due to this issue, we opted to use compact, self-inflating life preservers in our test. The Phoenix 60 UAS was the only aircraft we tested capable of flying in heavy rain. Ocean conditions with winds blowing 18 m/s are considered Beaufort Force 8, where wave height would be near 6 meters with breaking crests and streaks of foam and would thus make locating the victim a considerable challenge [6].

Lastly, some life preservers may be heavy so steps must be taken to avoid traumatic injury when they are dropped from above. Any hard or sharp edges of compact and throwable PFDs must be covered in soft foam so that they do not injure the victim when dropped from the UAS.

1.2.1.17 Requirements and Industry Input

In most of the US, firefighters provide water rescue as part of their service to the public. In addition to the aforementioned Fire and EMS workshop, we also conducted a separate interview with the Water Rescue division of the Cobb Island Volunteer Fire Department in Charles County, Maryland. Members of the water rescue team advised of multiple challenges, including a significant shortfall of current methods for dropping life preservers. The team advised that while quickly delivering one PFD to a drowning victim was useful, they often respond to boating accidents with multiple victims which are not always collocated. Furthermore, when victims are together, dropping an insufficient number of PFDs could do more harm than good, as even well-intentioned drowning victims will instinctively scramble for anything afloat and can take other victims down with them. Thus a new requirement was discovered to drop multiple PFDs in specific locations instead of dropping them all at one location.

1.2.1.18 Hardware

Two multirotor platforms were chosen to test water rescue coordination between the UMD UAS Test Site team and the Charles County Fire and Rescue Department on Cobb Island, MD. The first platform, intended to search for and locate the victim, was a Phoenix 60 hexarotor aircraft built by UAV Solutions in Jessup, MD, shown in figure 1.3. This aircraft was equipped with a DragonView sensor which included both an electro-optical camera and a Tau 2 Flir long wave infrared (LWIR) thermal sensor.

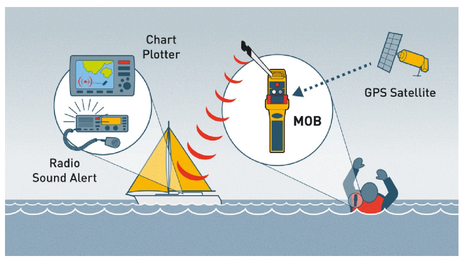

,

FIGURE 1.3: Phoenix 60 UAS with Dragonview Sensor, UMD

To carry and drop the life preservers, we selected a DJI S-1000 aircraft due to its substantial payload capacity of 4.5 kg.

The life preserver chosen for the task was the Rescue Stick, a COTS throwable PFD designed by Mustang Survival. This device was chosen for its compact size when stowed, in order to minimize problems controlling the aircraft in gusty winds. Similar in size and shape to a compact umbrella, the rescue stick automatically inflates using a built-in CO2 tank when it comes in contact with water. It also includes backup methods of inflation by manually triggering the CO_2 as well as a mouth tube.

To aide rescuers in locating the victim, we attached an Ocean Signal MOB1 GPS-enabled personal locating device to our flotation device (Figure 1.4). The trigger of the GPS locator was attached to the inflatable life vest so it would be triggered when the vest inflated upon contact with the water (Figure 1.4).



,

Figure 1.4: Communication of MOB1 Locator, Oceansignal.com

We designed a drop mechanism based on the industry requirement to drop multiple life preservers. A simple release mechanism based on a rubber band gun firing mechanism was downloaded from an online database of 3D printable files, printed and attached to a servo connected to a channel controlled by the pilot (Figure 1.5).

The DJI S-1000 aircraft uses the A2 flight controller and autopilot system for control and some autonomous functions. Like most COTS flight controllers, its autonomous functions include failsafe modes triggered by various scenarios including lost link, GPS failure and low battery.

,

Figure 1.5: PFD Drop Mechanism, designed and built by the author

1.2.1.19 Test Flight

We performed a flight test of our water rescue system in collaboration with Charles County Fire and Rescue department. We operated our Phoenix 60 aircraft equipped with a LWIR thermal sensor to locate the victim and our DJI S-1000 equipped with a life preserver drop mechanism to deliver a flotation device. Charles County Fire and Rescue supplied a fire boat crewed by water rescue personnel and provided a rescue training mannequin as a simulated drowning victim.

The mannequin was dropped into the water off Cobb Island, MD. A preflight inspection was performed and the Phoenix 60 aircraft was launched from shore to search for the victim. Stakeholders and subject matter experts observed and offered insight during the flight. News media documented the test on the front page of the local paper. Cameras were attached to the aircraft and ground station to record video of the test procedure.

1.2.1.20 Search

Our experiments with thermal cameras confirmed that it is difficult to distinguish a victim from surrounding water using a standard thermal camera such as the Tau 2 Flir camera which is built into the Dragonview sensor. Figure (1.6) shows a thermal image and a normal image of a rescue boat taken seconds apart, during the experiment.

,

Figure 1.6: Normal and infrared images of rescue personnel on a boat during a cold day, taken by the author

Personnel from NOAAs unmanned aircraft team were also present for the test and advised that standard cameras were most useful for locating overboard victims with the aid of lights at night. They advised that they had also tested thermal cameras and found little to no benefit over traditional cameras during man overboard searches.

1.2.1.21 Drop

The drop mechanism operated as expected, and the self-inflating life preserver deployed normally. The drop was recorded by a camera mounted on the aircraft, shown in Figure 1.7. Wind and current directions were taken into account so that the life preserver would drift towards the simulated victim. The life preserver included a padded area around its metal CO_2 inflation tank to prevent injury in case the life preserver struck the victim during the drop.


,

FIGURE 1.7: Our UAS dropping a self-inflating life preserver

The personal locating device attached to the life vest was triggered and behaved normally, however its signal was not received by rescue personnel due to the design of the MOB1 locating device. The locating device is designed to be associated with one specific boat and configured to trigger that boats radio to send a "man overboard" alert. Thus for this COTS locator device to provide the necessary information to rescuers, the aircraft or ground station must be integrated with a marine VHF radio capable of sending a standard man overboard alert signal and location to rescuers.

1.2.1.22 Future Water Rescue Work

Human interaction with the life preserver drop aircraft presented the greatest contribution to delay in getting a flotation device to the victim simulated in the water rescue test flights. The COTS flight controller is capable of autonomous launch and return. A simple computer vision system could be added to detect a victim and trigger a life preserver drop automatically.

Autonomous launch of UAS from integrated ship-mounted charging enclosures will enable crews on small ships such as racing sailboats to focus on reducing sail and turning the boat in order to return to pick up crew from the location where they went overboard. Inexpensive infra-red beacons integrated into life preservers could be easily located autonomously and lightweight paracord could be attached to life preservers so that crew could be pulled back to the position of the boat after the preserver was dropped by the aircraft. Aircraft could be equipped with LED spotlights to assist in locating victims during night rescue.

Open source libraries of 3d designs such as Shapeways.com can be used to make drop mechanism designs for new COTS UAS available for emergency services personnel and maritime crews to download immediately for 3D printing by technicians or local library personnel, expediting field deployment of new designs.

1.3 Discovering Parallels Across Use Cases by Grouping Requirements

It is intuitive to note patterns when analyzing requirements from for use case after use case. When patterns are recognized, requirements for similar functions can be grouped together. In taking extra care to avoid "reinventing the wheel," in our case, "reinventing the algorithm that performs a series of functions." The challenge lies in recognizing a "wheel," or "series of functions" already exists, and meets the requirements of multiple use cases. We accomplish this through stakeholder meetings and discussions with industry experts. The greater the catalog we amassed of requirement groups and sequences of functions, the more likely we were to find subsets of requirements in new use cases that could be satisfied by the same path planning algorithms we had already built to satisfy requirements in earlier use cases.

1.3.1 Package Delivery and Parallel Use Cases

One set of use cases commonly involves delivery of a package to a fixed or moving target, such the aforementioned case where we collaborated with water rescue personnel to deliver a life vest to a floating target. We discuss an experiment in which we modified a COTS UAS to deliver a flotation device in cooperation with water rescue personnel.

1.3.1.1 Package Delivery Variants

A variant of the package delivery use case involves carrying a scientific sensor such as a hyperspectral imager to a predetermined target area or along a specific path and aiming it at a predetermined point to simulate a satellite path or measure reflected wavelengths from various angles, often at the same time a satellite with similar sensors travels overhead. We carried a NASA experimental thermal imager in this way above a large, controlled fire on USDA land in conjunction during overflights of VIIRS-JPSS1, MODIS-Aqua and VIIRS-NPP satellites to aid with sensor improvements (fig 1.8)

The airborne repeater use case described by experts at the Fire and EMS workshop and independently described by Navy EOD personnel also largely parallels the package delivery use case, in that the aircraft must takeoff carrying an airborne repeater "package" and move it to a predetermined

,

Figure 1.8: Photo of a NASA UAS piloted by the author carrying experimental sensors above a fire during simultaneous overflight of VIIRS-NPP satellite

location.

1.3.2 Area Search and Parallel Use Cases

Another prevelant set of parallel use cases revolves around searching a large area for a person or other object of interest. For example, local fire and rescue personnel needed to search an area for missing persons, or law enforcement needed to search an area for missing evidence. This area search use case requires persistent observation of an area by one or more aircraft until the target is located and often subsequently following the object or person of interest in order to continuously update the user of their position. It is discussed extensively in Chapter 3. Another variant of the area search use case commonly involves capturing data over a large area for later use.

This type of mapping is distinct from the object or person search use case in that there are requirements to retain large amounts of data. It is also distinct in that it is typically less constrained by time, allowing for the use of complex post-processing algorithms and analysis of data collected.

Chapter 4 discusses implementation of our algorithms to solve a specific mapping problem, providing context including many unforeseen challenges related to implementation in the field. We demonstrate the usefulness of Model Based Systems Engineering techniques to include stakeholders in critical design decisions while managing complexity. Finally we include lessons learned from beyond line of sight field tests in Belize to propose a new algorithm for scanning large areas that will allow a system of COTS UAS to scan large areas at lower costs than previously possible.

1.3.3 Conclusion

Our extensive meetings with stakeholders in a wide variety of industry sectors allowed us to gain a fundamental understanding of use cases and group them by similarities into package delivery, area search, large area mapping and package delivery. Examining use cases through the perspective of key stakeholders such as public safety personnel and regulators also allowed us to conduct extensive testing within the NAS, including UAS flights in Class C airspace. These flights would have otherwise been prohibited.

Examining each use case and involving stakeholders in accordance with Model Based Systems Engineering techniques allowed us to develop and implement and continuously improve a variety of solutions to the aforementioned use cases. We found that cases with multiple identical requirements are not apparent to most single-industry stakeholders due to the limits of their perspective. For example, an engineer designing a UAS path planning algorithm for bridge inspection may never consider the problem of scanning a large swath of land with a small UAS, and vice versa. But once multiple use cases are known, natural groupings of functional requirements as we described can lead to cost savings in model creation and algorithm development. After implementing our process of grouping requirements, we saw a marked improvement in our design process for modeling and creating new algorithms. This allowed rapid verification and validation as well as faster project timelines and reduced project cost below estimates that were based on building algorithms from scratch.

Chapter 2

Path Planning for Mapping and Sensor Coverage

2.1 Area Search Use Cases

In the previous chapters we explored a wide variety of use cases through partnerships with industry and flight tests related to numerous use cases. In this chapter, we choose the area search use case common to many industries to explore more in depth using a Model Based Systems Engineering approach. Our method for grouping use cases established that the UAS sensor coverage problem is found in many industries from search and rescue, to agriculture scanning, to airport security. Many off the shelf UAS software products provide simple approaches to this coverage problem. We developed more complex approaches and ultimately provided performance metrics specific to cases of this problem where sensor coverage is provided by many UAS or many flights of a single UAS. Finally, we propose an algorithm to find sets of flight paths that meet the requirements in an effort to automate future scans.

2.2 Literature Review

A study was made of various tools and techniques in order to determine state of the art methods for handling the problem of aerial robotic search. The overall structure of our approach frames the problem as synergistic planning between multiple layers of abstraction, as imagined by Bhatia and Kavraki in their studies of motion planning with complex goals. This multi-layer synergistic approach is shown to improve computational time by an order of magnitude [7].

2.2.1 Approaches to Heuristic Path Planning

We investigated the use of various methods to create a heuristic algorithm for planning and re-planning.

