The Extension and Implementation of the Autonomous Movement Framework

Théo Rivière
Graduate Student at Illinois Tech
triviere@hawk.iit.edu

Héctor Gutiérrez Ayala
Graduate Student at Illinois Tech
hectorgutierrezayala@gmail.com

Jeremy Hajek
Industry Associate Professor at Illinois Tech
hajek@iit.edu

ABSTRACT

The internet changes the way we do business, with companies like Amazon, Uber, and Google reshaping the way commerce is done delivering packages. Companies, such as Nokia, are demonstrating drone fleets being used for public safety over large scale desert areas. Our research asked, could this technology be replicated on a small scale for independent operators to use? The initial goal of this project was to design and develop a framework to control and manage drone fleets for use in search and rescue and disaster relief. We were able to design a platform and framework that integrated common off-the-shelf drones and accessible Windows computers and Android Phones to build and deploy our Autonomous Movement Framework.

Categories and Subject Descriptors

J.2 [Computer Applications]: Physical Sciences and Engineering – Aerospace.
J.7 [Computer Applications]: Computers in Other Systems – Command and Control.
K.5.1 [Computing Milieux]: Legal Aspects of Computing – Governmental Issues.
C.3 [Special-Purpose and Application-Based Systems]: Real-time and embedded systems

General Terms


Keywords

Delivery Drone, 3DR Iris, Quadcopter, UAV, Ground Station, Charging Pad, Python, MAVProxy, MAVLink, Mission Planner, Autopilot, open-source, Pixhawk, Automation, Rechargeable Batteries

1. INTRODUCTION

This paper is an extension of the Autonomous Movement Framework (AMF) research initially presented by Mark Milhouse in 2015[3]. Inspired by the 2012 “Taco Copter hoax [7],” the goal was to create a platform that allowed an autonomously piloted drone to make a delivery using GPS and a mobile phone application.

Though the Taco Copter was an internet hoax, the Autonomous Movement Framework was not. The initial prototype work showed this was feasible with off-the-shelf drones. Mr. Milhouse succeeded in creating the pillars of the framework by having a working application that could allow the drone to drop a package at a required location.

We have seen the Fire Department of New York deploy drones for use in fighting warehouse fires with great success [8]. Our investigations wanted to be able to replicate these abilities through commodity based hardware and software. Since 2015, the AMF project has sought to do just that. The AMF prototype has been expanded to four units and has built a new field recharging system to allow a quick and autonomous recharging of drone batteries. This paper will focus on the usability of the updated framework and the new charging system.

2. FRAMEWORK

2.1 Hardware components

2.1.1 Drones

For this project, four Unmanned Aerial Vehicles, or UAVs, were used; two 3D Robotics Iris Quadcopters [1] and two DJI F450 Quadcopters [2]. The Iris Quadcopters were chosen for their configurability, their modular build with easily replaceable parts and their durability, despite a limited carrying capacity and travel distance [3].

![Figure 1: Aerials’ Specs](image-url)

<table>
<thead>
<tr>
<th>3D Robotics Iris Quadcopters</th>
<th>DJI F450 Quadcopter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Battery</td>
<td>Battery</td>
</tr>
<tr>
<td>5100 mAh 3S 8C lithium polymer battery</td>
<td>5100 mAh 3S 8C lithium polymer battery</td>
</tr>
<tr>
<td>Motors</td>
<td>Motors</td>
</tr>
<tr>
<td>AC 2830, 950 kV</td>
<td>E305 2312E Motor (960kv, CW)</td>
</tr>
<tr>
<td>Propellers</td>
<td>Propellers</td>
</tr>
<tr>
<td>IRIS+ Propeller Set (10.5 x 4.7 inches)</td>
<td>9450 Self-tightening Propellers (9.5 x 4.5 inches)</td>
</tr>
<tr>
<td>GPS</td>
<td>GPS</td>
</tr>
<tr>
<td>uBlox GPS with integrated magnetometer</td>
<td>Radiolink SE100 GPS Module for PixHawk</td>
</tr>
<tr>
<td>Telemetry</td>
<td>Telemetry</td>
</tr>
<tr>
<td>3DR Radio 915mHz</td>
<td>3DR Radio 915mHz</td>
</tr>
</tbody>
</table>

Figure 1: Aerials’ Specs
Initial work began by augmenting the default software, Mission Planner, used to send our drones flight plans. The goal of this software is to easily perform alterations on the drone’s settings and carrying out tests and calibrations. Ubuntu Linux was used on a virtual machine installed on Windows computers to do this. The Linux environment was used because of the compatibility issues that the Mission Planner software had when working on Windows.

However, recent released versions of Mission Planner have considerably diminished the number of compatibility issues and, because of that, the tests can occur on Windows installed computers. This opens our Framework up to be used by the 100s of millions of Windows based computers.

