Object Tracking using Drone Swarms

Project E25

Academic Supervisor: Richard Clare

Other Supervisors: Graeme Woodward, Mike Shurety, Dave Van Leeuwen

Team Members: Rowan Sinclair, Oli Dale, Nicholas Ranum, Connor O'Reilly, Alex Scott

Alex Scott

54208312

ars106@uclive.ac.nz

Executive Summary

There is a problem in entomology and other biological research fields, insects and other small creatures cannot be tracked by conventional methods. A GPS tracking chip is typically too heavy for the insects to travel effectively, and alternative methods of tracking are either not economically viable or restricted in usability.

This project was initially proposed by the University of Canterbury Forestry department's entomologist Steve Pawson while he was employed by Scion. The project seeks to use a swarm of drones to follow a target while tracking its location using harmonic radar. The Wireless Research Centre, Anastasia Lavrenko, and Scion have recruited engineering student groups in previous years to work on its completion. This year, the project is being run by the Wireless Research Center alone.

This project can be considered sustainable and beneficial for the environment. This is because the system's ability to further research into insects has a beneficial effect of the ecosystem and the system fairs well when examined under a triple bottom line analysis. Additionally, this system is more reusable and efficient than current alternatives.

As a continuation of the previous year's student group, this year uses the same equipment and work spaces and will not use additional materials. The current budget expended is $0 and is expected to continue with no spending for the foreseeable future.

The project has experienced initial delays getting the previous groups work running. However, the simulation is now operational and progress is expected to continue at a steady rate. The project is still expected to be completed on time.

My tasks revolved around failsafe implementation to ensure the safety of the drones during use. This involved research into the previous year's safety protocol, and the ground control software's failsafe function. The swarming logic appeared to have some issues involving multiple drones in a test flight. Due to complications with the previous year's data logging and presented conclusions, it was difficult to determine what had caused this problem. Research was done into potentially finding the cause, this produced unsubstantial conclusions.

Additionally, I have begun developing a computer vision landing program for identifying a safe area under the drone. This would replace the current landing function which lands directly with no regard for the, potentially hazardous, area below. The program still needs more development and optimization before it can be used by the drone's Intel NUC.

Table of Contents

1.0 Project Overview

The project being undertaken will use a swarm of drones to track a moving object. The drones should fly in a pyramid formation above the target with a central transmitter (mothership) drone directly above it and four receiver (slave) drones lower and offset both in longitude and latitude. The drones track and register their position using GPS. The target is tracked using a harmonic radar tag to return transmitted signals. The time to receive the signal again across the multiple slave drones is used to gain distance values and track its location.

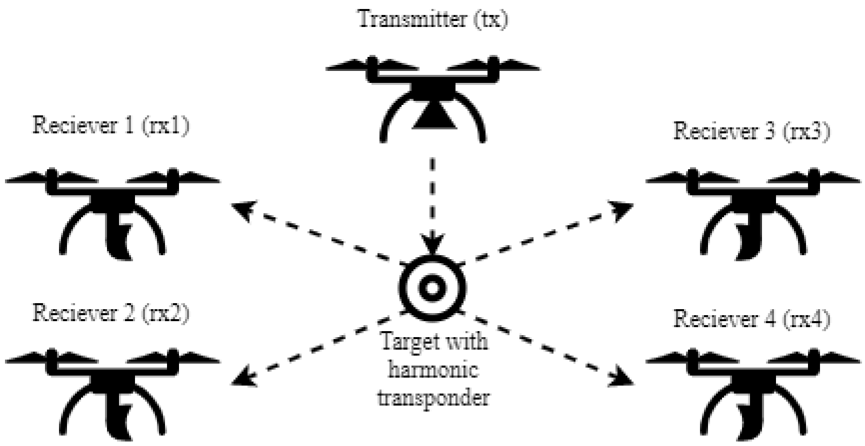

Fig 1: System diagram of the drone swarm

1.1 Project Purpose

The most common method of tracking a moving target in nature is by using a GPS chip. It is relatively cheap and requires no added components around the target. However, if a desired target is too small, a GPS chip is too heavy and is unable to be placed on the target. Insects are a good example of targets that are too small for a GPS chip. Furthering knowledge of their behaviour and movement patterns can be beneficial to businesses, conservation efforts, and academia.