Cellular decomposition is one method that has been used by many high and low level robotic search and patrol algorithms. Cellular decomposition of an underwater area has been shown to provide dynamic coverage by autonomous underwater vehicles [8]. Approximate cellular decomposition is similar to search methods used by insects in which traces of pheromones allow for areas to be efficiently explored and monitored [9]. Virtual corridors can be used in addition to, or in place of cells to cover a large search area. These are in some ways advantageous to decomposing the search area into cells because they allow each agent to retain some data about the known map. Dividing a search area into corridors also allows for detailed safety analysis. Constraint-based reasoning allows these corridors to be kept within the flight envelope of a particular aircraft by minimizing path characteristics such as snap (second derivative of acceleration) [10]. Previous approaches to underwater search have included the use of a multiresolution grid. When performing a local search, the robotic agent may have limited spatiotemporal information so it relies on a local navigation function. Global search can be performed using a different function that uses more abstract information about a larger area. Varying levels of local and global navigation can be managed using potential cost functions [11]. For example, after searching a local area for some time and finding nothing, each individual robot could gradually and automatically transition from its local navigation algorithm towards a global search function.

2.2.2 Existing Sensing Capabilities

For path planning that incorporates traffic collision avoidance, fast-approaching large aircraft can be modeled as moving obstacles or threats to be avoided. Until reliable, small-scale sensors become available, GBSAA (Ground-Based Sense And Avoid) is a reliable method for detecting approaching aircraft [12]. COTS PCAS (Portable Collision Avoidance Systems) are also available and may be integrated with unmanned aircraft as a secondary method to obtain position information about large aircraft nearby with transponders broadcasting ADS-B (Automatic Dependent Surveillance-Broadcast) signals [13]. To sense targets on the ground, onboard local sensing packages can be used by multiple robots to sense and dynamically track targets. Such tracking methods have been thoroughly studied [14]. More advanced turnkey sensing and target tracking systems are also available [15].

2.2.3 Approaches to Low-Level Control

Decentralized path generation and collision avoidance can also be accomplished using potential cost functions. Gradient descent algorithms can give robots a tendency to attract towards a target while repelling obstacles and other robots [16]. Artificial potential fields have also been used to provide collision avoidance to robotic manipulators [17]. Many aspects of the low-level control are hardware specific. We chose to demonstrate our approach on the increasingly popular electric quadrotor hardware platform, which can be configured to carry a useful payload of up to 58 kg or configured to fly for up to 85 minutes [18]. Model Predictive Control has been extensively studied for quadrotor aircraft and shown to improve the Integral Absolute Derivative control signal index for all measured states of flight [19]. Waypoint control has also been successfully implemented for the quadrotor hardware and static obstacles can be avoided even while maneuvering quickly [20].

2.2.4 Approaches to Formal Safety Verification

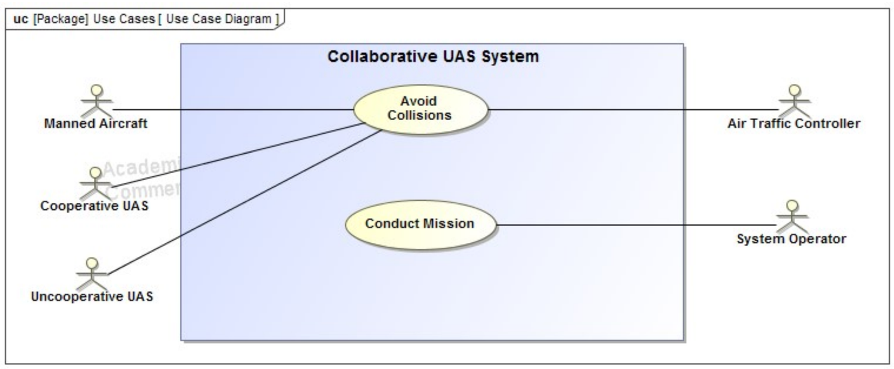
Given a potential gradient and some expectation of the future environment model, one can produce an estimation of the states of the aircraft and the environmental states using the underlying dynamics and expected controller inputs. This generates tubes around each agent of the UAS as well as larger, fast-moving aircraft following scheduled trajectories. To capture modeling uncertainty, sensor noise and unpredictable disturbances such as wind gusts, enlargement of the tubes to consider all uncertainties is essential. The safety verification process then requires computation of the locations of each tube and possible collisions between the tubes. The tubes are formally defined as reachable sets which can be numerically represented using various methods available in the hybrid system analysis community [21] [22] [23] [24] [25] [26] [27] [28] [29]. Reachability analysis for nonlinear systems such as biological systems and autonomous vehicles has been a main theme of recent hybrid systems research [27] [28] [30]. This analysis is increasingly popular because nonlinear systems can be captured using hybridization [30], and methods developed in hybrid system verification can be easily adapted to nonlinear systems [31].

2.3 High Level System Description

We propose a UAS scanning system designed to provide the user with sensor data covering a large area. The coverage time and data quality will be limited by geometry including altitudes flown and the field of view of the sensor. This balance between time and quality can be easily explored by the user with most COTS flight planning tools we tested, although care must be taken to ensure the sensor model used by the COTS tool accurately fits the geometry of the sensor. Raw data quality is typically measured in units of ground area per pixel, commonly referred to as GSD (ground sample distance). With this part of the problem and its related tradeoffs efficiently handled by COTS software, we can focus on the path planning. Our approach below can be configured to output waypoints which are easily followed by a modern COTS UAS, or low level control inputs which can be accepted by ultra low-cost COTS UAS for improved cost savings.

2.3.1 Use Case Diagram

The primary use cases will involve conducting the mission while avoiding collisions, as shown in Figure 2.1.



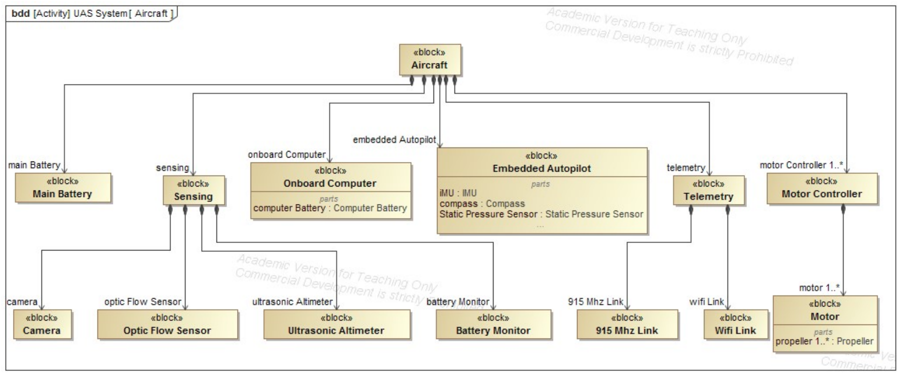
,

FIGURE 2.1: Use Case Diagram for UAS

2.3.2 System Architecture

Autonomous functions require each aircraft to have onboard computing capabilities. Typical small unmanned aircraft systems (SUAS) use an embedded autopilot for critical functions such as returning to the launch point. The embedded autopilot also uses integrated sensors to control stabilization. More advanced SUAS use an onboard computer such as an NVidia TX1 or an Intel Edison for additional processing. Aircraft will have the structure described in Figure 2.2.

We develop new fundamental methodologies for high performance and



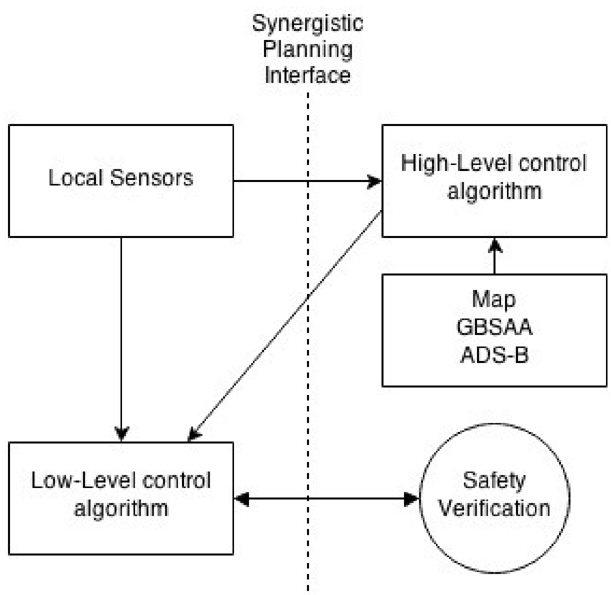
,

FIGURE 2.2: Block Definition Diagram of Aircraft

provably safe autonomous and collaborative control and operation of autonomous unmanned aerial systems (UAS) in the national airspace (NAS), where both UAS and manned aircraft fly. The proposed framework is model-based and emphasizes multiple scales in time and space as well as hybrid systems mathematics to capture both the analog and logical components of control functionalities. We develop on-line control laws that allow for multiple UAS agents to reconfigure to different formations with proofs of safety and convergence while navigating in integrated airspace with piloted air vehicles. We demonstrate our results in simulated scenaria with both cooperative and uncooperative air vehicles.

Our approach is structured with a high-level search algorithm (we use heuristic path planning as an example), a low-level control algorithm, and a formal mathematical verification of the systems safety and ability to avoid collisions, as shown in 2.3. This multilayer framework lends itself well to a wide variety of use-cases and robot tasks, and can be used to develop and analyze future algorithms. We propose this framework for our modeling environment which is modular in that a variety of algorithms, aircraft with varying flight envelopes, and sensor packages can be easily put together and tested.

In our initial approach using heuristic path planning as the high-level search layer, the synergistic interface in Figure 2.3 is a key part of our design process that allows us to integrate high and low-level controllers while maintaining formal safety verification.



,

FIGURE 2.3: Integrating existing air traffic detection systems within the concept of synergistic planning

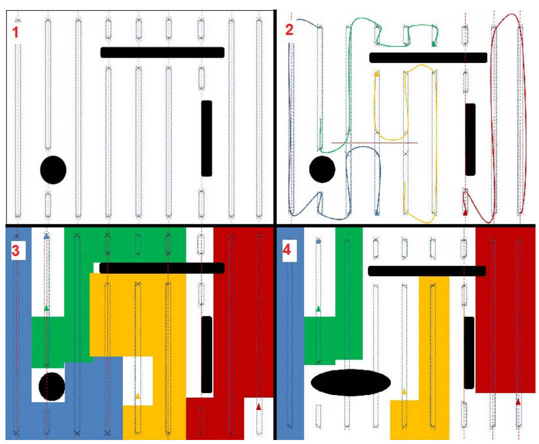
```
get target coverage area
get sensor range
set reference orientation: vertical or horizontal
set multiple UAS agents to arbitrary position in the known target coverage area

while
  if target coverage area is updated then
    do decompose the known target coverage area through reference orientation with defined width equal to sensor range
set corridors along the decomposed lines
    set waypoints at the end points of corridors

    if UAS agents are out of corridor then
      move agents to the nearest corridor
    end
  end

  if two agents are approaching in the same corridor then
    set two waypoints in the middle of the corridor
  end

  snap each agent forward
  record sensor information

  if agent reaches a waypoint then
    move the agent to the next waypoint
  end

  if preset time is elapsed then
    update target coverage area
  end
end
```
,

FIGURE 2.4: High Level Heuristic Planning Pseudocode

2.4 Algorithm for High Level Heuristic Path Planning

To demonstrate, the high-level heuristic planning algorithm begins by accepting input about the target coverage area at an arbitrary time step. The range of onboard local sensors and the desired orientation of decomposition corridors must also be input by the user. To test the algorithm, we arbitrarily place agents of the UAS on the given target coverage area with obstacle positions fixed.

The algorithm then decomposes the target coverage area based on given sensor range and desired decomposition orientation. Separation of decomposition corridors is determined by the radius of sensor range. Agent-to agent collision avoidance redundancy is added by the low-level algorithm and verified by the safety function. Corridors are defined along the decomposed lines with waypoints at the end of each corridor [8] [10]. Waypoints are represented with an x mark at the end of each columns in Figure 2.5.



,

FIGURE 2.5: Heuristic Patrol Algorithm

Initially in the test case, arbitrarily deployed UAS are out of corridors in the target coverage area at a fixed time. The UAS agents automatically move to the nearest waypoints.

Because agents move to the nearest waypoints without any other requirements besides low-level collision avoidance, one corridor may end up with multiple agents. More waypoints are then set in the middle of the corridor. The newly set waypoints indicate to agents where to switch to other corridors so that efficient sensor coverage is achieved without redundant coverage.

When any agent reaches a preset waypoint, it then moves to the next waypoint to continue its patrol. Any time a corridor ends up having multiple agents, the algorithm once again sets new waypoints in the middle of the corridor.

The coverage area updates regularly to reflect latest obstacle information [10]. Decomposition is repeated with the given sensor range and corridors are updated along the decomposed lines. New waypoints are set which may be different from the previous waypoint positions due to movement of dynamic obstacles.

