

# Autonomous Quadcopter Delivery System

HASAN-MUDHAFAR QAYS, AYMAN-DAWOOD SALMAN AND  
BASSIM-ABDUL BAKI JUM.  
*Computer Engineering Department  
University of Technology  
Baghdad, Iraq*

Drones have been used in many applications such as military, security & surveillance, service delivery, disaster rescue and much more due to its flexibility of flying. This paper introduces a system for delivery service that can be used in different scenarios. The proposed system is based on the Ardupilot software, autonomous flight algorithm, path planning using Next Nearest Location First algorithm (NNLF), user notification module, and power estimation module. Raspberry PI 3 is connected with the flight controller in order to send the appropriate MAVLink messages to the Ardupilot that is running on Pixhawk flight controller. The system takes the users requests through a website based on Google maps. DroneKit-Python Application programming interface (API) is used to develop the required software that interact with the flight controller. MySQL database is used to maintain all the data from users, quadcopter and the admin of the system. The proposed system supports dynamic multiple destination points. The features that the system support make it useful in wide range of applications.

**Keywords:** Autonomous quadcopter, Ardupilot, MAVLink, DroneKit-Python, Raspberry PI 3, Path planning, Power estimation, Flight controller, Pixhawk

## 1. INTRODUCTION

The Unmanned Aerial Vehicle (UAV) is a vehicle without a pilot on it. There is a system for the UAV called the Unmanned Aerial System (UAS) that allows communication with the physical drone [1].

Drones can be manually controlled by a human pilot which is a user input using the remote control that is known as Radio Controller (RC). But also, they can be autonomously controlled by the system integrated on the drones their-self without using the RC input. The UAS is installed on the companion PC which is called onboard computer [2],[3]. The drone was initially developed for a military purpose, had its test flight before World War II, and then was fully deployed in the Iraq War and the Afghan War from around 2000. [4]. A reliable UAV was appeared in the 1990s, and research and development of it became active in universities and other academic fields. In Georgia Institute of Technology, the first international flying robot competition was held in 1991 [5]. This was started as a university event but gradually expanded worldwide into the field of small unmanned aerial vehicle among government, industry and academia. Most of the studies focused on issues of higher performance hardware platforms, software systems, aerodynamic models, and autonomous flight control, and these state-of-the-art technologies competed. In addition, (Defense Advanced Research Project Agency) (DARPA) set up in 1997 a project of MAV

(Micro Air Vehicle), which led to the development of small drone with the diameter not exceeding 15 cm at an annual research funding of 35 million USD [4]. In 2009, 3D Robotics and the Swiss Federal Institute of Technology in Zurich introduce the open-source Autopilot system named (ArduPilot Mega) (APM) and its newer versions named Pixhawk [4]. Many versions were released such as ArduPilot in 2009, APM1 in 2010, APM2 in 2011, APM 2.5 in 2012 and Pixhawk in 2013. According to the Federal Aviation Administration (FAV), drone sales are expected to grow from 2.5 million in 2016 to 7 million in 2020. This massive popularity of drones is driven by the increasing sensing and communication capabilities combined with their falling prices. Modern drones are now equipped with a large number of sensors, such as temperature, humidity, air pollution, GPS, 4K camera, etc., as well as an expanding range of wireless communication capabilities including Wi-Fi, Bluetooth, 4G, RFID, and so on. Finally, drones have also improved in mechanical performance, such as flying speed, degree of autonomy, agility and are available in a variety of form factors [6].

These developments have made drones a very attractive platform to launch many Internet of Things (IoT) applications for examples: A drone equipped with the right sensors can be used to collect specific air quality data from hard to reach places in a smart city and transmit that data to an IoT server in real-time using the internet or Wi-Fi [6]. Some organizations are using UAVs specifically because of this high level of interest. Good examples are the world's largest online retailer, Amazon and the world's largest logistics company DHL, which are testing UAV delivery [7]. Many applications where the interfere of human teams is costly, dangerous or even infeasible; such as military, emergency management, search and rescue operations [8]. These applications and others are the subjects of the next section, which discuss how the researchers used the UAVs in different systems and applications. There are three major kinds of drones: fixed wings, single motor and multirotor. The multirotor is the most common among them, they are classified depending on the number of motors they have such as tri-copter (three motors), Quadcopter (QC, four motors), hexacopter (six motors) and octocopter (eight motors). The most common multirotor is quadcopter [9]. The traditional fixed-wing UAVs can fly for long distance but require runways or wide-open spaces for take-off and landing. On the other hand, the more trending multirotor UAVs are extremely manoeuvrable but cannot be used for long-distance flights because of their slower speeds and relatively higher consumption of energy. The flight of the multi-rotor is more flexible compared with a single rotor and the fixed-wing in terms of takeoff, landing and controlling the manoeuvring [10].