Insects play an important part in our ecosystem. Bees alone are needed to produce one third of the global food production [2]. However, the population of terrestrial insects are declining at a rate of 9% per decade [3]. Also in New Zealand, it is important that our timber exports do not contain insects that could cause serious problems overseas. Therefore, the ability to track insect behaviour and swarm movements is of particular interest.

1.2 Project Sponsors

This project continues from a student group's work last year. They were able to fly a single drone to track a target with the other drones being simulated. While they had also attempted to fly two drones, this test experienced critical issues caused by false readings. With Anastasia Lavrenko, Steve Pawson, and Graeme Woodward's involvement, they were able to produce a research paper on their progress with the project so far called ‘Autonomous Swarm of UAVs for Tracking of Flying Insects with Harmonic Radar’ [1].

Last year’s student group worked with the Wireless Research Centre (WRC) and Scion, a company specialising in research and technology within forestry. The primary purpose with Scion was centred around tracking insects specifically. This year, the WRC is continuing the project with Graeme Woodward as head, but Scion is not continuing this year. However, previous Scion entomologist Steve Pawson has begun working at the University of Canterbury Forestry department. He is still in support of this project and its benefit to the field of biologic study, specifically entomology.

1.3 Materials Required

Our contribution is building upon the project from last year. Because of this, most of the components required to succeed in this project were provided by the WRC. The assets provided were four drones each equipped with a Pixhawk flight controller and an Intel NUC, one larger transmitting drone, a workspace to be able to test those drones, a GPS module to provide the target GPS data, and the GIT repository of the previous groups’ files and work. For a test flight, four qualified drone pilots are required be on hand for each of the drones to manually take control and land them should anything go wrong during the test flight. Other parts like a camera can be used in implementing failsafes, but the tools provided are all that is strictly needed for the formation to work.

1.4 What needs to be done/defining success

The task of our group in this project is to achieve a demonstration of all drones present and tracking a moving target. With the previous year’s accomplishment of flying a singular drone with the rest simulated, our group is required to make sure the drone formation works with all drones physically present. Additionally, improving on the previous tracking system needs to be done to improve its efficiency and reliability. A successful project would achieve these goals and have time to work on stretch goals like tracking increasingly complex target movements.

The completion of this project has been split into four separate milestones. Milestone 0 is to have the previous group’s work running and be able to log appropriate data during test flights. Milestone 1A is to have all four receiver drones successfully fly in formation while a central computer gives transmits their positions to them. This achieves flight without using any tracking. Milestone 1B is to have the drones track the target effectively, without the drones physically flying in formation. Milestone 2 is a combination of both milestones which is having the drones fly in formation while tracking a moving target successfully.

While the project involves a considerable amount of collaborative work, there are sections of it that can be worked on individually and they are split up evenly among our group. My section focuses on flight failsafes. This means ensuring the drones can react to problems during flight and act in a way that will not cause damage to them.

2.0 Current Progress

In the project, my work focuses around implementing failsafes to keep the drone safe while they are automatically flying. This is to account for low battery, loss of connection, drone damage, leaving the geofencing, and the drones going out of formation.

2.1 Research – Previous Year's Work

This project continues from another student groups work last year. The previous group was able to fly and track a target using a singular drone. This means that most of the research for this project was focused on the reports and files of the previous group rather than external sources.

The reports of the group covered flight controls, communications, multilateration calculations, and swarm formation. Flight control ensured the individual drones experienced a steady flight and set up the simulation on Gazebo, the multilateration report focused on tracking the targets position, and communications focused on ensuring reliable communications between the drones. Swarm formation is the most applicable to failsafes because it focuses on keeping the drones in formation, which is also critical to achieving the milestone 1A.

The swarm formation can monitor the position of each drone and check their position in relation to each other. If they are considerably out of position, the formation will stop tracking temporarily to correct itself. The code also implemented a failsafe when the swarm is critically out of formation, causing the drone to stop and remain in position until it is manually landed. Multiple conditions activate this failsafe, the drone's positions having a critical amount of error, two drones' positions overlapping by longitude or latitude, or the drones' being within 3 metres of each other. The results were reported to be quite successful. The swarm formation worked using the simulated environment, and in the test with one physical drone and other drones being simulation. But, while not mentioned in the report itself, the two flight tests using two physical drones both failed to due false critical formation readings.