After a corridor has been covered by the system, it will be assigned a coverage decay time. When the coverage area is updated, waypoints will be also created for areas whose coverage has decayed beyond the time limit. The decay times are shown by colored areas in Figure 2.5. This decay creates a continuous patrol state which the map continuously updates. If individual agents are reassigned, become disabled, require maintenance, refueling or recharging, remaining, those agents undergo a state change and are then governed by another algorithm. Remaining agents will distribute themselves to maintain coverage, and collisions are avoided by the low-level control algorithm which still runs on every agent. The coverage decay time may be adjusted by the user for any desired map refresh rate.

2.5 Low-Level Control Using Potential Fields

As described in the overview, our framework also employs a low-level control algorithm for basic target following and traffic collision avoidance.

We describe a hardware-agnostic low-level controller that consists of a minimized potential function. Since this function is implemented above a hardware abstraction layer, it can be applied to any aircraft hardware given that adequate dynamical models and MPC controllers exist for that hardware. The potential function receives information about flight envelope and velocity from separate, hardware-specific models. Taking a gradient descent approach then gives the potential controller a tendency to repel around static obstacles and attract towards a desired path given by the high-level heuristic path planner. This technique has also been shown to enable robust flocking and formation capabilities in decentralized systems [16].

This potential field method of robotic control may seem elegantly simple, however it has been shown to create a risk of the robot becoming trapped in local minima of the potential field [32]. Most prior results only cause convergence to local minima, so the potential field method has been difficult to use for global control of a robot swarm. Several options exist to address the problem of becoming trapped in local minima, making the potential field method more useful globally. Perhaps the most interesting solution to local minima trapping is the adaptation of methods previously used in image processing and computer vision to the control of robotic swarms. It has been shown that Gibbs Random Fields can be used to allow robot swarms to complete global objectives using only local interactions while maintaining the ability to avoid becoming trapped in local minima [33]. A useful tool was also created specifically to address the local minima problem in robotic swarm control using Markov Random Fields, building on the aforementioned Gibbs Random Fields method [34]. These recent solutions are promising ideas for using the potential field method to navigate a global map using only local data. Once modular simulations are integrated as we have proposed, our system will be tested to use the potential method globally for our UAS, possibly eliminating the need for the high-level heuristic method described in the earlier section.

Simplistic use of the potential field method will be attempted first, using input from the high-level heuristic to keep the UAS agents near only desired local minima and keep them from becoming trapped. While not elegant, this simplistic method is sufficient since it only requires convergence to local minima, which most previous approaches using the potential field method have allowed for as long as initial positioning is close enough to the desired position [35] [36]. Global objectives are handled by our high-level heuristic algorithm.

The potential function is designed so that agents could be pushed away from obstacles and the paths of large aircraft using virtual potential fields while being attracted toward its desired path, received from a high-level controller. The potential function of every agent has multiple terms to give the agent input on how to deal with various types of obstacles and targets.

$J_a(\boldsymbol{x}_a) = \sum_{i=1}^{N_a} b_i f(\boldsymbol{x}_a, \boldsymbol{x}_{ai}, \boldsymbol{v}_{ri})^{-1} + g(\boldsymbol{x}_a, \boldsymbol{x}_m, \boldsymbol{v}_m)^{-1} + \|\boldsymbol{x}_a - \boldsymbol{x}_{\gamma a}\|_2^2$

$\boldsymbol{x}_a \in \boldsymbol{R}^3,\ 0$

$\boldsymbol{x}_a$ presents the agent position within 3D position space except the obstacle space . The first term of the potential function is

$$\sum_{i=1}^{N_a} b_i f(\boldsymbol{x}_a, \boldsymbol{x}_{ai}, \boldsymbol{v}_{ri})^{-1}$$

In this expression, a special function, f is used to shape virtual potential fields around other aircraft in the collaborative swarm. b_i is a Boolean variable representing whether the i^{th} agent is close to the host agent a. N_a is the total number of collaborative agents. For the inputs:

Agent position: $\boldsymbol{x}_a$

Position of the ith agent: $\boldsymbol{x}_{ai}$

Relative velocity towards ith agent: $\boldsymbol{v}_{ri}$

$$f(\boldsymbol{x}_a, \boldsymbol{x}_{ai}, \boldsymbol{v}_{ri}) = (\boldsymbol{x}_a - \boldsymbol{x}_{ai})^T P_{ai} T_{ai} P_{ai}^T (\boldsymbol{x}_a - \boldsymbol{x}_{ai})$$

where

$$P_{ai} = \begin{pmatrix} \frac{v_{rix}}{\|\mathbf{v}_{ri}\|} & & \\ \frac{v_{riy}}{\|\mathbf{v}_{ri}\|} & \vec{o}_1 & \vec{o}_2 \\ \frac{v_{riz}}{\|\mathbf{v}_{ri}\|} & & \end{pmatrix},$$

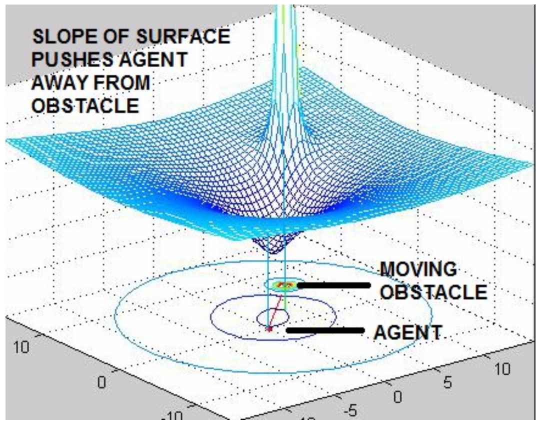
$$D_{ai} = \begin{pmatrix} \|\mathbf{v}_{ri}\| & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

where $\overrightarrow{O}_1, \overrightarrow{O_2} \in \Upsilon^3$ are picked to make the columns of the matrix $P_a i$ orthonormal. This function f is similar to a squared Euclidean norm, but is modified to incorporate relative velocity. This term pushes the agent away from nearby agents, thus the potential function is inversely proportional to the value of $f(\boldsymbol{x}_a, \boldsymbol{x}_{ai}, \boldsymbol{v}_{ri})$. This function serves to capture position and velocity information about nearby UAS agents and create individual potential fields based on the relative velocity between the agent and each of its collaborators within a given communication radius.

In the second term of the potential function,

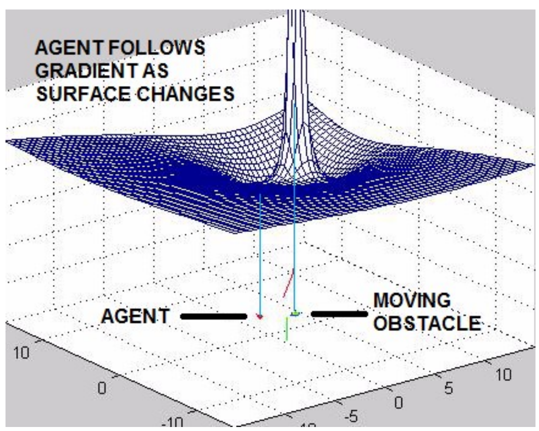
$$g(\boldsymbol{x}_a, \boldsymbol{x}_m, \boldsymbol{v}_m)^{-1} = \left((\boldsymbol{x}_a - \boldsymbol{x}_m)^T P_m D_m P_m^T (\boldsymbol{x}_a - \boldsymbol{x}_m)\right)^{-1}$$

a similar function, g is used to shape virtual potential fields around the trajectories of piloted aircraft. P_m and D_m are constructed similar to P_{ai} and D_{ai} but with $\boldsymbol{v}_{ri}$ changing to relative velocity $\boldsymbol{v}_m$ of the piloted aircraft and the host agent. This function, g is designed to create the steepest



,

Figure 2.6: Visualization of Dynamic Surface



,

Figure 2.7: Visualization of Dynamic Surface

gradient of all the terms in the potential control algorithm. As a result, the unmanned system will always have the strongest of its virtual potential fields pushing UAS agents out of the way of any piloted aircraft. As described in the literature review, the positions and velocities of incoming piloted aircraft might be determined via GBSAA or by an onboard system that interfaces with ADS-B. The final term of the potential function,

$$\|\boldsymbol{x}_a - \boldsymbol{x}_{\gamma a}\|_2^2$$

serves to attract each agent to the trajectory it has been assigned by the high-level control algorithm. The agent is attracted to a moving target point $\boldsymbol{x}_{\gamma a}$ forward of it along the path γ_a, which presents itself as a moving minima that will be chased by the UAS agent as it moves around. An algorithm was created to cause this target to move along the path and, when necessary, wait for the UAS agent. This algorithm takes as its input the path from the high-level heuristic algorithm. The target moves along the path while the UAS is within a given range. If the UAS falls too far behind the target, the target stops moving until the UAS agent is once again within range. Thus if the agent is slowed down for any reason (perhaps waiting for a piloted aircraft to pass, as dictated by previous terms in this function) then the target point will stop and wait, only to resume moving on the programmed path when the agent is within a given distance from the target point.

Summing these potential terms creates the theoretical dynamic surface, visualized in Figures 2.6 and 2.7. Descending the gradient of this sum gives the controller a tendency to roll downhill, steering agents of the UAS towards their moving target and around collisions.

2.6 Algorithm to Account for Wind During Large Area Scanning With UAS

In an effort to provide a more automated system for future scans of the same area, we revisit our Heuristic Patrol Algorithm to develop a new algorithm for scanning large areas with small UAS. When remotely operating UAS it is usually left to a human operator to decide how much battery the aircraft will need to return safely to the launch point, also referred to as the home point. For simple flights 25-30% battery capacity is often used as a rule of thumb to trigger the return home function. However, the energy required for the flight home varies depending on the component of wind velocity that aligns with the aircrafts course towards the home point. For example, the aircraft needs more energy to return home if its homeward path leads it into a strong wind than it does if the aircraft has tailwind on its way home. During downwind operations beyond the line of sight, the flight home can use as much as half the battery. Furthermore, inefficiencies resulting from the rule of thumb method accumulate when scanning large areas that require multiple flights per day for multiple days. Thus a new method is needed that takes into account variation in the amount of energy needed to return home.

It is not common practice for small portable UAS to be used for scans in excess of a thousand acres due to their limited flight duration, a function of battery life. Thus a challenge is presented to estimate the schedule and eventually automate the scans of large areas using multiple small UAS or many flights with one small UAS. When remotely operating UAS it is usually left to a human operator to decide how much battery the aircraft will need to return safely to the launch point, also referred to as the home point. For simple flights, 25-30% battery capacity is often used as a rule of thumb to return. However inefficiencies resulting from the rule of thumb method accumulate when scanning large areas that require multiple flights per day for multiple days. A new method is needed that takes into account variation in the amount of energy needed to return home. The energy needed varies depending on the component of wind velocity that lies along the aircrafts course towards the home point. For example, the aircraft needs more energy to return home if its homeward path leads it into a strong wind than it does if the aircraft has tailwind on its way home.

Discussions with industry experts at the time of writing were discouraging, advising that the scan area was too large to be scanned by anything but very large, expensive UAS. Undeterred, we altered our previously discussed heuristic patrol algorithm to take area priority and battery life into account for scanning large areas with small electric UAS. This resulted in a more detailed algorithm specifically for this type of the sensor coverage problem.

2.6.1 Algorithm Description

For the purpose of this algorithm, we assume the UAS platforms flight envelope allows the absolute speed (speed over the ground) to be slow enough such that the endlap between sensor captures from the height above the ground at the sensors fastest capture rate will be sufficient for post processing. This is an important constraint that must be checked prior to UAS, sensor and processing tool selection. If the post flight data processing tool has not been selected at the time of flight planning, it has been our experience that 75% of endlap is sufficient for nearly all processing tools.

Line 1 get sensor range that defines distance between scan corridors

The sensor range referred to in our earlier work is determined by the cameras focal length, sensor size s_p, and altitude above the ground H. The

```
1       get sensor range that defines distance between scan corridors
2       get initial boundary points for scan area
3       get home location
4   while {
5       get positions of clouds, TFRs and other obstacles
6       get current wind velocity
7       set new area clear of obstacles
8       while {
9           do decompose new area into corridors based on wind velocity
10          while {
11              do decompose corridors into a waypoint path each i UAS
12              #(or the ith flight of a single UAS)
13              if obstacle positions change {
14                  set new area clear of obstacles
15              } end
16              if time elapsed {
17                  update scan area boundary to remove parts already covered
18              } end
19          } end
20      } end
21  } end
```
,

FIGURE 2.8: Heuristic Algorithm to Cover Large Areas with Multiple UAS or Many Flights with One UAS

distance between scan corridors is reduced by the need for the processing software to have a minimum amount of sensor overlap o_side as required by the post processing image stitching algorithm [37].

$$g_{side} = f(o_{side}, s_p, H)$$

Thus the value g_{side} represents a measure of distance along the ground, perpendicular to the scan corridors.