## 2. RELATED WORKS

Several studies have been devoted and deployed with various issues towards design and implementation of using quadcopter for many applications. An examples on how a drone could be utilize in an everyday life context is given in [11], where the authors designed a system that uses the drone in agricultural monitoring. The agricultural farm is surveyed by an infrared camera which will show the colour image displaying the difference between infected or diseased crop and matured crop. The authors in [2], introduced auton-

omous quadcopter for product home delivery using the android device as on-board processing unit and connecting to APM flight controller. The context in [3], provided a system of quadcopter for drug shipments. They introduce a very useful android application to be used by the client and pharmacies to request such a medicine. Author in [12], introduce the design of quadcopter to be guided autonomously using GPS module. In health care field the author in [13], provided a study to use drones for clinical specimens in rural areas. This study uses a Travel salesman algorithm (TSP) to implement multiple destination delivery to visit all the points only once for the shortest path. Another useful example to transfer video using quadcopter in [8], where the authors introduced a monitoring system using quadcopter with an attached camera to the Raspberry PI to evaluate the video transfer using different protocols. This study introduced an interesting evaluation for video transfer using Ad-hoc network depending on the quadcopter. The authors in [14], introduced a new platform to control, monitoring the drone status and managing the mission of multiple drones at the same time.

### 3. PROBLEM STATEMENT AND MOTIVATION

As listed in the literature survey the problem statements are , A fully autonomous drone is required. A dynamic path planning algorithm is required especially when the drone flying and receive new destination points. The power estimation must be implemented to any application that uses the drone since it has a limited time of flying. The Email notifications is a very useful way to inform the users or the admin regarding the status of the drone or sending the collecting data to the server. The web interface, that allows the users to send their request to the drone is required for easy to use the system. To overcome these drawbacks, the proposed system in this paper provide the following solutions: An **algorithm of fully autonomous drone** using the RPI3 as an on-board processing unit, Introduce a fast and simple algorithm for path planning **Next Nearest Location First** (NNLF) algorithm is used for multiple destination points. This algorithm takes into consideration any new points added by the user and approved by the admin of the system while the drone is flying, Power estimation module that read the estimated power level of the battery from the auto-pilot and checks if the available power is enough to fly to the next destination point or not, **Notification module** that uses the Gmail services to send an email notification to the registered users and to the admin too and **User interface module** that provides a website using google maps to let the users send their requests easily.

This system can be used by other researchers in a different application not only for mail delivery, such as monitoring, video transfer, surveillance areas that considered dangerous for human, for example, detect gases leakage.

### 4. THE PROPOSED SYSTEM

The proposed system, as shown in (Fig. 1), has two main parts:

1. The *first part* is server-side which consist of *Ubuntu operating system, MySQL database server, path calculation module* and the *webserver*.

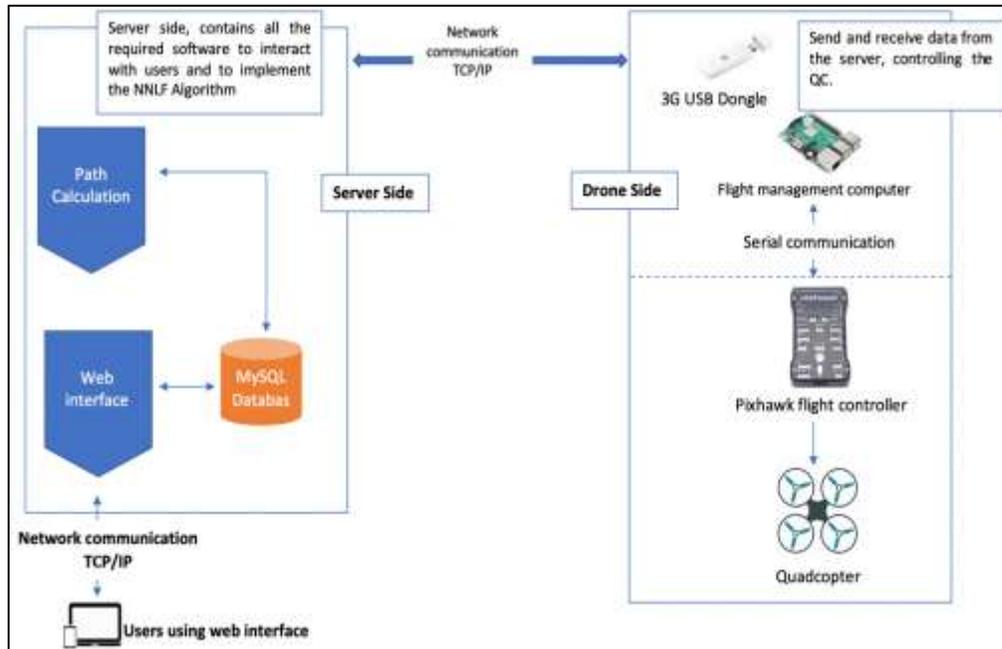


Fig. 1 General design of the proposed system.

