Unmanned Aerial Vehicles for Volcanic Plume Sampling

Candace Walden
Department of Electrical and Computer Engineering
California State University
Northridge, California, United States

Mircea Teodorescu, Sebastian Hening, Jeremy Baumgartner
Department of Computer Engineering
University of California
Santa Cruz, California, United States

Small, inexpensive unmanned aerial vehicles (UAVs) constructed from off the shelf components can be used in coordinated teams to sample the airspace and provide an accurate mapping of distributed particles from recent volcanic eruptions . The UAVs are constructed from hobbyist RC planes, where the remote control is bypassed by a programmable Arduino autopilot. These UAVs are equipped with low power MEMS based sensors to detect particles and XBee communication devices running on the ZigBee protocol, which allows for mobile mesh networking. The focus was on creating programming that ensured the UAVs were always within communication range with the home base while flying a mission, either themselves or through a network of relay planes.

I. INTRODUCTION

A. Motivation

Each time a volcano erupts it ejects high quantities of pulverized rock, known as volcanic soot, which is propelled into the high altitude layers of the atmosphere by a highly toxic mixture of superheated gases [1][2]. Though the volcanic soot is composed of microscopic pieces, it is extremely hard rock and poses a danger to aeronautical jet engine turbine blades during high velocity impacts [3]. This affect means thousands of flights are grounded during and after any major volcanic activity. These dense plumes, when conditions are right, have the ability to spread throughout and remain in the atmosphere for substantial amounts of time. This spread in the upper atmosphere, the imprecision of satellite based measurements and our current lack of in-situ measurement of the volcanic plumes means we do not have an accurate model of current conditions. However the cost of most methods of accurate volcanic plume measurement is prohibitive [4][5][6].

B. Objectives

The goal of this project is to create a flexible, decentralized network of unmanned aerial vehicles (UAVs) which can work cooperatively in order to measure the distribution of the particle and gaseous components of volcanic plumes. The UAVs should be able to respond to changes in requested data gathering sites, relay the information back to the home station at all times and continue the mission if any individual unit is unable to.

Ideally the UAV team will be affordable, easy to deploy and control if needed once deployed, and reliable for volcanic plume sampling as well as other aerial scientific missions.

C. Background

Recent advances in computer-aided modeling, shape-memory materials (alloys, polymers etc.), design and manufacturing as well as in machine learning and artificial intelligence are partly responsible for the recent popularity of unmanned aerial drones. Together with newer, more accurate, low power and low cost (often MEMS based) miniature sensors, a whole new class of affordable unmanned aerial vehicles (UAV) designed for a variety of scientific missions has emerged.

The architecture of an appropriate UAV to sample volcanic plumes was published in 2005 by IEEE, which detailed a 4000 meter minimum altitude and using electric motors to avoid contaminating the gas samples [4]. The VOLCAN Project, published in 2008, is working on a single UAV to sample volcanic plumes which incorporate features like Hardware-In-the-Loop (HIL) simulations and an autonomous navigation system[5][6].

Nation Science Foundation (NSF)

II. THE UAVS

A. *Construction*

Each unmanned aerial vehicle (UAVs) is constructed from off the shelf components. We currently have two different models. Both utilize carbon-fiber reinforced expanded polyolefin (EPO) foam fuselages.

The Blixer 2 is a fast and maneuverable design. It has a 1.5 meter wingspan, weighs on 0.76 kilograms (without the installed electronics), and has a 0.5 kilogram payload potential. The plane's 1300 kilovolt brushless motor is powered by one 4000 milli-ampere-hour, 11.1 volt lithium polymer (LiPo) battery. This plane has an estimated flight time of 50-60 minutes. The Blixer 2 is aptly suited to measure data at a specific point or act as a relay for another plane. The Flyzone Calypso is almost identical to the Blixer 2 in power and weight, but has a 1.8 meter wingspan. This long wingspan makes the Calypso more suited to travel and sensing over long distances.

The best scenario would include both types of UAVs in each mission in order to increase the efficiency and ability to detect and analyze targeted areas.

B. *Autopilot and Scientific Sensors*

Each UAV is equipped with multiple sensors to take measurements of the surrounding area and guide the UAV. The Arduino Mega microcontroller is used to communicate over serial ports to the communication modules and to the autopilot, and read all the scientific instruments.

The autopilot used is an ArduPilot Mega 2.5. This includes a 3-axis gyro, accelerometer, and magnetometer as well as a high-performance barometer. It also interfaces with the on-board GPS and Pilot tube. The main advantages of this autopilot are that it is open source, low cost, and widely used. It has multiple modes of operation including waypoint navigation, stabilize (pilot-in-loop), and return to launch (RTL).