Line 2 get initial boundary points for scan area

This area is to be selected by the user. Most UAS ground stations include GIS functions that allow an area to be defined by using sets of GPS coordinates to draw a polygon. The Mapping Toolbox for Matlab also has provisions to import and export sets of GPS coordinates in the form of kml or gpx files.

$$\boldsymbol{S}_0$$

To prevent unnecessary complexity beyond what is warranted by the requirements of this use case, the scan area $\boldsymbol{S}_0$ is represented on a fixed, two-dimensional plane tangent to the earths surface at altitude H above the ground at the home point, as is the convention for aerial vegetation scanning by UAS. The vegetation scanned by this process will be the area $\boldsymbol{S}_0$ projected onto the earths surface.

Line 4 get home location

This is the takeoff and landing point of the UAS, in many cases also the location of the ground control station. This location can be obtained by reading the GPS of the UAS prior to takeoff.

$$\boldsymbol{x}_{ih}$$

Line 5 get positions of clouds, TFRs and other obstacles

Obstacle locations are input in the same manner as the initial boundary points. Determining the actual locations of obstacles will be addressed in a subsequent chapter on implementation and testing.

Line 6 get current wind velocity

Well-established vegetation scanning practices dictate that each scan corridor for our fixed-wing hybrid VTOL platform should be perpendicular to the wind. Thus wind direction must be taken into account when planning the orientation of UAS scan corridors. The algorithm will also use wind velocity later to estimate flight paths that ensure the UAS has enough battery remaining to return from its current position.

$$\boldsymbol{V}_{W0}$$

The vector $\boldsymbol{V}_{W0}$ represents the wind magnitude and direction projected onto the two-dimensional scanning plane.

Line 7 set new area clear of obstacles

This step updates the scan area to remove the obstacles that were just defined or updated to create a new scan area.

$$\boldsymbol{S}_1 = \boldsymbol{S}_0 \backslash \boldsymbol{O}_0$$

Where $\boldsymbol{O}_0$ is the known set of obstacles including a margin of safety represented by polygons on the same plane as the set $\boldsymbol{S}_0$.

Line 9 do decompose new area into corridors based on wind velocity

This step relies on the heuristic described earlier to decompose $\boldsymbol{S}_1$ into parallel corridors of width g_side. As previously described, our requirements and constraints dictate the use of a UAS platform to be a fixed-wing or fixed-wing VTOL hybrid UAS, which requires the orientation of decomposed columns to be perpendicular to the wind direction. When creating scan paths for multirotor UAS, the orientation of scan corridors may be altered by the user or approached as a bicriteria path problem. This decomposition is analogous to the lawn mowing problem which is NP-hard. In this case the problem can be approached more simply by estimating a tour solution for the traveling salesman problem (TSP) on a simple grid graph [38]. However when any wind is present, the ground speed of the UAS will change when traversing the short links between corridors, as it will be either flying directly into the wind or directly downwind in this use case. We define the set of scan corridors using a function:

$$\boldsymbol{C}_0 = f(\boldsymbol{S}_1, \boldsymbol{V}_{W0}, g_{side})$$

Where $\boldsymbol{C}_0$ is the set of ordered scan corridors (defined by waypoints at the end of each column) that are g_side distance apart, on the same plane as $\boldsymbol{S}_1$.

Line 11 do decompose corridors into a waypoint path for each i UAS (or the ith flight of a single UAS)

This step is critical for performing accurate schedule estimates of UAS scans of large areas, especially where wind is a factor. Simply put, we must plan for each aircraft to have enough energy to return and land, even if the aircraft must fight against a strong headwind in order to make it safely home. To this end, we first create a function to constrain the UAS flight based on its fuel capacity or battery life:

$$E_{Ri} - E_{xh^i} > 0$$

Where

$$E_{Ri}(t) = E_{0i} - E_{Ui}(t) - E_d$$

And

$$E_{\gamma ih}(t) = f(\boldsymbol{x}_{ih})$$

In which E_{0i} is defined as the initial amount of fuel or battery energy contained aboard aircraft i (or the ith flight of a single aircraft), $E_{Ui}(t)$ is defined as the energy used by aircraft i at time t (or the ith flight of a single aircraft at time t), and E_d is an adjustable term to for energy safety factor to provide enough reserve fuel to ensure the aircraft returns safely home. Systems identification should be performed to ensure this safety factor term accounts for any increased energy needed for the landing phase of fixed-wing VTOL hybrid aircraft. Constraints related to battery chemistry should also be considered to ensure enough energy is reserved to prevent battery damage.

With

$$\boldsymbol{x}_{ih}(t) = f(\boldsymbol{x}_i, \boldsymbol{V}_{W0})$$

Here $\boldsymbol{x}_i h$ is the path (a set of waypoints) initiated by the flight control computer of aircraft i (or the ith flight of a single aircraft at time t) at time t to return safely home in the given wind conditions. In this use case we expect only large, simple obstacles and due to Constraint 1 flights will be postponed if many moving obstacles are present. For use cases with more obstacles or complex moving obstacles, a variety of advanced methods exist to ensure the path home avoids known obstacles, and those can be added to ground station control software in case of complex arrangements of obstacles [39]. The path of aircraft i (or the ith flight of a single aircraft) can then be estimated by combining $\boldsymbol{x}_i h(t)$ and the ith flight path along the centerline and end-links of scan corridors within $\boldsymbol{C}_0$ for the discreet time values that $E_{Ri} - E_{xh^i} > 0$ remains true.

Line 14 set new area clear of obstacles

$$\boldsymbol{S}_2 = \boldsymbol{S}_0 \backslash \boldsymbol{O}_1$$

Where $\boldsymbol{O}_1$ is the latest set of areas containing obstacles, defined by polygons. Line 17 update new area to remove parts already covered

$$\boldsymbol{S}_3 = \boldsymbol{S}_2 \backslash \boldsymbol{S}_C$$

Where $\boldsymbol{S}_C(t)$ is the scan area that has been adequately covered at time t.

This algorithm can be used to provide highly accurate estimates of scanning schedules and eventually allow total automation of the scanning process. It is designed for ease of implementation with standard UAS ground stations and flight controllers, and can also be used as part of future models to find the number of flights or the number of aircraft required to cover a large area. Most importantly, the algorithm takes wind into account, as higher scanning altitudes are nearly always subject to wind conditions that strongly influence small UAS.

Chapter 3

Models

Scanning large areas with multiple vehicles or many flights of the same vehicle often require extensive planning to deliver personnel, ground equipment and vehicles within range of the target area. For our project scanning the jungle in Belize project the travel and transportation cost was a large portion of project cost. Flight operations in Belize had to be scheduled months in advance, due to the challenges of shipping UAS batteries and many other logistical issues. Essential personnel had schedule constraints that also added complexity.

Reliable schedule and cost estimates were required long before platform and sensor selection were finalized. We created models of various UAS and sensors to estimate schedule and cost.

We develop a high-level model for the large area scanning use case that has a wide variety of uses.

3.0.1 Sensor Coverage Cost and Schedule Model

We create a cost model to determine the number of flights or the number of aircraft to cover a large area of interest using many flights of one UAS or many UAS at once 3.1. This model requires the following inputs and outputs:

3.0.2 Inputs

1. Desired Data Quality - Maximum Size on Ground (cm)
2. Sensor Image width (number of pixels)
3. Sensor Image height (pixels)
4. Sensor FOV width (degrees)
5. Sensor FOV height (degrees)
6. UAS Maximum Flight Duration (minutes)
7. UAS Cruise Speed (mph)
8. UAS Maximum Range (miles)
9. UAS Rate of Climb (feet per second)
10. UAS Takeoff Sequence Duration (seconds)
11. Duration of UAS Transition to Forward Flight if UAS is VTOL
12. Fixed-Wing Type (seconds)
13. UAS Descent Rate (seconds)

14. UAS Landing Sequence Duration (seconds)
15. Duration of UAS Transition to Hover Mode if UAS is VTOL Fixed-Wing Type (seconds)
16. Forward Overlap Required by Post Processing Algorithm (
17. Side Overlap Required by Post Processing Algorithm (
18. Maximum Sun Angle Required for Accurate Reflectance Capture (degrees)
19. Area to cover (acres)

3.0.2.1 Outputs

1. Required Flight Altitude (feet above average ground level)
2. Total Flight Duration (hours)
3. Number of Good-Weather Flight Days Needed to Cover Area

Our preliminary Belize jungle scanning model takes into account that post processing software often requires adjacent scan corridors to overlap by as much as 75%, thus dramatically reducing the effective lateral range of the sensor. Nevertheless, the model showed we could cover up to 600 acres per flight with some platform and sensor combinations while still maintaining 75% overlap. This coverage rate would require special permission from the authorities to fly at an altitude of 1500 feet. While no requirement had been given by the stakeholders for ground sample distance or pixel size (measures of data quality that are poorer at higher altitudes), our simple model showed

Edit Green Blocks and Read Yellow Ones		
Altitude (feet)	1500	
Altitude (m)	457	
Camera Image width (pixels)	1280	
Camera Image height (pixels)	960	
Camera FOV width (deg)	47.2	
Camera FOV height (deg)	35.4	
Flight plan grid width (m)	99.87	45662
Pixel size on the ground (cm)	30.8	15 cm min
Max flight time (minutes)	40	
Max speed (mph)	30	
Max range (miles)	7	
Frontlap	75%	
Sidelap	75%	
Area to cover (acres)	20000	
Number of flights	56	
Number of flight hours	37	
Calculations are below		
Pixel width (cm)	31.2	
Pixel height (cm)	30.4	
Half scan width (deg)	23.6	
half scan width (rad)	0.4118977	
Scan width (m)	399	derived from FOV angle
Half scan width (m)	200	derived from FOV angle
Multiply by 1-overlap to get grid width	99.87	width*25%

Micasense Rededge Camera		
focal length: 5.5 mm		
Imager resolution: 1280 x 960 pixels		
Imager size: 4.8 mm x 3.6 mm		
FOV 47.2 degrees horizontal, 35.4 degrees Vertical		
Recommended overlap: 70% to 75% (both directions)		

Frontlap	0.75	
Altitude (m)	457.199985	
half scan height (deg)	17.7	
half scan height (rad)	0.30892328	
Half scan height (m)	146	
Scan height (m)	292	
Multiply by dist. to get snap dist	72.96	
Flight speed (m/s)	13	15
m/s to mph	2.23694	2.23694
flight speed (mph)	30	33.5541
Flight area (m^2/s)	1339	c9* speed
Flight area (km^2/s)	0.001339	

Flight duration (s)	2400		Side length (square mission)	Passes (square mission)	side length (square mission) miles	
theoretical mission area (m2)		3214591	1793	18	1.114076629	
Remote fly home distance (miles)		4				
Remote fly home time (at max speed) (s)		960				
Normal fly home time (s)		120	tested at 1km away, 457m altitude to nearby loiter at			332
Takeoff (s)		40				
Climb out (s)		150	tested to 457m	2.8	0.208781067	ascent ra
Descent (s)		150	tested from 457m			
Land (s)		210	tested from 457m			
Remote travel %		40%				
Normal travel %		5%				
Traversing grid (s)		134	* for a square mission of 2400 seconds			
% of time crossing grid		5.6%				
Remote mission duration (s)		1076	18	minutes		
Normal mission duration (s)		1870	31	minutes		
Remote mission area (km2)	1.44					
Normal mission area (km2)	2.50					
Remote acres per mission	356	35.29123225	12574.07231		12.25	7840
Normal acres per mission	619	12	7425.927687		0.166666667	18.6667
		24	flights to cover GC from CS			
	32.3191943	16	flights to cover GC from GC			

,

FIGURE 3.1: Our model for sensor coverage and flight times of various UAS and Sensor configurations, built around a simplistic geometric sensor model by Pix4d

that even at 1500 feet, the ground sample distance of our data would be an order of magnitude better than equivalent multispectral data captured by satellite. We scheduled initial flight tests to gather sample data over the area from this altitude and the sample quality would prove to be satisfactory to the anthropologist, the principal stakeholder in this project.

3.0.3 Test Flights and UAS Model Validation

We traveled to Belize to perform flights that validate the aforementioned model we had created for the large area scanning use case. During these tests, we scanned 1000 acres of jungle using the Micasense RedEdge camera attached to a 3DR Solo COTS UAS. Data logs of the aircraft position during the model validations flights can be seen in Figure 3.2.