2.2 Research – Pixhawk Failsafes

The Pixhawk flight controller is the system that controls the motors and, therefore, movements of the drone. Using QGroundControl, the flight of the drones can be monitored from a ground control position. Before the flight, QGroundControl can also be used to implement geofencing and certain failsafes. Geofencing ensures the drones remain in a desired area during operation. Because of the 730% increase in drone related airport incidents in recent years [4], it is critical that our operations stay in the desired area and do not cause problems for other aircraft. The geofencing will be set to have boundaries around Ilam Fields for the demonstration.

The QGroundControl failsafes monitor all potential issues an individual drone can experience during flight. These include low battery, connection losses, and geofence breaches. The battery monitoring can be set for both low and critical levels. Connection loss can monitor both the remote-control signals and data signals.

Once a failsafe has been triggered, the QGroundControl function can implement any of the following actions:

- Return to base, which involves flying vertically upward to a set altitude and safely returning to the home point.
- Hold mode, which holds the drone stationary in its current position in the air while adjusting against outside influences such as wind conditions.
- Land mode, which has the drone descend directly downward until it has landed.
- Warning, which sends a warning message back to ground control.

These four will find use under different conditions in the project. There are two other actions. Flight termination, which activated external failsafe measures like a parachute. Lockdown, which disables the motors and lets the drone drop. These two will not find use in the project.

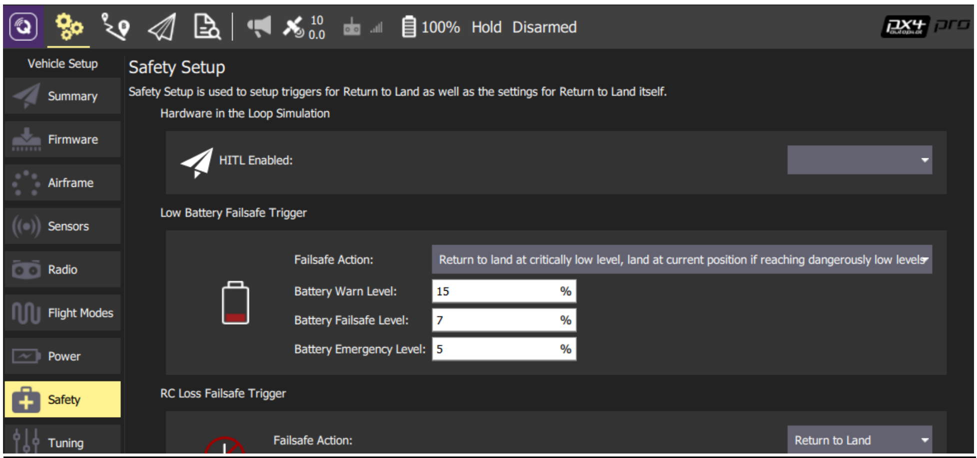

Fig 2: QGroundControl's safety settings panel [5]

2.3 Problems Encountered

The primary problem encountered was difficulty and delays achieving milestone 0: Replicating the work from last year. The simulation took longer than expected to get reliably operational. This was caused by numerous issues with Linux and general set up that needed to be troubleshooted. Team members Oli Dale and Nicholas Ranum were able to get the simulation operational, but only within the last two weeks from the date of this progress report.

Having no active simulation greatly affects progress on developing failsafes and achieving milestone 1A. Milestone 1A requires the drones to fly successfully in formation, and the swarm logic code from last year is already in place to accomplish that. However, further drone flight tests are required to determine the issue that the code is experiencing with multiple drones. Without the simulation, we were unable to conduct the required flight tests.

Another problem is the previous groups datalogging and reporting on results. The conclusions presented around the flight tests and the swarming logic appear to present a more favourable outcome than the test results showed. As mentioned earlier, the case with two drones in flight did not work and had problems that the swarm logic report does not address. The research paper concludes "a simplified setup with up to two UAVs" was able to track the target successfully [1]. However out of the two flight tests using two drones, only the first of them had the setup successfully track the target briefly before registering that it was critically out of formation. Because of the contradictions, a deeper look into the flight logs was deemed necessary to determine what error had occurred.