The scientific sensors were required to be light-weight, power-efficient, inexpensive and have a relatively high response time, though since our planes are moving relatively slowly, slower response times can be tolerated in favor of the other important qualities. We settled on two particle sensors, the Grove SEN12291P particle sensor and the Sharp GP2Y1010AU0F optical sensor, which can detect particles down to 1 micrometer and only use 90 milliamperes and 20 milliamperes of current respectively. The temperature and humidity sensor used is a Grove SEN51035P sensor. It is able to measure temperatures from -40 C to 80 C with 0.1 C resolution and humidity from 5 to 99 percent with 0.1 percent resolution. Cost effective sensors for the relevant gasses are still being sought.

C. *Communication*

The UAVs are each equipped with a low power consumption Digi XBee-PRO Series 2B radio in a relay configuration. The XBee line of radios relies on the ZigBee networking protocol (which rides on top of the IEEE 802.15.4 radio protocol). The ZigBee protocol produces a mobile ad-hoc network (MANET) for communications. A MANET allows mobile planes connect automatically to one another if they are in range without the need for a centralized or static network configuration. This means that in order to have a full network, UAVs just need to be directed close enough to one another allowing maximum flexibility. The radios are in relay configuration which involves a coordinator (a base station), routers (each relay UAV) and an end device (the UAV of intended communication) as seen in Fig. 1. By linking multiple relay UAVs together in this fashion the transmission range is only limited by the number available and line of sight obstacles can easily be overcome with strategic placement of the UAVs.

The communications system was tested under lab conditions. The signal of the end device was attenuated from approximately one mile to one meter and placed so it could no longer communicate with the coordinator. When the router was activated the signal information, temperature and humidity, was passed from the end device, through the router to the coordinator. When the end device was deactivated the information flow stopped, confirming it was the end device transmitting the data.

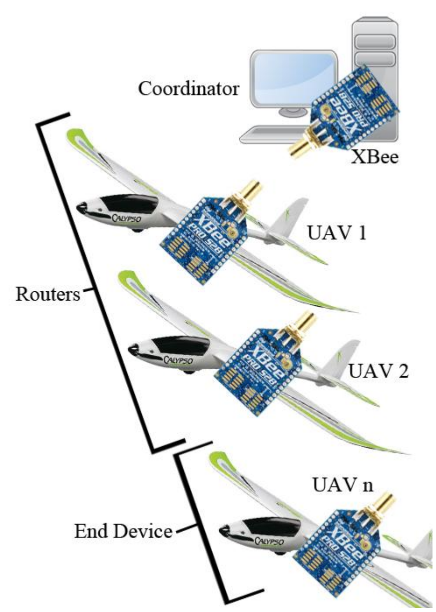
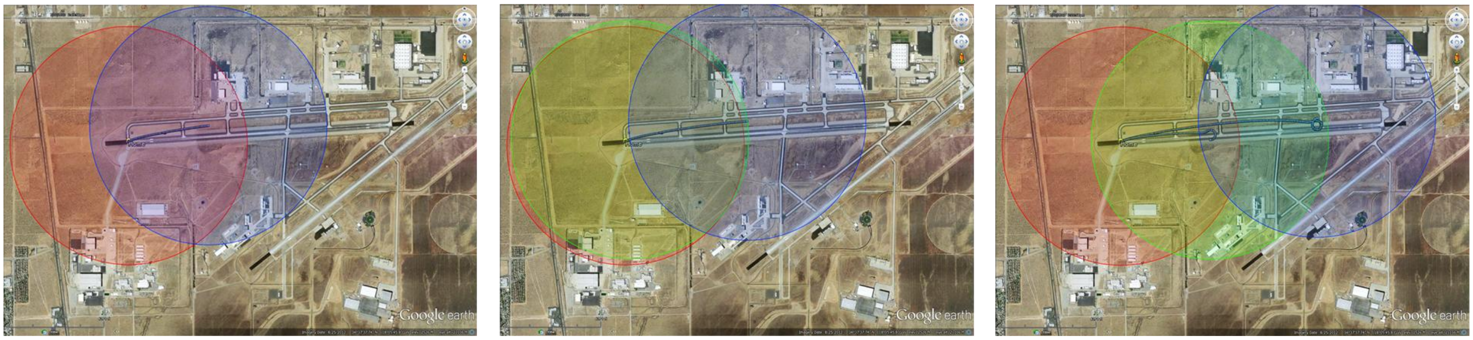