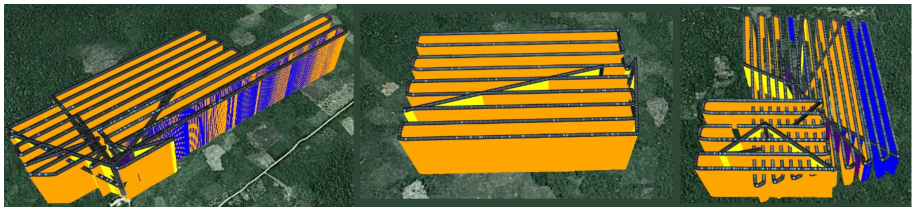
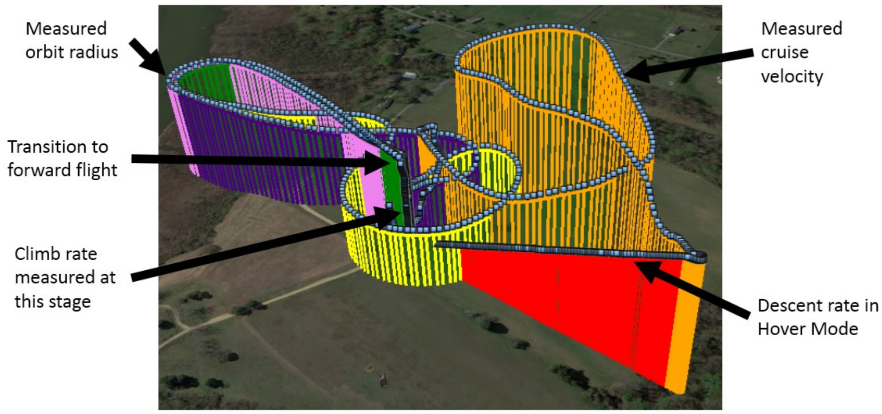
,


FIGURE 3.2: Flight paths during model validation and sensor test flights over 1000 acres of jungle, captured by the author

These initial flights also allowed us to develop more operational knowledge of the sensor and the complex logistics required to operate in a remote area, some parts of which was only accessible on foot. We also validated the model by comparing the area covered in the number of scheduled flight days with the models predictions. With the model updated we then went about selecting a platform. As previously mentioned, some performance metrics are sensitive to the flight envelope and cost of the platform. Our process for platform selection and modification is duscussed in Chapter 4.

Upon receipt of the UAS platform, we performed a series of test flights. We first tested the Return Home function and other safety related failsafe functions of the UAS. We then conducted basic systems identification to validate the flight characteristics and UAS responses such as flight envelope and rate of climb in order to update the aforementioned model. Data recorded during these flights is shown in Figure 3.3.

With our confidence in the model confirmed, we set about creating a weather model for the flight area.

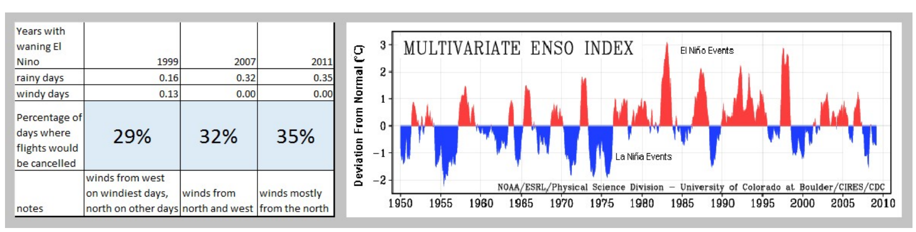


,

Figure 3.3: Recorded flight path of sysid flight to update model parameters for this UAS platform

3.0.4 Weather Model for Southern Belize

A lack of weather stations in the isolated region of our flights made it difficult to know what weather to expect. We consulted with meteorology experts and created a simple weather model with their guidance. The model is shown in Figure 3.4 alongside a graph of the El Nio cycle.



Years with waning El Nino	1999	2007	2011
rainy days	0.16	0.32	0.35
windy days	0.13	0.00	0.00
Percentage of days where flights would be cancelled	29%	32%	35%
notes	winds from west on windiest days, north on other days	winds from north and west	winds mostly from the north

,

Figure 3.4: We developed this weather model to estimate the percentage of flight days cancelled for weather. El Nio data courtesy of NOAA.

Combining the two models allowed us to inform stakeholder decisions using a tradeoff analysis, which we discuss in detail in Chapter 4. We ultimately determined a need for 6 good-weather flight days over the scan area in Belize. At this point we began to reconfigure the UAS platform for our large area scanning use case. Taking the waning El Nio pattern into account, our model accurately predicted that conditions would be too windy or too rainy for our flights 1 out of every 3 days of our operation. Our weather model proved useful, with 4 of the 10 days having periods of rain during which scanning flights were not possible.

Another notable example of savings attributable to these models was a request from Global Medic to assist in creating disaster relief maps in the island country of Dominica after it was devastated by hurricane Maria. Due to a grouping of requirements, we successfully used our models to create cost and schedule estimates with various sensors and aircraft to help the humanitarian stakeholders plan their effort to map the damage for the local government and providers of international aid. Ryan Henderson, a pilot from the University of Maryland UAS Test Site, and two industry personnel successfully mapped 5900 acres of the island [40] thanks to the use of our models.

Chapter 4

Case Study Using our MBSE Process

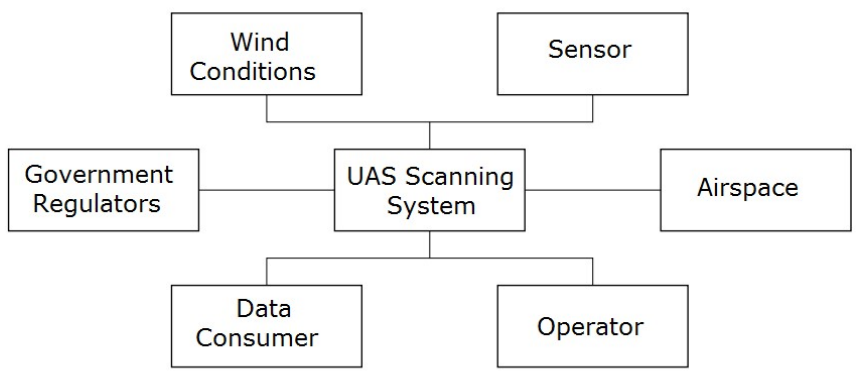
In this chapter, we discuss how we used the models mentioned in Chapter 3 to characterize the relationship between performance metrics and design parameters for this project. We present the primary stakeholder with a tradeoff analysis which is used to select the final parameters. Then we build and operate the system, determining if our parameters were selected correctly.

4.1 MBSE Approach to a Mapping Problem in Belize

We partnered with University of Maryland Anthropologist Sean Downey on an NSF project to scan large areas of jungle in Belize. We use this opportunity to test the theories and algorithms developed in the earlier chapter for the sensor coverage problem. The goal of this real-world sensor coverage problem is to provide anthropologists with high resolution multispectral images of vegetation surrounding remote villages in southern Belize. In addition to the anthropologist stakeholders, other stakeholders include Qeqchi Maya residents of the villages being scanned, pilots of passenger aircraft traversing the area, regulators (in this case the Belize Department of Civil Aviation), the Belizean military (due to tensions along a border near the flight area) and operators of the system.

4.1.1 Operational Context

The following paragraphs offer a detailed analysis of the behavior of a large area scanning UAS in an operational context.



,

Figure 4.1: Operational Context of a UAS Scanning System

Based on the operational context described in Figure 4.1, we discuss unique issues surrounding this use case and how they affect requirements and constraints.

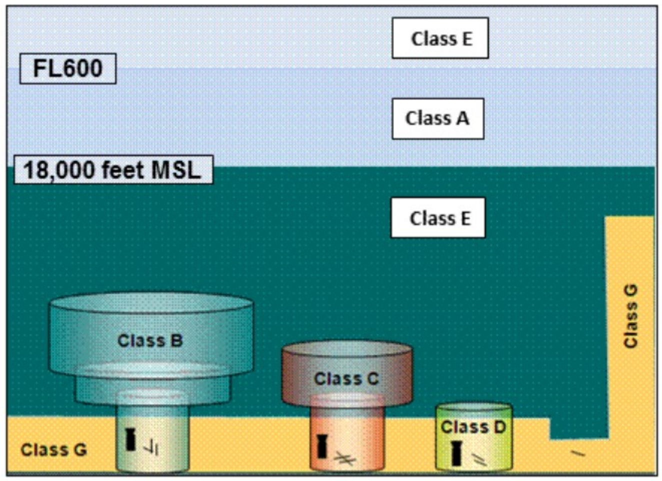
4.1.2 Motivation

Our early experiments with open source flight control software and Arduino-based embedded computers made clear that highly-effective low-level flight control algorithms have been universally available since at least the beginning of the Ardupilot open source flight controller project in 2009. Much like mobile phones and computers, COTS flight controllers and complete aircraft can now be purchased and easily modified for a small fraction of the time and cost required to develop an aircraft from scratch. For the price of a quality laptop, an aircraft can be purchased with built-in camera stabilization, waypoint navigation, vision based navigation, collision avoidance and automatic failsafe functions to guard against the most common risks associated with UAS. Thus, our work includes use case development, requirements analysis, hardware selection and modification, and high-level algorithms. We also found a lack of understanding of regulatory and stakeholder viewpoints, thus we also provide context essential to meeting stakeholder needs in this chapter.

4.2 Regulatory Stakeholder Perspective

Prior to flight testing testing for any outdoor use case, regulatory constraints must be carefully examined. We began with flight tests inside the United States national airspace (NAS).

The US national airspace is extremely complex, divided into 6 subclasses. The system is managed by approximately 14,500 air traffic controllers spread across 600 facilities (Figure 4.2).



,

FIGURE 4.2: Airspace Classes

In order to operate safely within this airspace, UAS must either be segregated to their own airspace (typically below 400 feet), or comply with the same operating requirements for manned aircraft to communicate and avoid collisions.

4.2.1 Risks Common to All UAS Use Cases in the NAS

All of the use cases discussed in this thesis involve outdoor operation of UAS 1.5 kg or larger. Flight durations of these aircraft range from 10 minutes to over 2 hours, thus safety of personnel on the ground and any manned aircraft operating nearby must be considered above all else.

The risks and costs associated with large aircraft colliding with small objects in flight is well understood within the military and commercial aviation industry sectors. But newcomers from the fields of robotics and small UAV development may not have the necessary structural knowledge of large aircraft and engines to understand the risk of damage incurred by collision by UAS as small as 250 grams. A lack of knowledge is made evident by outdoor operating procedures followed by researchers and hobbyists across the globe, where small aircraft are unfortunately operated without adherence to safety best practices or government regulations. Conversations with researchers still occasionally lead to a researcher asking the question How can something so small cause damage to a big commercial jet? While only a limited amount of military data exists regarding aircraft collisions with UAS, collision with birds are well documented enough to provide an analogous area of research. "From 1960-1988, there were 104 human fatalities and 34 serious injuries resulted from 19 bird strikes with non-military planes." While one fatality is too many, the total number of fatalities over 28 years is not remarkably high. The cost, on the other hand, is huge. Air Force planes alone averaged 3200 bird strikes per year, with a cost 45$millioneachyearfrom 1987-1993, averagingtoaround$14,000 per collision (Conover, 1995).

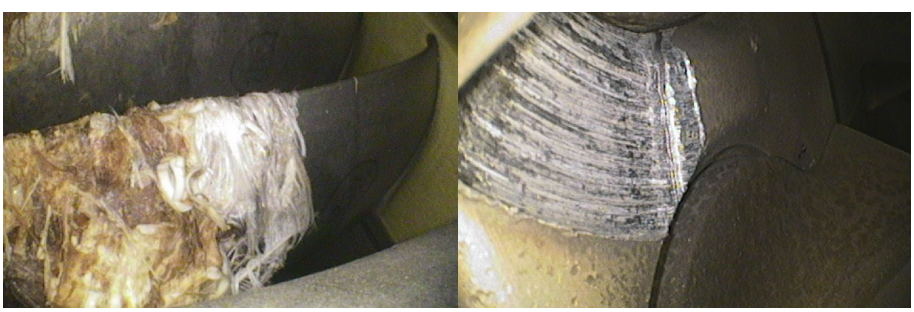
,


FIGURE 4.3: Birdstrike damage to the Second Stage Stators in the Core of a CFM56 Passenger Jet Engine, Taken by the Author

Most UAS also contain hard metal parts and lithium batteries, in contrast with birds which are relatively soft, with hollow bones. A conservative estimate for the mass of an average bird is 0.5 kg, while the regulatory size bracket for small UAVs is below 55 pounds, or 25 kg. So it is possible that a small UAV would cause as much or more damage to an aircraft as a bird in the event of a collision. Figure 4.3 shows damage to a passenger jet engine caused by a bird. For larger UAVs the potential for aircraft damage is even greater in the event of a collision. If this engine had ingested a UAS instead of a bird, the damage may have been much more severe and may have resulted in an in-flight engine failure. Recent studies have shown that UAS collisions pose a greater risk than birds [**?**]. Thus, UAS designers and operators must be aware of the risks to life and limb as these systems and concepts of operation are developed.