The most recent tests were run on a MSI GS60 6QE running on Windows 10. The server was connected to the drones via a 915 MHz serial radio telemetry transceiver. It was also connected to the internet in order to be able to receive orders from our mobile application. The code and deployment instructions for this software are kept on the project’s Github account as we wanted to keep the project always available for access, download, and inspection.

Both types of drones use PixHawk Flight Controller. Pixhawk is an open-hardware project which goal is to provide high-end autopilot hardware [4]. The PixHawk is the core element of this project. The device has a set of inputs and outputs that are connected to the drone flight instruments, allowing to control it and have a feedback of flight data.

### 2.2 Software Components

#### 2.2.1 PixHawk Firmware

PixHawk Firmware gives the ability to easily modify features of the drone without modifying the source code. Features such as max speed or max altitude can be altered to adapt the drones to the requirement of the user. This is what allows Mission Planner, for example, to easily interact with the drone. The simplicity of use of the PixHawk was the main reason the Iris Quadcopter was chosen in the first place, and also the reason why it was used again with the DJI F450 Quadcopter.

#### 2.2.2 MAVProxy and Ground control station

The protocol to communicate between a ground control station and the UAV is what allows many kind of software, such as Mission Planner, to use a common set of functions on different kind of drones. The MAVlink protocol is a common standard of communication written in Python. In order to be able to send flight commands to our drones after sending an order from a phone, MAVproxy had to be modified. The phone app sends the coordinates to the server which then use MAVproxy to send flight data to the PixHawk that controls the drone. In order to simplify the whole process, the server has been designed in Python too. Originally, the server could only handle one drone trip at a time but, after some updates, it is now able to control many UAVs at a time. To do that, the ground station reprograms the single radio’s netID to the one corresponding to the desired drone. By this we gain a many to 1 relationship (1 radio to ~500 radios). But in the case of reprogramming netIDs we lose the ability for the drones to use their radio for sending telemetry back to a base station since we are constantly re-programming the base radio’s netID, there is nowhere to return and signal too. Solutions for that are being considered today. One would be to have another station on 915 MHz out-of-band, that would collect all broadcasted telemetry for logging purposes and in software we would sort and parse the data.

#### 2.2.3 Mobile Phone Application

The front end of the framework is the phone app. After many iterations and discussion, the application is OpenSource and runs on the Android platform. The app functions in the following way; the client is the mobile phone app, which would be a worker in the field needing a drone to bring equipment to their location. They would launch our app on their phone, upon hitting the “Request Order Now” button using the Google Maps API, the GPS position of the phone would be sent to the command and control server via the Internet (3G, Wifi, etc.).

Then, as it can be seen in figure 4, a location must be chosen and then, the user must click on “Request order now”. A message box will pop up to ask the user to confirm the location. After that, a confirmation message will appear to announce the departure of the drone. The benefit of this method is that the phone’s GPS would default to the user’s location, but by integrating the Google Maps API, we now give the user the ability to “send” the drone to an area that is a different location, and give the ability for drones to “meet you there.” This is a key feature of our Autonomous Movement Fleet.

GPS accuracy varies per device and location, but our target has never been precision accuracy but relative accuracy.

---

*Figure 2: 3D Robotics Iris Quadcopters*

*Figure 3: DJI F450 Quadcopter with 3D-printed shell*
3. THE BATTERY CHARGE

In order to reach the goal of developing an autonomous fleet of drones, many individual aspects of the system have to turn into self-governed features. Along with those aspects, there is the charging process of the quadcopter. This is the most complicated process from an automation standpoint, but the most critical in regard to being able to operate; no battery no flying.

3.1. Autonomous charging systems in the market

For solving the non-autonomous charging issue, the first task that has to be done is the research of the current market. Currently, the most common “autonomous charging system” implemented in the market would be the wireless pads for charging mobile devices also known as Qi charging. The charging mechanism is based on inductive charging technology, which allows a pad to charge a compatible phone just by placing it on the charging pad.

The inductive charging system works as follow: the device transfers energy from the charging station to the battery of the phone using electromagnetic induction, which is based in the principle that electricity travelling through a conductor produces a magnetic field around it. This magnetic field can induce current in a wire nearby as long as the wire is inside the range of the field. Besides, the magnetic field is concentrated by coiling the wire, having a more intense field that can easier induce current into the other wire. According to this, in the charging device there would be two wire coils: one for transmitting energy from the pad, and another one to receive the energy at the phone device.

But, apart from the chargers for mobile devices, there are other fields where autonomous charging is used, even if they are not as well-known. For example, Plugless [5] has developed a way of charging electric vehicles wirelessly. This charger adds comfort in the use of electric vehicles; it allows the driver to charge its car by parking on the right spot, where a charging plate is located. No further actions being required.