Fig. 1 Schematic representation of the proposed UAV network



(a) UAV 1 is nearing the edge of communication (b)UAV 2 is active and just leaving home base (c) All UAVs are at their destination

Figure 2. The communication ranges of the 3 nodes during a mission simulation

III. Simulation of Cooperative Navigation

A. *Simulation Tools*

In order to simulate flights and flight navigational plans, Hardware-In-the-Loop (HIL) simulations were used. By connecting the ArduPilot autopilot to our workstation computer, in conjunction with open source software developed by DIY Drones and a flight simulator as seen in Fig. 3, full flights with selected conditions could be tested. Additionally, the HIL set-up allows missions to be loaded and then deployed in a non-simulation setting; the flight we design can be taken by the actual plane.

The Xplane10 simulator was selected as the virtual flying environment due to its ability to interface with the flight planning component and its varied flight options including different planes, weathers and locations.

Mission Planner creates an interface between the ArduPilot and the flight simulator while allowing control over the flight patter and mode that the autopilot is engaging in. It utilizes point and click as well as scriptable missions, collects flight logs with extensive data based on the flight simulators presented conditions and relays this information to the autopilot [7].

Mission planner accepts Python scripts and allows a socket interface to send complex instructions to the autopilot and receive real-time data about the flight. This allowed us to create complex mission logic depending on plane's locations and distance to way points and distance from the main communication base. Additional way-points during flight could be sent using the specified GPS coordinates as well as a target altitude.

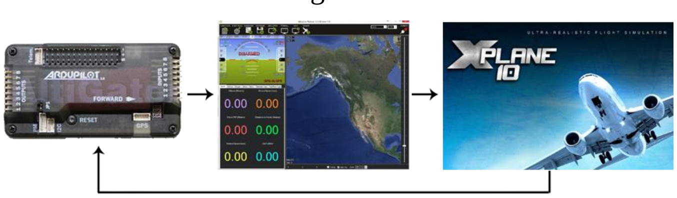


Fig. 2 Hardware-In-the-Loop (HIL) simulation feedback loop

B. *Single Point, Two Plane System*

A simple mission was designed in order to test the capabilities of the simulation software as well as design a basic mission script to start all more complicated scenarios from. The mission involved two main steps:

- Plane one leaves the home base headed to a specified way point out of range of the communications node at home base (Fig. 2-a).
- When plane one nears the edge of the communications range, plane two is sent (Fig. 2-b) to a point calculated at runtime where it can communicate with plane one's final destination and the home base simultaneously (Fig. 2-c).

Two planes were able to be simulated using two HIL systems and the multiplayer environment provided by the flight simulator. There were three main scripts, one which was run on Mission Planner to send the current location and distance data of plane one, a script to receive and decode the data, and a final one which calculated the ideal location for plane two and sent the way points to both planes at appropriate times.

During these simulations, the proper time to send plane two based on plane ones distance from home was determined. The planes should fly for as little time as possible due to their limited battery life, so finding the proper time to minimize flight time but still maintain constant communication with plane one is essential. With an ideal transmission range of 1 mile, plane one can travel .9 miles before sending plane two, but plane one stays barely within range at that wait period as seen in Fig. 4. Sending plane two when plane one has traveled .7 miles away from the home point, allows plane one to stay more firmly within the transmission range of the system. When deciding on the ideal time to send plane two, weather conditions and other obstacles that would affect communication range should be considered. For simulations purposes, sending plane two when plane one was .8 miles from the home point was selected.

Fig. 4 The distance of plane one from the nearest communication point when plane two leaves when plane one is .7 miles (red), and .9 miles (blue) from the home communication base.

C. *Multipoint, Two Plane Systems*

Allowing plane one to travel between multiple different way points increases the complexity of the mission plan and varies the data each plane is able to gather. Using the single point, two plane system as a base and adding script that sends plane one two the next way point in the array when it has reached the first way point. We found another script had to be added in order to refresh the port between each command. The flight path taken during this mission can be seen in Fig. 5.

Calculating the best place for plane two in this system became much more complex. With only one way point, the GPS coordinates of that point and the home base could be averaged to give a good approximation of the middle of the two and so an even transmission distance. Simply averaging the points with multiple way points biases the point for plane two toward any clusters which are a common configuration for plane one's way points. An acceptable location for plane two can be found using intersecting circles. The GPS coordinates are normalized as best as possible with each unit representing a mile. Then circle equations are found around the home base's location and each of the plane one's way point locations. The intersections of all these circles are calculated then normalized back into ordinary GPS coordinates. Using the Haversine method of GPS coordinate analysis, shown in (1), (2) and (3), the distance between the points of intersection and each location are determined; points which are outside the range of any location are discarded.

$$a = \sin^2(\frac{\Delta lat}{2}) + \cos(lat_1) \cdot \cos(lat_2) \cdot \sin^2(\frac{\Delta lon}{2}) \quad (1)$$

$$c = 2 * atan2(\sqrt{a}, \sqrt{1 - a}) \quad (2)$$

$$D = R_{earth} * c \quad (3)$$

Fig. 5 The flight paths of plane one (red) and plane two (green) when plane one has three way points.