The risk of UAS flying over persons on the ground is also often underestimated. Operators grow complacent during long periods of operation without incident. However most multirotor UAS in operation today use designs exhibiting single points of failure. In spite of extensive design review and testing, many low-cost UAS are not built to the same standard as manned aircraft and must be operated with great care to prevent injury to persons on the ground.

Figure 4.4 shows a 50 lb UAS in the seconds after a single component within the power system failed during flight. Clearly even at a low altitude this type of failure could present a serious hazard to personnel on the ground below where the UAS is operating.

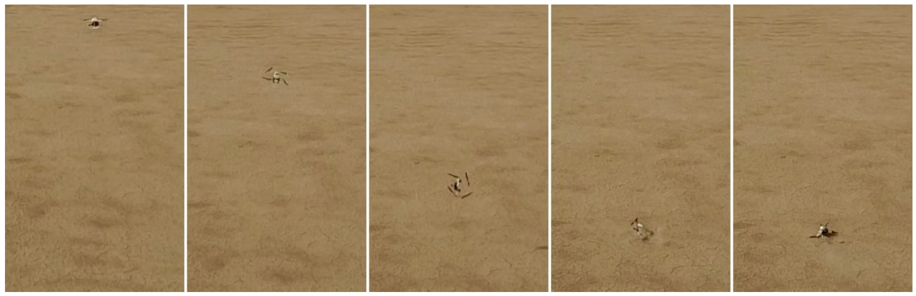
, 

FIGURE 4.4: Hazard caused by failure of a UAS power component, taken by the author from a chase UAS

4.2.2 Public Trust in Automation as Context for UAS Automation

Where academic experiments require real world testing, regulations become an obvious constraint. Aviation regulations often stem from skepticism towards autonomy and autonomous functions. This is exemplified by specific prohibition of autonomy in the latest FAA UAS regulations at the time of writing. However regulators ultimately serve the public, and public trust in human operators is beginning to give way to trust in automated systems.

Historically in aviation, regulations have allowed human operators to make the decision to override systems or protocols based on judgement and experience. Decisions are analyzed after the fact, through government and industry investigations, but humans are rarely penalized unless a clearly established rule was broken. It is far more common for personnel be penalized for turning an automated system ON that had been deactivated because of performance problems, or even resetting a protection for a system that may have been tripped by overcurrent. Thus human aircraft operators are incentivized to rely on experts who have a detailed understanding of automated systems.

This "pilot is always right" mentality began because aviation laws were modeled after age-old nautical rules in which the captain's word is law. Obviously one problem with this approach is how heavily it depends on the soundness of human judgement. There was a time when large aircraft required a crew of 8, thus any erratic behavior by a crew member could be thwarted by the others. Now even large aircraft only require only two human operators, and can be operated by only one. Capabilities for automation have made human operators redundant in many situations, the prime example being ICAO Category IIIb Instrument Landing Systems which currently allow many large aircraft to land automatically in weather conditions, where humans otherwise could not see well enough to land.

Waning public trust in human operators is underscored by public response to the tragedy of Germanwings Flight 9525, where the public has called for capabilities to override aircraft systems from the ground, or implementation of flight control algorithms in such a way that existing Ground Proximity Warning Systems (GPWS) can override inputs by human pilots. Malaysia Airlines Flight 370 also resulted in similar calls for capabilities to track and take control of aircraft from the primary human operator. With regulators operating at the whim of the public, loss of public trust in human operators will lead to calls for more and more robust automation of manned and unmanned aircraft systems, likely removing human operators from the control loop in the years to come. The implications of removing human operators from the loop go against a century-old philosophy on which every aircraft system has been designed to date. Changes in public and regulatory trust in automated systems over humans require a serious rethinking of the command and control philosophy of both manned aircraft and UAS, on which developers and regulators must find common ground.

Even now, in the early stages of automated aircraft systems, increased automation has presented new problems. The crash of Air France flight 447 was a wake-up call for human factors engineers to carefully examine what is referred to as the out-of-the-loop performance problem. Simply put, when a highly automated control system is degraded due to partial failure, the operators have a hard time operating it in manual mode because they do not have enough practice operating the system manually. The need for further study of this problem has been underscored by a litany of incidents including Asiana Airlines Flight 214, and Turkish Airlines Flight 1951. These imminent changes in regulatory philosophy must be taken into account by UAS developers as we design and test systems to ensure safety of flight along with future regulatory compatibility.

4.2.3 Developing safety requirements

Rather than risk overlooking critical safety of flight issues by inventing safety requirements from scratch, for this project we built upon decades of knowledge obtained by hobbyists and US military personnel. To develop a comprehensive set of safety requirements, requirements were derived from hobbyist best practices and military airworthiness standards for UAS.

We began with a 2014 analysis of hobbyist safety protocols. We derived our initial safety requirements directly from the safety rules of the DC Area

Drone User Group, a group of local hobbyists with many decades of combined UAS experience. These rules were initially created by the group to govern entries to their Search and Rescue Challenge competition. We then compared this initial set of safety requirements with mature and more detailed safety requirements used by the AUVSI Seafarer Chapter to govern entries to their Student UAS Competition, an annual event held since 2002.

4.2.4 Process for Documenting Airworthiness Approval

As part of a university-wide effort to to assure the highest standard of UAS flight safety, the University of Maryland UAS Test Site developed its own internal airworthiness review process. Based on the UAS airworthiness process developed by the US Navy, his data-centric process uses subject matter experts to scrutinize the UAS and collect technical data about the system and its components.

Once each subsystem is certified by a subject matter expert, the system is sent to the Chair of Aerospace Engineering, the Dean of the A. James Clark School of Engineering, and finally the VP of Research of the University System of Maryland to approve the system for flight. The documentation provided by this process assures system safety and helps assure regulators and other stakeholders that our systems can be operated safely as long as prescribed operating limitations are adhered to.

,

FIGURE 4.5: The author serves as a subject matter expert during an airworthiness review of a Lattitude HQ-60 fixed-wing hybrid VTOL UAS

4.2.5 Operating in Close Proximity to Large Passenger Aircraft

The University of Maryland UAS Test Site obtained a Certificate of Waiver of Authorization from the FAA to perform some test flights of our UAS in close proximity to large passenger aircraft at Atlantic City International Airport as part of the FAA UAS Pathfinder program. This was were the first time the FAA authorized UAS flights in Class C airspace within the United States. We were able to maintain safe separation while operating simultaneously with Lockheed C-5 Galaxy and Airbus 320 aircraft. Due to the extremely high cost of malfunction, and the need for immediate adherence to voice commands from human air traffic controllers, only one UAS was operated at a time and it was continuously monitored with the ability to manually take control as needed. In accordance with the approved COA and industry best practices, we kept the UAS within visual line of sight, paying special attention to control link quality and ensuring all UAS fail-safe functions were configured to avoid interfering with the flight patterns of nearby manned aircraft. This experience was paramount in building our case to regulators to allow further testing of the system in the vicinity of manned air traffic.

4.2.6 Operating Beyond Line of Sight Near Passenger Aircraft

To develop our system further and test it beyond line of sight, we partnered with University of Maryland Anthropologist, Sean Downey to perform aerial scanning flights of large areas of jungle in southern Belize for an NSF project studying land use patterns. We approached the Belize Department of Civil Aviation and made a case that our previous experience operating within the NAS gave us the necessary skills and best practices to operate our UAS safely beyond line of sight, communicating with and avoiding passenger aircraft traversing the scan area in southern Belize. Our request to fly beyond line of sight at an altitude of 1500 feet was approved by the Belizean Government on December 19, 2016.

4.2.7 Mitigating Risks Specific to the Belize Use Case

While the aforementioned algorithms discussed in previous chapters were proposed to solve the UAS sensor coverage problem are designed to meet requirements in a general use case, the Belize jungle scanning problem adds a unique set of risks resulting in a new set of project-specific requirements and constraints.

4.2.8 Constraints specific to the Belize jungle scanning project

1. Must mitigate the risk of loss of life and limb due to collision with air passenger traffic to the satisfaction of regulators and industry best practices

2. Must mitigate the risk of propeller injury during launch of aircraft in a remote area far from medical services to the satisfaction of the aircraft operator (in this case, the author)

3. Must mitigate the risk of escalating an international conflict by entering hostile airspace along the Guatemalan border to the satisfaction of Belizean military leaders

4. Schedule is constrained by travel dates and complex logistics of shipping batteries to Belize by boat.

4.2.9 Requirements specific to the Belize jungle scanning project

1. Perform a multispectral scan of more than 10,000 acres of remote jungle

2. Minimize the risk of incomplete coverage due to system malfunction or aircraft loss

3. Minimize the risk of incomplete coverage due to bad weather

4. Minimize Risk of damage or loss of expensive sensor and UAS systems

4.3 Satisfying of Requirements and Constraints Specific to the Belize Use Case

In the next paragraphs we discuss design choices and characterize their relationships with the constraints, requirements and metrics of this use case.

4.3.1 Platform Selection for Belize Use Case

The UAS platform selected to perform these scanning flights needed to cover a large area of over 10,000 acres. However traditional hand-launched fixed wing UAS can put the operator at a slightly increased risk of injury due to the operators proximity to the propeller during launch procedures. This slight increase in risk of injury is typically acceptable, however for the Belize jungle scanning project the operator will be hours or even days away from advanced medical care. Thus injury to the operator not only may put his or her life at risk, but even a minor injury requiring medical attention could reduce the area covered by the project due to the need for medical attention pushing the timeline too close to schedule constraints. Furthermore, traditional fixed-wing systems require more operator training for runway landing functions. Alternative landing functions exist such as deep stall but technology is not yet mature and may put sensor at risk of damage. We performed many test flights with traditional fixed-wing UAS and these flights showed an elevated risk of aircraft and/or sensor damage during landing. Using a hybrid fixed-wing VTOL (vertical takeoff and landing) UAS platform thus satisfies Constraint 2 and partly satisfies Requirements 2 and 4.

The remoteness of the scan area and the need to engage stakeholders within the Mayan community also influenced platform selection. Some of the stakeholders reside in a village with no roads, which is only accessible on foot or with an offshore passage by boat. No clearings in that village are large enough to operate a traditional fixed-wing UAS. Thus to operate from the remote village, a VTOL UAS also satisfies Requirement 1.

The need to satisfy Constraints 1 and 3 drove us to select an onboard flight control computer or (hereafter referred to as flight controller) with a long history of field testing and demonstrated operational success. Unfortunately the architecture of standard flight controllers meant some of the more elegant low level potential algorithms described in the solution chapter could not be implemented in this iteration of our design.

4.3.2 Functions of Common Flight Controllers

An important function of off the shelf flight controller is the return home function. The return home behavior can be automated using the flight controllers programmable battery failsafe function, in which case the operator selects a battery voltage that will trigger the Return Home function also known as the RTL function. However the user-selected voltage to trigger the battery failsafe function is a parameter of the flight controller that should not be altered during flight. The Return Home function is also typically programmed to initiate in the event of a loss of the command and control radio link. Our flight tests have shown that changing flight controller parameters during the flight increases the risk of malfunction resulting in possible aircraft damage or injury to persons on the ground. Thus the parameters for battery failsafe and loss of command and control link behaviors must remain constant while the aircraft is in flight.

Req or Constr	#	Description	Satisfied by
Constraint	1	Must mitigate the risk of loss of life and limb due to collision with air passenger traffic to the satisfaction of regulators and industry best practices	FPV, Airworthiness Review, Backup C2 Link, Flight Termination Function
Constraint	2	Must mitigate the risk of propeller injury during launch and recovery in a remote area far from medical services to the satisfaction of the aircraft operator (in this case, the author)	Hybrid VTOL Design
Constraint	3	Must mitigate the risk of escalating an international conflict by entering hostile airspace along the Guatemalan border to the satisfaction of Belizean military leaders	FPV, Airworthiness Review, Backup C2 Link, Flight Termination Function
Constraint	4	Schedule is constrained by fixed travel dates and complex logistics of shipping batteries by boat	Weather model, Hybrid VTOL Design
Requirement	1	Perform a multispectral scan of more than 10,000 acres of remote jungle	Hybrid VTOL Design
Requirement	2	Minimize the risk of incomplete coverage due to system malfunction or aircraft loss	Hybrid VTOL Design
Requirement	3	Minimize the risk of incomplete coverage due to bad weather	Weather model
Requirement	4	Minimize Risk of damage or loss of expensive sensor and UAS systems	Hybrid VTOL Design

,

FIGURE 4.6: Table of Requirements and Constraints

4.4 Trade Space

The choices we make in our effort to provide the best solution for the Belize jungle scanning problem are informed by investigating design trade-offs. We study how design choices affect performance metrics which are tied to our initial requirements.