3.2. Inductive charging vs direct contact

After researching the market products, the following step would be to design the autonomous charger. But, which path would be better to take? The inductive charging or the direct charging? It is true that both of them could work, but there are some advantages and disadvantages related to them that had to be considered:

- **Maintenance**: with inductive charging the contact between the parts is reduced. This is an indirect measurement of the wear that the parts would undergo and, thus, the maintenance needed. Furthermore, the inductive charging provides a higher isolation of the components. So, they would be more protected from corrosion or electrical failure and the system would need less maintenance.

- **Energy efficiency**: the biggest disadvantage related with the inductive charging against the direct contact is that the system is much less efficient than the conventional charger, wasting more energy in the process and needing more time to charge the entire battery.

- **Costs**: inductive charging tends to be more complex, carrying higher investment costs than the conventional charging system. The energy cost would be higher because of the lower charging efficiency of the inductive technology. However, the costs related to the maintenance, which would be higher for the direct contact, have to be also taken into account.

3.3. The charging mechanism

Both mechanisms have the same main problem: how to place the drone in the exact spot where it is going to be charged? There are basically two ways of placing the quadcopter in the charging spot:

The first option would be to land the UAV directly onto the plug/charging-pad, which, at first glance, is easier in inductive charging than in direct contact. Indeed, the surface covered by the wire coil used for inductive charging is greater than the one covered by a plug used in direct contact. But the commercial GPS margin of error is too large for this operation (4.9 meters) in our framework [9]. If this option is desired to be used, thus, the drone would need to be assisted by some kind of mechanism at the landing time, with the aim of increasing the accuracy of the location of the UAV. One example of assisting the drone would be to add to it a camera and do pattern matching or use computer vision to land or use a physical mechanism in the
landing spot that would place the drone in the correct location wherever it lands. At that exact location, even inductive or direct contact could charge the battery.

But, what if it was not necessary to approach the UAV to an exact location, and the drone could land within the GPS margin of error as we have built a modular landing pad that works as a charging pad at a greater size than this margin or error? We would modify the units to contain direct charging plates that could be deployed successfully upon landing and this would be called the Autonomous Charging Framework. Initial provisional patent research is ongoing and is the reason our details here are light.

4. RESULTS

4.1. July 2016 Demo

On July 2016, a demonstration took place at Illinois Institute of Technology to show the proof of concept [6] which was a success. The Iris drones were used for this demo as the DJI F450 were not available during that phase of the project. At this time, it was already possible to fly two different drones on different delivery travels. The demonstration was a success having the AMF manage two independent deliveries. One thing to note, the initial test was done with xBlox GPS receivers, the current research is being undertaken with a newer Radiolink SE100. Consumer GPS has a plus or minus of 5-10 meters. Which the project was willing to trade specific accuracy for usability.

Furthermore, during this demo, two separate antennas were used to control both UAVs. Indeed, it was required to have an antenna for each drone in flight. However, the AMF makes it possible to control several drones with one single reprogrammable antenna.

![Figure 5: Two drones flying at the same time during July 2016 Demo](image)

4.2. Discussion of test results

4.2.1. Enhancing the framework

Our framework was a success with 2 units showing us that multiple units would be possible. Having multiple autonomous units in the air now brings new challenges.

4.2.2 Air Traffic Control

Theories of management of units in space will need to be developed. While obstacle avoidance and computer vision are suggested starting points, we believe a 3-dimensional pre-programmed grid may be the most efficient way to proceed.

Integration with the FAA when our service is in use would be a critical software piece to provide to them and any local emergency service, via a API.

4.2.3 Transportation

As the number of units increase, the question of how to we box them and ship them? If we have 500 or 5000 or 50000 units? This presents a logistics issue that would need to be further studied. Do we create custom carrying cases? Can we build a custom shipping container? What about an air drop or even a rail-car based way to transport our fleet? These need to be considered and researched.

4.2.4 Manufacturing

Using off-the-shelf drones for the proof of concept has a high cost. Investigation of materials and manufacturing capabilities and economies of scale. Can the drone shells be 3D printed, or stamped aluminum, or another source altogether?

4.2.5 Radio communication security and telemetry collection

A method will need to be devised to be able to collect the telemetry from each drone in the air as it broadcasts and this method will need to be secure to prevent hackers from injecting or broadcasting fake data.

4.2.6. Increasing the automatization

The Autonomous Movement Framework started with the goal to be able to control drone movement. It swamps into fleet management, and now covers fleet control, charging, and deployment. Beyond cheap advertising tricks and delivering pizzas we believe this framework can be used successfully in aiding search and rescue and natural disaster relief but extending human carrying and deployment capabilities.

5. ACKNOWLEDGMENTS

This research would not be possible without the support of Professor Jeremy Hajek, who is the head of the Autonomous Movement Framework project. Along with Jeremy, thanks to the rest of the team for the helping and working along through the different steps of the project, as well as Dr. Krishnamurthy, who has shown interest and has gotten involved in the project.

6. REFERENCES