These points are averaged to give an acceptable location for the planes to maintain a constant communication. This method guarantees to find an acceptable location if one exists without any bias toward a cluster of points.

Two plane, multipoint systems are limited in the arrangement of the points because each point must be within transmission range of a single point. The closeness of some of the way points can be seen in Fig. 5. Allowing the plane two to travel between points is not reasonable because ensuring the planes are in sync while flying is more complex than adding another plane to the system.

D. *Multiplane Systems*

There are two separate scenarios in which multiple planes would be needed to accomplish a particular flight plan. Either simultaneous sampling of two different areas is desired, or the desired points cannot be reached with a single jump, which can be caused by either a varied distribution of points, or too great a distance from the home point. In order to send multiple planes out in simulation, more workstations must be added to simulate each addition plane and their movements must be accounted for in the mission plan.

Being able to send and handle multiple planes sampling different areas is essential for the scientific missions planned. By adding each additional planes contact information and a loop, each plane can be sent to a different location. The problem of finding an appropriate location for the router plane reduces to that of multiple way points. While that solution may still have the issue of limited arrangements, that must be covered by the algorithm for multiple router planes.

In many instances more than one router plane is required in order to connect all end devices to the coordinator. The simplest case of additional router planes involves an end node with a single cluster of points reachable by one router that is more than double the transmission range away from the coordinator. In that case simply daisy chaining multiple router planes at even intervals across the distance would allow for communication. If way points of a single plane or of multiple end node planes are not within a cluster that can be reached by a single router plane, then different clusters must be formed and linked to the overall network. We are still developing the process which can accurately cluster and connect waypoints in varied locations that does so dynamically and efficiently. Once an algorithm is found that can do so, dynamically networking the planes to a communications center is possible.

IV. Conclusion

A flexible, decentralized network of affordable UAVs is proposed for volcanic plume sampling in its diffused state. The UAVs are built using small EPO-foam fulslages, equipped with low-power, high speed sensors for temperature, humidity, particles and gasses as well as an Arduino based autopilot and navigation sensors. The UAVs will communicate this data through XBee PRO radios using the ZigBee netoworking protocols in relay configuration. Through HIL simulation, networking solutions have been found for two plane systems with multiple way points and multiple plane systems with one way point. While a solution has been proposed for multiple plane systems with multiple way points, it is still being developed.

Acknowledgment

Candace Walden thanks the National Science Foundation for providing this research opportunity.

The authors thank NASA-UARC program for providing funding that enabled this research.

References

[1] Horrocks L., Burton M. and Francis P., "Stable gas plume composition measured by OP-FTIR spectroscopy at Masaya Volcano, Nicaragua, 1998-1999" Geophysical Research Letters 26(23), 2012.

[2] Shinohara H. "A new technique to estimate volcanic gas composition: plume measurements with a portable multi-sensor system", Journal of Volcanology and Geothermal Research, 143(4), 319-333, 2005.

[3] Grindle T and Burcham, Jr. F. W. *Engine Damage to a NASA DC-8-72 Airplane From a High-Altitude Encounter With a Disuse Volcanic Ash Cloud*, NASA/TM-2003-212030, 2003.

[4] Caltabiano D., Muscato G., Orlando A., Federico C., Giudice G., Guerrieri S., "Architecture of a UAV for volcanic gas sampling", 10th IEEE Conference on Emerging Technologies and Factory Automation 2005, 1, 744-750, September 2005.

[5] Astuti G., Longo D., Melita C. D., Muscato G. and OrlandoA. "HIL Tuning of UAV for Exploration of Risky Environments", International Journal of Advanced Robotic Systems, 5(4), 419-424, 2008.

[6] Astuti G., Giudice G., Longo D., Melita C. D., Muscato G. and Orlando A., "An overview of the Volcan Project: an UAS for exploration of volcanic environments", Journal of Intelligent and Robotic systems, 54(1), Springer, 21 July 2008.

[7] 3D Robotics. "Mission Planner." http://planner.ardupilot.com