4.4.1 Design Parameters for Large Area Scanning System

Planning large scans in remote areas requires extensive travel planning and logistics. For many electric aircraft batteries, large batteries restricted from transportation by air freight and must be shipped by land and sea. Furthermore, seasonal constraints on vegetation scanning as well as busy academic schedules dictate anthropologists and UAS operators dictate that travel logistics may need to be planned well in advance. Thus the number of days

Parameter
Number of scheduled flight days
Flight altitude (m)

,

Figure 4.7

dedicated to scanning a specified area of jungle is an important parameter in designing the UAS scanning system capture process. In addition, scanning large areas with small UAS with a limited schedule may require higher altitudes than usual resulting in a need to go through a lengthy permitting process to obtain flight clearance from regulating authorities.

We select performance metrics most important to stakeholders and develop models for cost and data quality to inform stakeholders about tradeoffs as we develop the scanning process.

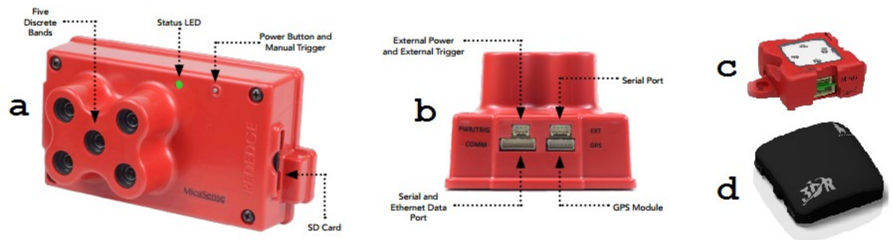
4.4.2 Best Practices for Vegetation Scanning

The current best practice for manually maximizing the scan area during a single flight is for the operator to periodically calculate energy required to return home based on prevailing wind and the aircraft location, and to manually initiate the Return Home function when the aircraft has just enough battery remaining. This ongoing calculation by the operator during each flight results in added operator workload and introduces an unnecessary risk due to human error. The recommended solution will use a flight monitor software module running on the aircraft ground station or a connected computer that sends a signal to the aircraft initiating the return home function.

4.4.3 Sensor Selection

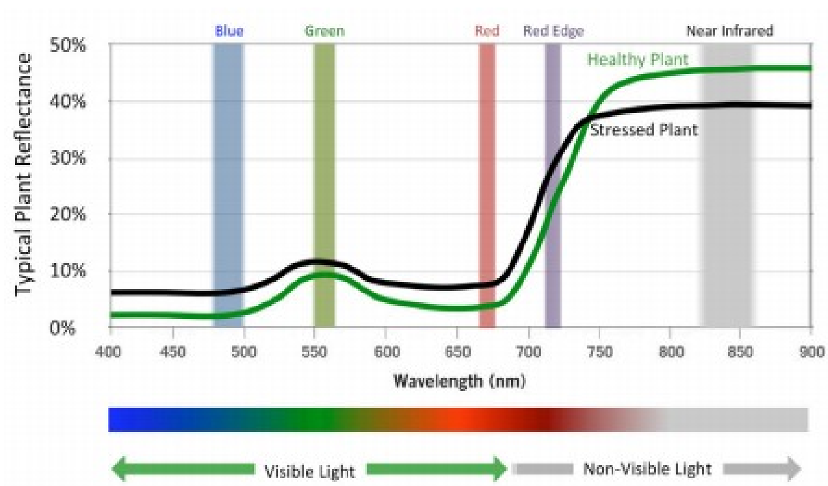
Based on tradeoffs outlined above we understood sensor cost, resolution, and field of view would influence our system performance. To select a sensor capable of collecting the required multispectral data for vegetation analysis, we attended trade shows and discussed our vegetation scanning requirements with a large number of sensor manufacturers and experts. After using the models described in Chapter 3 to analyze the effects of various sensors on overall system performance we determined the largest effect would be decreased risk of an incomplete scan associated with having a sensor that had not yet been subjected to extensive field testing. Furthermore we already owned one multispectral sensor and our familiarity with its operation further decreased the risk of an incomplete scan due to sensor failure. Thus our sensor of choice was the Micasense Rededge multispectral camera due its ease of use as a standalone sensor, its ease of integration with a variety of UAS platforms, and to a lesser extent, its higher resolution compared to some other multispectral sensors.

Multispectral sensors have long been used in agriculture and vegetation analysis. The five cameras in the Micasense Rededge correspond to five discrete wavelengths shown in Figure 4.9.



,

FIGURE 4.8: Micasense Rededge front view. B. Side view. C. Down-welling Light Sensor. D. Dedigated GPS

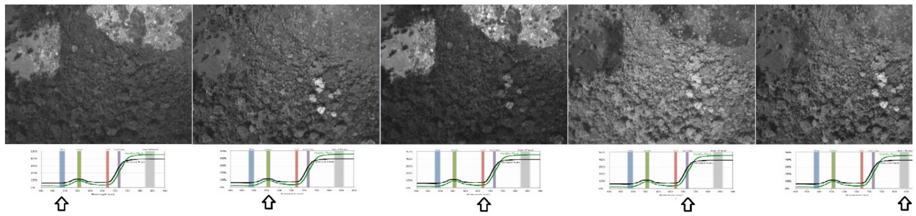


,

FIGURE 4.9: Wavelengths of the five cameras within the Micasense Rededge Multispectral sensor, image courtesy of Micasense

4.4.4 Sensor Testing

Our ongoing tests of the Micasense Rededge camera over various types of vegetation data are helping us develop new processes to determining vegetation health. During our test flights in Belize we obtained thousands of images from the sensor as well as corresponding images in the visible spectrum from a GoPro camera mounted on the aircraft. Figure 4.10 shows vegetation reflectance in a variety of wavelengths.



,

Figure 4.10: A single capture of raw sensor data with corresponding wavelengths marked below

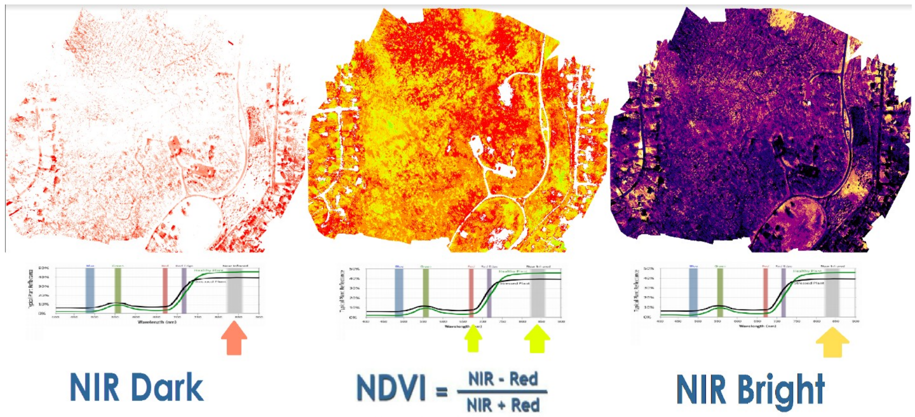
,

Figure 4.11: Corresponding data from the visible spectrum

4.4.5 Processing Sensor Data

We explored a variety of tools for stitching and processing multispectral aerial photographs and chose Pix4d Mapper Pro to stitch the images captured by our sensor and process the data.

The Pix4d Mapper software tool uses position metadata saved by the RedEdge camera to stitch the images into a single mosaic image. One measure of image quality is ground sample distance, measured in pixel size, which is a dimension of the geometric footprint on the ground covered by one pixel in the image. The camera focal length and sensor size are fixed, so pixel size becomes a function of altitude and camera tilt. The scans in



,

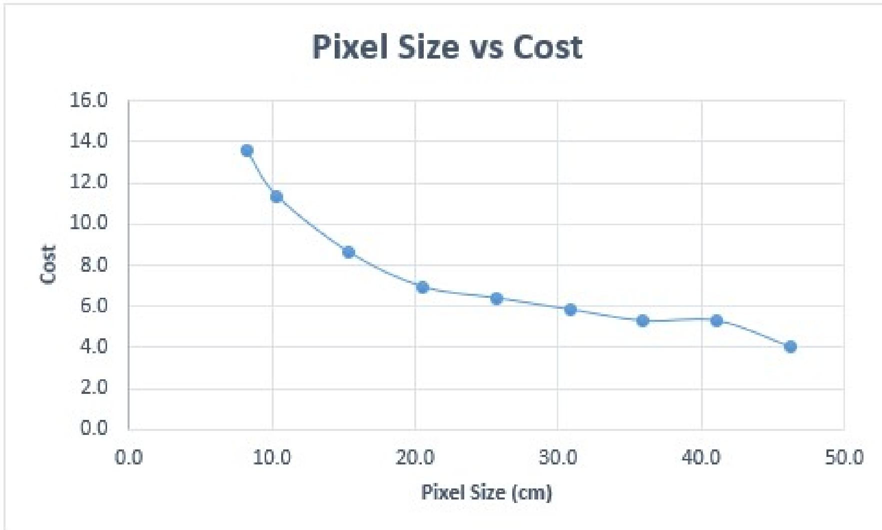
FIGURE 4.12: Post-processing Micasense Rededge data collected by the author

Figure 4.12 were taken from an altitude of 400 feet, giving a pixel size of 8cm.

4.4.6 Tradeoffs

Using the above models we estimated the system responses to a range of values for each input parameter.

Stakeholders were briefed on the tradeoff between cost and pixel size and decided that a pixel size of 30.8 cm was a sufficient improvement over existing multispectral satellite vegetation data. The corresponding input parameters to the this configuration of the process were a flight altitude of 457 meters and a flight schedule of 10 days, 4 of which our weather model predicted to be too windy or rainy to fly.



,

FIGURE 4.13: Flight paths during model validation and sensor test flights over 1000 acres of jungle, captured by the author

4.4.7 Scanning Platform

The BirdsEyeView Aerobotics Firefly6 Pro hybrid fixed-wing VTOL UAS platform was ultimately selected. A spare identical system including ground station and sensor was purchased to ensure scanning could continue in the event of technical problems or loss or damage to aircraft/sensor.

4.4.8 UAS Configuration

We worked extensively with the UAS manufacturer to design substantial modifications to the UAS in order to increase the operating range while remaining in compliance with industry best practices for beyond line of sight flights in the vicinity of manned aircraft. For its command and control link, Firefly6 Pro system originally included a 915 mhz radio manufactured

,


Figure 4.14: Firefly6 Pro Hybrid UAS in hover mode, taken by the author

by 3DR installed on the aircraft combined with an RFD900+ radio modem on the ground side. Based on input from the aircraft manufacturer, we narrowed our configuration options to the following four designs for a new command and control link architecture.

1. Disable and remove PS4 gamepad from ground control station. Install Dragonlink v3 433 MHz transmitter with a Futaba 14SG controller.

    - Pros:
        - Huge safety factor for range (DL has a 50+ km range, only 6km required)
        - Our previous experiments with other aircraft having the same Pixhawk flight controller exhibited no signs of problems
        - The Firefly6 Pro AvA ground station software still allows monitoring and mode changes through AvA graphic user interface buttons without use of the PS4 gamepad
        - If this configuration works we will log hundreds of flights which can be used to prove this setup for very long range

  - It is reversible; If the configuration does not work we will during initial tests we can change it back to the way it was with a parameter file and a clean install of the Firefly6 Pro AvA ground control station software

- Cons:
  - We would help needed setting up flight control parameters in the Futaba controller as well as RC control parameters in the Pixhawk flight controller (perhaps we might use parameters from the Firefly6 DIY configuration)
  - Filters would be needed to prevent 1.2 GHz video transmitter interfering with the DragonLink command and control link (but our operations will be well within the range limits of the DragonLink radio)
  - This is not a proven configuration for the Firefly6 Pro

2. Install a servo multiplexer as a kill switch to shut off the front 4 motors via a separate radio transmitter and receiver (such as the Dragonlink v3).