2. The *second part* is the quadcopter side which consists of: *quadcopter hardware, Raspberry PI 3, autonomous flight module, power estimation module* and the *user notification module*.

#### 4.1 Ardupilot

Autopilot is an essential part of the proposed system. Ardupilot is an open-source autopilot software that is based on MAVLink protocol. It can be used to monitor, read sensors input, and control the vehicle according to the data received from the sensors. Ardupilot can control many types of vehicles such as drones, rovers, and boats. This software is cross-platform system that can be run on different operating system and boards such as (Pixhawk and APM). In this paper the Pixhawk flight controller is used to control the quadcopter. (Fig. 2) shows the Ardupilot architecture and describes the content of each layer [15].

### 4.2 Pixhawk Flight Controller

Pixhawk is an independent open-hardware project that aims to provide the standard for readily-available, high-quality and low-cost autopilot hardware designs for the academic, hobby and developer communities. Pixhawk supports multiple flight stacks: PX4 and Ardupilot. This Board has the following sensors which are used in this paper to control the quadcopter:

- ST Micro L3GD20H 16-bit **gyroscope**
- ST Micro LSM303D 14-bit **accelerometer/magnetometer**
- Invensense MPU 6000 3-axis **accelerometer/gyroscope**
- MEAS MS5611 **barometer**

Also, Pixhawk contains many interfaces that used to connect different hardware. The UART interface is used to connect the Raspberry PI 3 with Pixhawk as on-board control unit as described in (Fig. 3) [9].

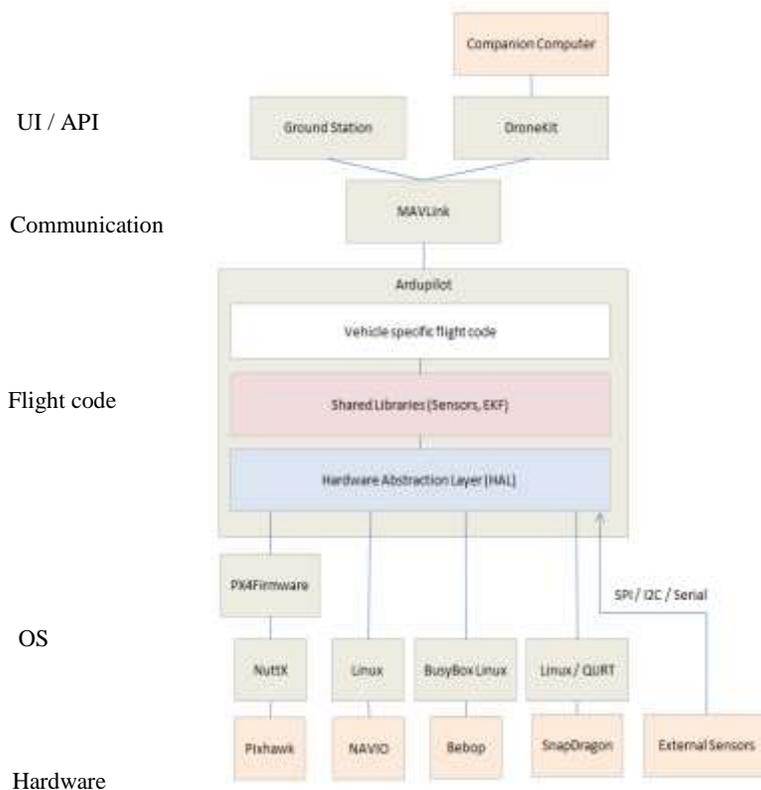


Fig. 2 Ardupilot architecture.