The git repository contains three sets of logs, the PX4 logs which tracked the movement of the drones during the flight tests, the ROS logs which tracked the commands going into the Pixhawk flight controller, and the flight-logs which was the data logging implemented by last year’s group.

Flight-logs contained a read me that detailed the three test flights, the one drone test which worked well, the first two drone test which worked initially then falsely registered a critically unsafe formation before it went stationary as per the safety requirement, the second two drone test did the same as the first test but failed earlier. The other files in flight logs show, drone positions which shows no data of the drones moving. The target positions and estimated target positions which appear to work well, but without a drone position reference cannot be evaluated properly.

I evaluated the PX4 and ROS logs to find more information about the two-drone flight tests:

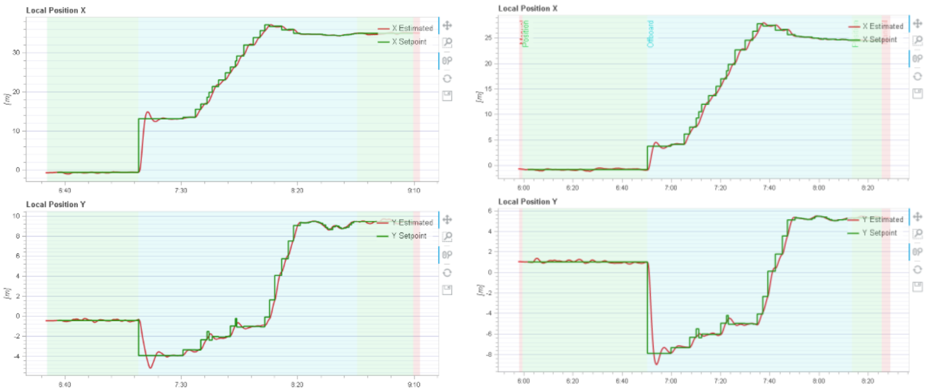

Fig 3: PX4 logs of both drones’ X and Y positions for the first test

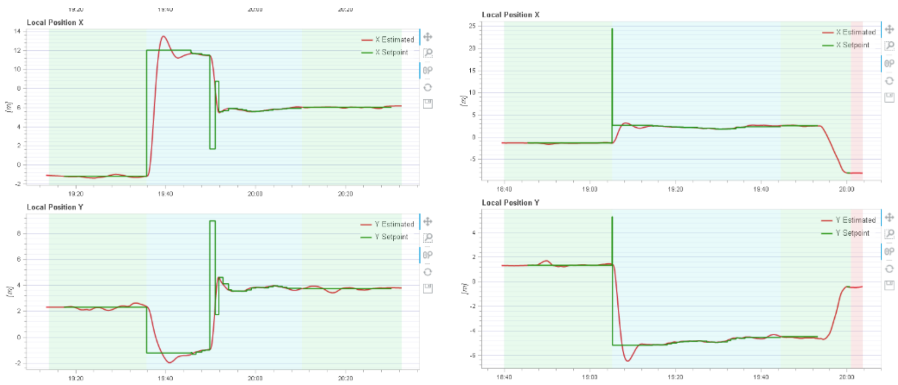

Fig 4: PX4 logs for the second test

To confirm the statement in flight-logs, the first test does appear to be stable and successfully tracking a target for a given time. However, the second test appears to go unstable almost immediately and the drones attempt to change their position very rapidly.

The ROS logs showed both the positions of each drone and the desired positions of the drones. The data showed no overlapping of the two in the X or Y coordinates and appeared to be moving steady and without increasing error before they went stationary.

The first test shows that the code can work with multiple physical drones and can track the target for a while. However, it still went stationary at the end as if it were critically out of position. The second test shows that the current system can go unstable at the start of automatic flight. The current theory is that an anomalous GPS reading must have caused the system to think the formation was unstable. Although, there is no presence of an outlier value in the ROS files so it must have been registered in the NUC only if that is the case. Another theory for the second flight is that the two drones started critically out of position before they switched to automatic flight which would explain the sudden jump in desired position. Further testing with proper data logging is needed with to be certain of the cause.