   - Pros:
     - This meets the best practices and legal requirement for redundant command and control.
   - Cons:
     - Not much safety factor in range
     - The servo multiplexer adds an additional single point of failure for the entire system

3. Replace the 915 Mhz 3DR radio with an RFD900+ on the aircraft side to increase the range of the Firefly6 Pro AvA ground station and continue using the PS4 gamepad
   - Pros:
     - This would increase the stock control range
     - This would increase the stock control range
   - Cons:
     - This may not meet the backup command and control requirement, depending on regulatory interpretation
     - We may experience flight control chatter due to the high power transmissions from the RFD900+ radio interfering with the signal cables to the control servos (chatter could perhaps be improved with filters and a BEC to power the RFD 900+)
4. Add a second 3DR (or RFD900+) radio on board the aircraft to increase the range of the Firefly6 Pro AvA ground control station with the PS4 gamepad
   - Pros:
     - Indisputably meets the requirement for backup command and control
   - Cons:
     - Two powerful radios on board could cause a brownout of the aircraft power system (which could be solved by powering radios with a BEC directly connected to the battery)

After receiving input from various subject matter experts in academia and industry, we upgraded the aircraft radio to a more powerful RFD900+, powered directly from the aircraft power distribution board to prevent flight controller brownout. We relied on the disarm function as a last resort, to terminate the aircraft in an unpopulated area in the event of an imminent collision with a manned aircraft or a flyaway within 1 km of the nearby restricted airspace. We also added a 2.5 Watt analog video transmitter manufactured by Readymade RC with the appropriate filters and shielding to prevent interference with the existing GPS receiver. The analog video link allowed us to search for traffic, monitor the weather at altitude, and navigate home by landmarks in case of GPS failure. Once on site in Belize, we harvested a 40-foot length of bamboo and erected an antenna tower using guy wires, placing the ground station radio at the top of the tower to ensure a command and control radio link beyond 4 miles without the use of directional tracking antennas. We used powered USB hubs to extend the USB connection for ground side the RFD900+ radio from the ground control station PC to the top of the tower.

The platform and all modifications were approved for flight using the comprehensive airworthiness review process developed by University of Maryland UAS Test Site Director of Operations, Tony Pucciarella and approved by the Chair of Aerospace Engineering, the Dean of the A. James Clark School of Engineering, and finally the VP of Research of the University System of Maryland. Although schedule constraints would prevent us from testing this configuration on long range flights prior to our arrival in Belize, the configuration would prove to be a resounding success in the field, exceeding the 4-mile range requirement and maintaining a link for UAS command and control with video at a range of 4.3 miles. It would become another example of the impressive benefit the University of Marylands airworthiness process can provide for a UAS project.

4.4.9 Range Testing

Due to schedule constraints, we were unable to perform flight tests beyond line of sight to fully validate our large area scanning model prior to our scheduled operations in Southern Belize. Thus our first attempts at scanning large areas were on site in Belize with the UAS scanning system and processed developed by our systems engineering approach.

4.4.10 System Operation

We set out to deploy the UAS scanning system in January of 2017. Flight and research permits were obtained from the appropriate government agencies in Belize and assistance from the University of Maryland UAS Test Site ensured that all equipment logistics were arranged.

A multi-day journey including muddy roads soaked through by the rainy season preceded our initial tests beyond line of site. Humid, tropical conditions initially wreaked havoc on some of the more delicate components, but after many hours of frantic soldering and field repairs, we had the system hardened and ready for operations in the humid environment. We deployed the UAS scanning system on January 10, 2017, the first of five annual scans of the same area. We performed 24 successful scanning flights over the 10 day period, collecting vegetation data over an area of 14,000 acres. The

,


Figure 4.15: Equipment loaded for the final leg of transportation into the jungle

system met all requirements and operated successfully within constraints, with the flight operations coming in very near the estimated cost.

4.5 Lessons Learned

Applying or at least attempting to apply the models and algorithms we created to this real-world problem was instrumental in evaluating their effectiveness. Upon reflecting on the strengths and weaknesses of our MBSE approach, one major challenge was estimating and subsequently justifying the cost of the MBSE process for this relatively small project. Most textbook project case studies were large-scale projects with teams of systems engineers and IT personnel to support and manage powerful modeling and optimization software tools. We investigated these tools to gain familiarity but found that none of our projects had a scale large enough to support them. We did however observe that our MBSE methodologies were effective even without powerful tools and provided obvious and substantial cost savings while allowing stakeholders to make key decisions.

4.6 Conclusion

Our careful attention to regulatory stakeholder perspectives allowed us to safely operate small UAS beyond line of sight in Southern Belize with other air traffic present.

Successful design and implementation of a large area mapping system was made possible by the systems engineering methodology described herein. Approaching the Belize jungle scanning problem as a specific use case of a system designed to solve the sensor coverage problem allowed us to create sufficiently detailed models to accurately determine how design parameters would affect system metrics. Tradeoffs and modeling approaches were paramount in providing stakeholders with the information needed for critical project decisions.

Going beyond the focus on simple, textbook problems, we applied Model Based Systems Engineering to a complex and literally dirty problem of large area mapping with resounding success. We showed that the model based systems engineering approaches developed by the University of Maryland Institute for Systems Research provides a robust framework for reducing complexity and assuring stakeholder engagement, resulting in processes that can provide improvements across sectors with many use cases involving UAS.

Continuing in our work, we will fully implement the large area mapping algorithm to take the wind into account with the goal of autonomously mapping a large area in Belize over many flights. Our goal is to improve data collection and reduce UAS operation cost over the next three annual scans of Southern Belize for our project with Dr. Sean Downey.

Bibliography

[1] Drowning fact sheet, Jan 2018. URL http://www.who.int/mediacentre/factsheets/fs347/en/. Accessed: 2018-03-25.

[2] Cruise passengers overboard. http://www.cruisejunkie.com/Overboard.html. (Accessed on 04/13/2018).

[3] Useful and timely delivery drone drops life preserver to australian swimmers - ieee spectrum. https://spectrum.ieee.org/automaton/robotics/drones/useful-and-timely-delivery-dr (Accessed on 04/13/2018).

[4] Eduard F van Beeck, CM Branche, David Szpilman, Jerome H Modell, and Joost JLM Bierens. A new definition of drowning: towards documentation and prevention of a global public health problem. *Bulletin of the World Health Organization*, 83:853–856, 2005.

[5] H3dynamics. https://www.h3dynamics.com/products/drone-box/. (Accessed on 04/13/2018).

[6] Beaufort wind force scale - met office.

https://www.metoffice.gov.uk/guide/weather/marine/beaufort-scale. (Accessed on 04/13/2018).

[7] Amit Bhatia, Matthew R Maly, Lydia E Kavraki, and Moshe Y Vardi. Motion planning with complex goals. *IEEE Robotics & Automation Magazine*, 18(3):55–64, 2011.

[8] Susan Hert, Sanjay Tiwari, and Vladimir Lumelsky. A terrain-covering algorithm for an auv. In *Underwater Robots*, pages 17–45. Springer, 1996.

[9] Howie Choset. Coverage for robotics – a survey of recent results. *Annals of Mathematics and Artificial Intelligence*, 31(1):113–126, Oct 2001.

[10] D. Mellinger and V. Kumar. Minimum snap trajectory generation and control for quadrotors. In *2011 IEEE International Conference on Robotics and Automation*, pages 2520–2525, 2011.

[11] S. Gupta, J. Hare, and S. Zhou. Cooperative coverage using autonomous underwater vehicles in unknown environments. In *2012 Oceans*, pages 1–5, 2012.

[12] P. R. Drake and R. J. Stamm. Gbsaa radar with altitude processing supporting uas in the nas. In *2011 Integrated Communications, Navigation, and Surveillance Conference Proceedings*, pages 1–12, 2011.

[13] *DGDL 39 Portable ADS-B and GPS Receiver Users Guide*. Springer.

[14] Lynne E. Parker. Distributed algorithms for multi-robot observation

of multiple moving targets. *Autonomous Robots*, 12(3):231–255, May 2002.

[15] FLIR. Thermal imaging cores and components, url = http://www.flir.com/cs/emea/en/view/?id=42103.

[16] Pedram Hovareshti. Consensus problems and the effects of graph topology in collaborative control. phd thesis, university of maryland, 2009. URL `http://drum.lib.umd.edu/handle/1903/9304`.

[17] Oussama Khatib. *Real-Time Obstacle Avoidance for Manipulators and Mobile Robots*, pages 396–404. Springer New York, New York, NY, 1990.

[18] Olaf Frommann. Hk beerlift 58,7kg. URL `https://www.youtube.com/watch?v=VpW`$_B 9NHwaQ$.

[19] Manuel G. Ortega Raffo, Guilherme V. and Francisco R. Rubi. An integral predictive/nonlinear h control structure for a quadrotor helicopter. In *Hybrid Systems: Computation and Control*, volume 46, pages 29–39, 2010.

[20] Adam Bry Charles Ritcher and Nicholas Roy. Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments, 2013. URL `http://www.michigancmes.org/papers/roy7.pdf`.

[21] Olaf Stursberg and Bruce H. Krogh. Efficient representation and computation of reachable sets for hybrid systems. In Oded Maler and Amir Pnueli, editors, *Hybrid Systems: Computation and Control*, pages 482–497, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

[22] Eugene Asarin, Olivier Bournez, Thao Dang, and Oded Maler. Approximate reachability analysis of piecewise-linear dynamical systems. In Nancy Lynch and Bruce H. Krogh, editors, *Hybrid Systems: Computation and Control*, pages 20–31, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.

[23] Antoine Girard, Colas Le Guernic, and Oded Maler. Efficient computation of reachable sets of linear time-invariant systems with inputs. In João P. Hespanha and Ashish Tiwari, editors, *Hybrid Systems: Computation and Control*, pages 257–271, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[24] Colas Le Guernic and Antoine Girard. Reachability analysis of hybrid systems using support functions. In Ahmed Bouajjani and Oded Maler, editors, *Computer Aided Verification*, pages 540–554, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[25] Colas Le Guernic and Antoine Girard. Reachability analysis of hybrid systems using support functions. In *Reachability Analysis of Linear Systems Using Support Functions, Nonlinear Analysis: Hybrid Systems*, volume 4, pages 250–262, 2010.

[26] M. Althoff, D. Althoff, D. Wollherr, and M. Buss. Safety verification of autonomous vehicles for coordinated evasive maneuvers. In *2010 IEEE Intelligent Vehicles Symposium*, pages 1078–1083, 2010.

[27] M. Althoff and J. M. Dolan. Set-based computation of vehicle behaviors for the online verification of autonomous vehicles. In *2011 14th*

International IEEE Conference on Intelligent Transportation Systems (ITSC), pages 1162–1167, 2011.

[28] M. Althoff and J. M. Dolan. Reachability computation of low-order models for the safety verification of high-order road vehicle models. In *2012 American Control Conference (ACC)*, pages 3559–3566, 2012.

[29] Matthias Althoff. Reachability analysis of nonlinear systems using conservative polynomialization and non-convex sets. In *Proceedings of the 16th International Conference on Hybrid Systems: Computation and Control*, HSCC '13, pages 173–182, 2013. ISBN 978-1-4503-1567-8.

[30] Thao Dang, Colas Le Guernic, and Oded Maler. Computing reachable states for nonlinear biological models. In Pierpaolo Degano and Roberto Gorrieri, editors, *Computational Methods in Systems Biology*, pages 126–141, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[31] Yuchen Zhou. *Motion planning and controls with safety and temporal constraints*. PhD thesis, University of Maryland, College Park, 2016.

[32] Y. Koren and J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. In *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pages 1398–1404 vol.2, 1991.

[33] J. S. Baras and Xiaobo Tan. Control of autonomous swarms using gibbs sampling. In *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, volume 5, pages 4752–4757 Vol.5, 2004.

[34] Xiaobo Tan Xi, Wei and John S. Baras. Gibbs sampler-based coordination of autonomous swarms. In *Automatica*, volume 42, pages 1107–1119, 2006.

[35] J. S. Baras, Xiaobo Tan, and P. Hovareshti. Decentralized control of autonomous vehicles. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, volume 2, pages 1532–1537 Vol.2, 2003.

[36] Wei Xi Tan, Xiaobo and John S. Baras. Decentralized coordination of autonomous swarms using parallel gibbs sampling. In *Automatica*, volume 46, pages 2068–2076, 2010.

[37] How to select camera focal length and flight altitude considering the desired ground sampling distance and terrain to map? support. https://support.pix4d.com/hc/en-us/articles/202558849-How-to-select-Camera-Focal- (Accessed on 04/14/2018).

[38] Esther M Arkin, Sándor P Fekete, and Joseph SB Mitchell. Approximation algorithms for lawn mowing and milling. *Computational Geometry*, 17(1-2):25–50, 2000.

[39] A Pandey, S Pandey, and DR Parhi. Mobile robot navigation and obstacle avoidance techniques: A review. *Int Rob Auto J*, 2(3):00022, 2017.

[40] Drones map devastated dominica - aopa. https://www.aopa.org/news-and-media/all-news/2018/march/19/drones-map-devastated- (Accessed on 04/13/2018).