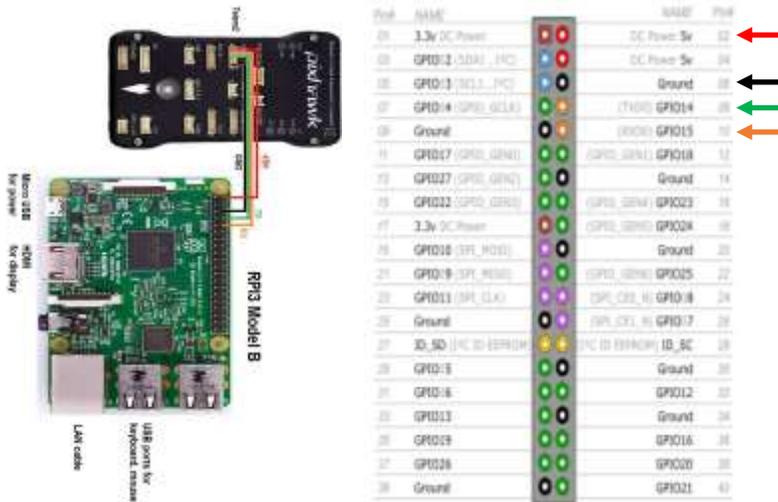


Fig. 3 Connection RPI3 with Pixhawk Microcontroller.

### 4.3 DroneKit-Python API

It is a cross-platform Application programming interface that can be used on many different operating systems. It helps to produce applications for Drones or any MAVLink capable vehicle. This API has many versions for each developing environment. It is available for Android OS as DroneKit-Android, and for Python programming language as DroneKit-Python and for IOS devices but it's still under development. There are two methods available to control the movement of the quadcopter: Target position and Vehicle's velocity. These two methods are used according to the knowledge of the destination point. If the destination point is fixed and known then the target position must be used while if the destination point is not fixed or unknown then vehicle velocity method must be used [16]. In this paper the first method (target position) is used because the destination points are always fixed and known.

### 4.4 Server-Side

The server is responsible for *managing the MySQL database, path calculation module* and *hosting the web interface* which is used by the users to interact with the system.

**(A) Path Calculation Module:**

This module is depending on Haversine formula to calculate the distances between two locations. The haversine formula is an equation important in navigation, giving great-circle distances between two points on a sphere from their longitudes and latitudes. It is a special case of a more general formula in spherical trigonometry, the law of haversines, relating the sides and angles of spherical triangles. The first table of haversines in English was published by James Andrew in 1805, [17]. Florian Cajori credits an earlier use by Jose de Mendoza y Ríos in 1801[18]. The term haversine was coined in 1835 by Prof. James Inman [19]. For any two points on a sphere, the haversine of the central angle between them is given by the Eq. (1).

$$\text{Haversin} \left( \frac{d}{r} \right) = \text{haversin}(\varphi_2 - \varphi_1) + \cos(\varphi_1) \cos(\varphi_2) \text{haversin}(\lambda_2 - \lambda_1) \quad (1)$$

Where:

- **Haversin** is the function of :

$$\text{haversin}(\varphi) = \sin^2 \left( \frac{\varphi}{2} \right) = 1 - \frac{\cos(\varphi)}{2} \quad (2)$$

- **d** is the distance between the two points.
- **r** is the radius of the sphere.
- **$\varphi_1, \varphi_2$** : latitude of point 1 and latitude of point 2.
- **$\lambda_1, \lambda_2$** : longitude of point 1 and longitude of point 2.

On the left side of the equals sign  $d/r$  is the central angle, assuming angles are measured in radians (note that  $\varphi$  and  $\lambda$  can be converted from degrees to radians by multiplying by  $\pi/180$  as usual). Solve for  $d$  by applying the inverse haversine (if available) or by using the arcsine (inverse sine) function as in Eq. (3):

$$d = r \text{haversin}^{-1}(h) = 2r \arcsin(\sqrt{h}) \quad (3)$$

Where:  $h$  is haversin ( $d/r$ ), or as in Eq. (4):

This formula was implemented using python in the path calculation module. When the quadcopter has more than one destination point to visit it, the path planning is required. As the quadcopter has limited flight time, therefore shortest path or best path to fly is needed. The Next Nearest Location First (NNLF) algorithm is used to let the quadcopter always fly to the next nearest location. The algorithm depends on the distances between the destination points and the quadcopter.

The flow chart in (Fig. 4) describes how to calculate the distances. This module takes into considerations any new request to the system while the quadcopter implement the current mission. If the users add new request and the admin approve that request this module will include these new destination points into its calculations.

**(B) Web Interface Module:**

The web interface is developed from scratch using PHP, jQuery, Bootstrap and integrated with google map. The system has two privileges at the software level: *administrator* and the *users*. The admin has full privileges and he should check all the waypoints before submitted to the server. After checking all the data, the mission will be approved and send all the waypoints to the server.