The final problem encountered was one with our team structure. Our initial leaderless model resulted in most weeks passing with every member working individually on their own section with no collaboration. As a team we decided to spend set times in the provided workspace working collaboratively. This increased our productivity and cohesion significantly and gave everyone in the team a good understanding of what each other member was doing. This also helped everyone keep accountable for work they needed to do.

2.4 Failsafe implementation and landing function

The setup for drone failsafes is as follows:

- Low battery issues a warning to the ground control so the operator can be aware of it.
- Critically low battery when operating in an open aerial space issues the drone to return to base.
- Critically low battery when operating in a crowded aerial space (tree canopy above or planes and other aircraft commonly above) the drone will land.
- All connection losses will cause the drones to enter hold mode until the connection returns. If it does not return after a certain time, the drone will follow the same protocol as critically low battery.
- A substantial breach in the geofencing will have the drones return to base.

The drone landing function is necessary in the system but could endanger the drone in its current state. It causes the drone to directly descend with no regard for the environment below. While useful in an unpopulated urban setting or open field where the ground below the drone is typically flat and safe, this does not apply to forests and other environments this project will need to track in. Descending in a forest without vision of the area below the drone could cause it serious damage.

To solve this problem, I looked at applying computer vision to survey the area below the drone and determine a safe area to land. Prior research on this topic showed few results. Most computer vision studies on automatic drone landing use fiducial markers, which indicate a predetermined safe area for the drone to land. So, most studies were not applicable to this application. One research paper uses monocular simultaneous localization and mapping (SLAM) to detect a safe landing zone. This paper gave a starting point for the program, but it starts from too high and suffers from scale drift to be directly applicable [6].

Scale drift is the when the camera effectively loses track of how big objects are from the receiving frames. While it can be accounted for, generally this will cause inaccuracy problems when operating for a long enough time which could cause a problem. Scale drift does not affect the mapping if it also receives depth values to read the distance, and therefore scaling, directly. Because of this, a D435 depth camera was chosen for this purpose.

The code designed for the landing protocol was made using OpenCV and Python. It has four steps for the procedure:

1. Use pyrealsense to get frames from both the RGB and depth camera.
2. Use the open3d's SLAM algorithm to create a 3d point map of the area from the frames provided.
3. Use pyransac3d's ransac algorithm to fit a plane to the 3d map, finding the greatest area of flat ground.
4. Create a box in the point map directly under the drone that is the proposed landing area. Determine whether that area is safe to land.

This program also reads the distance and angle between the camera and the plane. The distance can act as a secondary reading for the drone's altitude and the angle determines whether the plane is on an appropriate slope to land. It also requires multiple consecutive obstruction positive readings to register an obstruction. This is to combat noise which produces false readings in some anomalous frames.

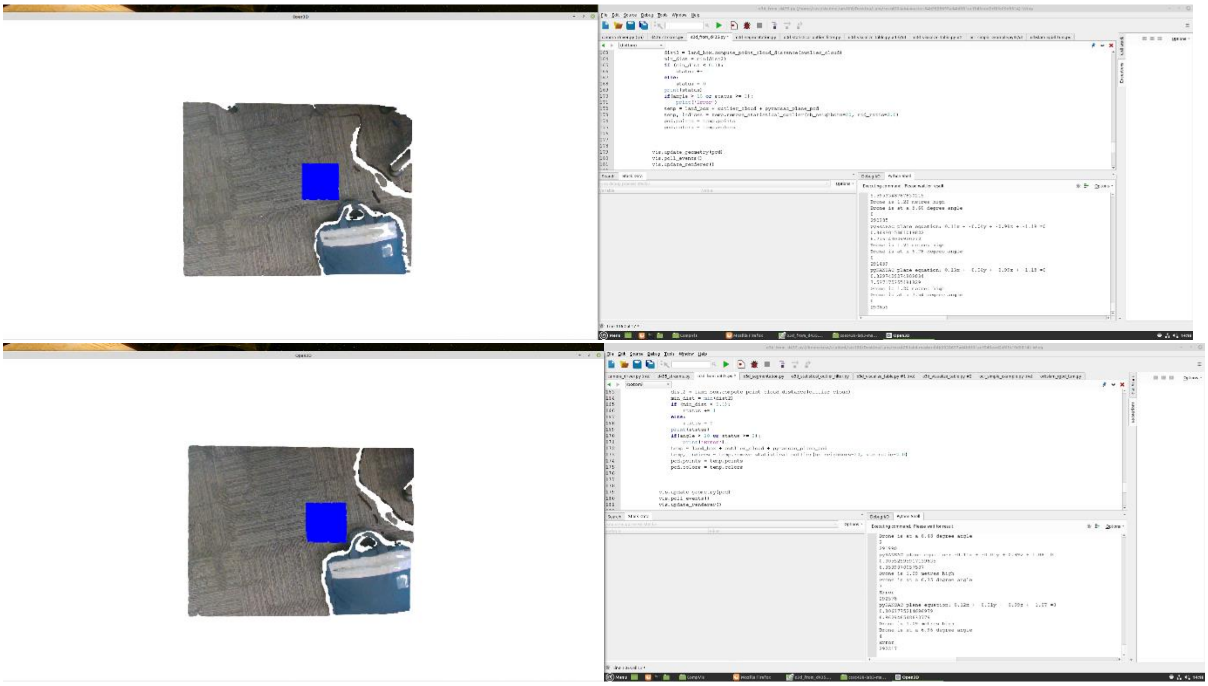

Fig 5: Results of landing protocol code. Safe landing case above, obstructed landing case below.

Shown in Fig 5, the blue box represents the area just below the drone. It can successfully detect when the bag is too close to the area to land safely, and it can detect if the angle of the ground is too steep to land.

The code still needs to be improved before it can be applied to the system. Primarily, the code fails to recognize obstacles that are considerably higher than the ground plane. This is because the function to detect distance between points works in three dimensions. So the difference in height is also read as a safe distance. Additionally, the code will need to be optimized and the point cloud will likely need to be down sampled from its current form. This is because the code is slow on a regular computer and will take up too much processing power when run by the Intel NUC.

3.0 Remaining Tasks

As mentioned in earlier, the project has experienced some setbacks and problems. Primarily getting the simulation working, which has only been recently solved. Because of that, all previously set milestones have yet to be completed. The goal continuing forth is to carry on with completing those milestones.

3.1 Task Outline

Table 1 details each member's tasks to be done and when they are expected to be completed. My tasks are highlighted in yellow.

Table 1: Future group tasks

Week	Milestone	Team Members				
		Oliver Dale	Rowan Sinclair	Nicholas Ranum	Alex Scott	Connor O'Reilly
Mid-year Break	M0 and M1B	Implement multilateration module and test using sim.	Incorporate Kalman Filter and test using sim.	Improve simulation software, launch files, and plotting tools.	Test simulation with two physical drones being held. Full flight test with 2 drones	Improve mothership-slave UAV dependency sync correction software. Success of data collation.
1+2		Test robustness of localisation for integration.	Add Loss detection to Kalman Filter and calibrate for practal test.	Investigate the effects of adding artificial network delays / constraints.	Achieve 2 physical drones flying, look to achieve all four drones flying in formation	Account for large higher-order derivatives (inertia / drift). Axial rotation around mothership. UAV GPS coordinate double check
3+4	M1A	Perform practical test.	Perform practical test.	Perform practical test.	Perform practical test.	Perform practical test.
5+6		Integrate localisation module into improved communication and swarming code.	Improve tracking capibility by measuring noise and reducing dropout.	Investigate methods of reading GPS timestamps to group radar readings.	Optimize landing function. Look to complete milestone 2	Improve swarming states, optimise code efficiency
Mid-sem Break	M2	Prepare for practical test.	Prepare for practical test.	Prepare for practical test	Prepare for practical test	Prepare for practical test
7	Prepare for Final Inspection					
8	**Final Inspection – September 24th**					
9						
10	**Oral Presentation – October 8th**					
11						
12	**Final Report – October 22nd**					

3.2 Fixing Drone Formation Errors and Milestone 1A

Now that the simulation is running, my priority will be focused on testing and troubleshooting errors in the swarm logic code. The problem being causing the false critical formation readings is still unknown. Appropriate data logging will be set up by team member Connor O'Reilly to record drone position, desired drone position, target position, and estimated target position as well as error messages. With this information being tracked, determining the error being caused will be significantly easier.

A qualified drone pilot is required for each individual drone when performing a test flight. It requires a considerable amount of coordination and organisation to set a test flight up. Because of this, the first tests of the 2-drone swarm formation will be performed without flight and instead have our team holding the drones and measuring the results. If the problem cannot be determined using this method, the flight tests will need to be done with data logging.

Once the problem with the two-drone flight is found and solved, the next step is to perform a successful flight with all four drones in the air. Given that a successful two-drone flight means that multiple drones can interact without error, implementing two more should not be as difficult.

Accomplishing four drones flying in a successful formation means that milestone 1A is completed. Team members, Oli Dale and Rowan Sinclair have made significant progress on the target tracking aspect of the project. They are now in the process of implementing their improved tracking code into the previous year's system. Once they have achieved that, milestone 1B will be completed and the group will progress into milestone 2.

3.3 Milestone 2

Milestone 2 is the combination of both milestones 1A and 1B, therefore the whole team will be working toward completing it. This requires all drone to fly in formation while tracking and following a moving target, which will be someone holding a laptop and walking. If we can achieve this, we will attempt tracking a faster target or perhaps one that is flying such as another smaller drone.

While this has the potential to work on the first attempt, it is a likely possibility that some unforeseen errors will occur in its implementation. Multiple test flights will be required to gain confidence in the system's ability. Once milestone 2 is achieved, the system will be ready for the live demonstration.

3.4 Landing Protocol

The landing protocol is beneficial to the overall project and can help with future work should the project continue in future years. However, it is not needed to complete the required milestones of the project and therefore is not a priority. The project can experience delays such as waiting for the next test flight. During down periods in milestone progress, I will continue working on this function. I expect to have the code optimized and useable on the NUC by the end of this project. An added stretch goal would be connecting a D435 camera to a singular drone, implementing the landing code into the NUC, and testing it.

3.5 Budget

This project has only used the tools and equipment provided by the WRC. We intend on maintaining that equipment, so it is usable by future student groups. Therefore, our group has not used any of the budget and are not expected to for the remainder of this project.

4.0 Sustainability Analysis

This project, upon completion, will allow for a reliable and cost-effective way to track insects and other small creatures. Out of 4000 insect species surveyed in New Zealand, 1000 were determined to be a threatened species. But more importantly 1500 of them were determined to be data deficient [7]. This means with roughly one quarter of insect species threatened, and more than one quarter cannot be accounted for. Lack of reliable, cost effective insect tracking solutions are a problem. A solution to this would be greatly beneficial to preserving this key part of the ecosystem.

This project will yield a positive sustainable outcome, however, the way in which it is implemented should also be sustainable. Therefore, this project will be compared to other alternatives such as retroreflective tags, one use chips (small self-sustaining chip that cannot be reused), and harmonic radar tracking using large stationary beacons.

Sustainable engineering and development means operating in a way that satisfies the needs of the present without harming the opportunities and ability of future generations. Performing a triple bottom line analysis means considering the environmental, social, and economics effects of the project, which will provide scope on whether the project is sustainable.

4.1 Environmental Effects

There are multiple environmental benefits to tracking insects:

- Tracking the movements of invasive insects that threaten native species could help reduce their damage on the ecosystem.
- Tracking threatened species and learning about their movement could help with preserving them.
- Tracking the movement of insects that infest harvested timber can bring confidence that certain logs are not infested. This would reduce the use of methyl bromide (a greenhouse gas) which is currently being used on all logs.

The environmental benefits of our solution compared to alternatives are:

- The drones use rechargeable batteries, durable materials, and have failsafes to prevent breakages. So, the usage of the drone should be as reusable as possible.
- In comparison to the one-use chip being used, our system can be reused multiple times as opposed to the chip which losses power in approximately 48 hours then cannot be recovered so it’s discarded in the environment [7].
- While further testing is needed, the power required to run the drone swarm system could be less than that used for the larger harmonic radar systems.

4.2 Social Effects

Common social issues around drones are:

- Airport and airspace disruption
- Noise pollution
- Operation over public areas or private property
- Drone invasions of privacy

Because our project operates in remote areas and utilizes geofencing to keep the drones there, these issues should be avoided.

Additionally, invasive species can sometimes be a source of fear. The ‘murder hornets’ are a great example of this, which have recently spread to the United States [7]. Better understanding these species and how to prevent them spreading will help will the civil unrest they can cause.

4.3 Economic Effects

As mentioned earlier, having greater knowledge of how insects move will help reduce use of harmful chemicals on logs. Having to spray less while keeping trade partners in timber is an economic benefit.

The greatest economic effect caused by this system is by being more effective than the alternatives:

- The one-use chip is not economically viable for surveying insects. It costs far too much for too little data. Our system would be cheaper and reusable.
- Retroreflective tags can be just as cheap as our system. However, they are far more reliant on line of sight and good lighting conditions. Our system would be more flexible and reliable.
- The large harmonic radars are not mobile. So, once an insect leaves the surveyed field, no data can be collected on it. They would also be more expensive to construct and only could survey one area. Our system would not experience these problems.

4.4 Summary

Our project is beneficial in all three sectors and would be better than current alternatives. Because of this, it can be considered sustainable.

5.0 Conclusion

While the project has experienced delays surrounding reproducing the previous years results, the simulation is now running, and practical testing will commence within the coming weeks. The research done has given the group a better understanding of where to go and how long each step in the process will take. The project is also progressing much faster with a restructuring of the teamwork style,.

The swarm logic code from the previous year is already able to achieve milestone 1A in a simulated environment. However, during a flight test with multiple physical drones it experienced an issue which caused it to falsely register a critically dangerous formation and go stationary mid-air. Progressing milestone 1A could not be done without the simulation running to test the swarming logic code. Because of this, I conducted research into the previous year’s data logging to potentially find the cause of this problem. This was not an ideal set of data, but some conclusions could be made. The problem was likely to be an anomalous GPS value that caused the false reading, but this cannot be confirmed because of the suboptimal data logging.

Another personal involvement of mine was starting development on a safe drone landing protocol. This was deemed necessary because the current landing function does not consider the, potentially dangerous, area below it. The program is currently able to detect obstacles but requires tweaks, improvements, and optimization before it can be effectively implemented.

The new multilateration and tracking algorithms are ready to be incorporated into the previous year’s code, the data logging has been implemented, and the simulation up and successfully running. Now the primary focus will revolve around practically testing the drones. With data logging in place, the swarm logic code can be properly tested for the issue that is preventing multiple drone flight. Once solved, this should complete milestone 1A. With the simulation running, the newly implemented tracking can be tested. Once it is successfully working, milestone 1B should be completed.

Once milestones 1A and 1B are completed, completing milestone 2 is simply combining the two. However, unforeseeable problems could occur and must be found and solved through further flight tests. With proper data logging implemented, this should be a streamlined process.

6.0 References

[1] A. Lavrenko, Z. Barry, R. Norman, C. Fraser, Y. Ma, G. Woodward, S. Pawson, “Autonomous Swarm of UAVs for Tracking of Flying Insects with Harmonic Radar”, 2021 IEEE 93rd Vehicular Technology Conference, 2021

[2] “The Importance of Bees, May 20 World Bee Day, https://www.worldbeeday.org/en/about/the-importance-of-bees.html#:~:text=The%20greatest%20contribution%20of%20bees,of%20food%20depends%20on%20pollination.&text=%E2%80%9DEnding%20hunger%20is%20everyone's%20responsibility.%E2%80%9D

[3] R. van Klink, D. Bowler, K. Gongalsky, A. Swengel, A. Gentile, “Meta-analysis reveals declines in terrestrial but increases in freshwater abundances”, *Science* Vol.368 Issue 6489, pp.417-420, Apr 2020

[4] K. Larbey, “Drone disruption at airports: A risk mitigation insurance response”, https://www.willistowerswatson.com/en-US/Insights/2019/07/drone-disruption-at-airports-a-risk-mitigation-and-insurance-response , July 2019

[5] “PX4 User Guide”, https://docs.px4.io/master/en/ , last updated: 14th March 2021

[6] T. Yang, P. Li, H. Zhang, J. Li, Z. Li, “Monocular Vision SLAM-Based UAV Autonomous Landing in Emergencies and Unknown Evironments”, *Electronics 2018,* **https://doi.org/10.3390/electronics7050073**, May 2018

[7] Steve Pawson, meeting with author, Christchurch, May 28th, 2021