**(C) Database Server:**

MySQL database is used to manage all the data from users and from the quadcopter. The idea of determining the status of each destination point and quadcopter status is depending on status flags that are created in different tables. Noting that while the quadcopter is flying or implement a particular mission, the system is able to receive new locations entered by users and approved by the admin. When that is done the path calculating module will consider these new locations in the calculation. The path calculating module will calculate the distances between quadcopter location and all other destination points that have status flags non-reached and location\_status is approved. When the calculation is done, it updates the distance field in the locations table with these values.

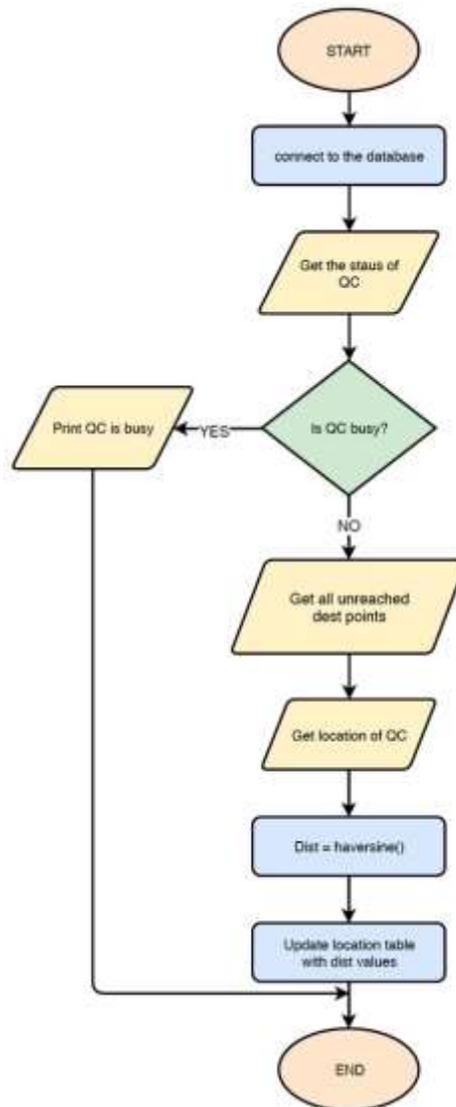


Fig. 4 Path Calculation Module.

#### 4.5 Quadcopter Side

Quadcopter – side consists of the *hardware of the quadcopter* and the *Raspberry PI 3 as onboard control unit (OCU)*. OCU is the companion PC which commands the quadcopter. It's connected with the flight controller directly using a serial connection. The quadcopter side has the following modules:

**(A) Autonomous flight module:**

This module is used to implement the autonomous flight, and it's integrated with other modules. This module is used to control the quadcopter without using RC input. It implements autonomous flight using GUIDED mode of Ardupilot. (Fig. 5) shows the whole algorithm of drone delivery system. The main loop is depend on the following functions: *UpdateInfo\_busy()*, *NearstLocation()*, *sendEmailRPI()* and *goto()* with Power estimation module.

1. The *first* function is *UpdateInfo\_busy()* which is used to change the status of the quadcopter by update the status flag *robo\_busy = 1*.
2. The second function is *NearstLocation()*, which is used to get the nearest destination point to the quadcopter according to the distance value in the location table and it returns the point with minimum distance to the quadcopter.
3. The third function is *sendEmailRPI()* used to send the emails to a particular user.
4. The fourth function is *goto()*, used to command the quadcopter to fly to any destination point. This function takes one parameter which is the destination point. When the quadcopter reaches the destination point the OCU use the function called *emailReach()* to send the second notification email to inform the user that the quadcopter is reached.

**(B) Power Estimation Module:**

used to check the available power and determines if it's enough for flying to next destination point or not. At each point the OCU use the Power Estimation Module (PEM) as following:

1. Get the previous distance (*pdist*).
2. Get the next distance (*ndist*).
3. Get the starting battery level (*slevel*).
4. Get the current battery level (*clevel*).
5. Calculate the dissipation ( $diss = slevel - clevel$ ).
6. Calculate the required power per meter (*PPM*) at the previous distance.
7. ( $PPM = \frac{diss}{pdist}$ ).
8. Calculate the required power to the next destination point (*Pn*) ( $Pn = PPM * ndist$ ).
9. Calculate the remaining power (*Pr*) ( $Pr = clevel - Pn$ ). If the remaining power (*Pr*) is less than 20, then the OCU send an alarm to the admin. This alarm email contains the location information (Latitude, Longitude) where the quadcopter is landing.

(C) Notification Module:

used to send emails to the users to notify them when admin approved their locations and when the quadcopter reached the location

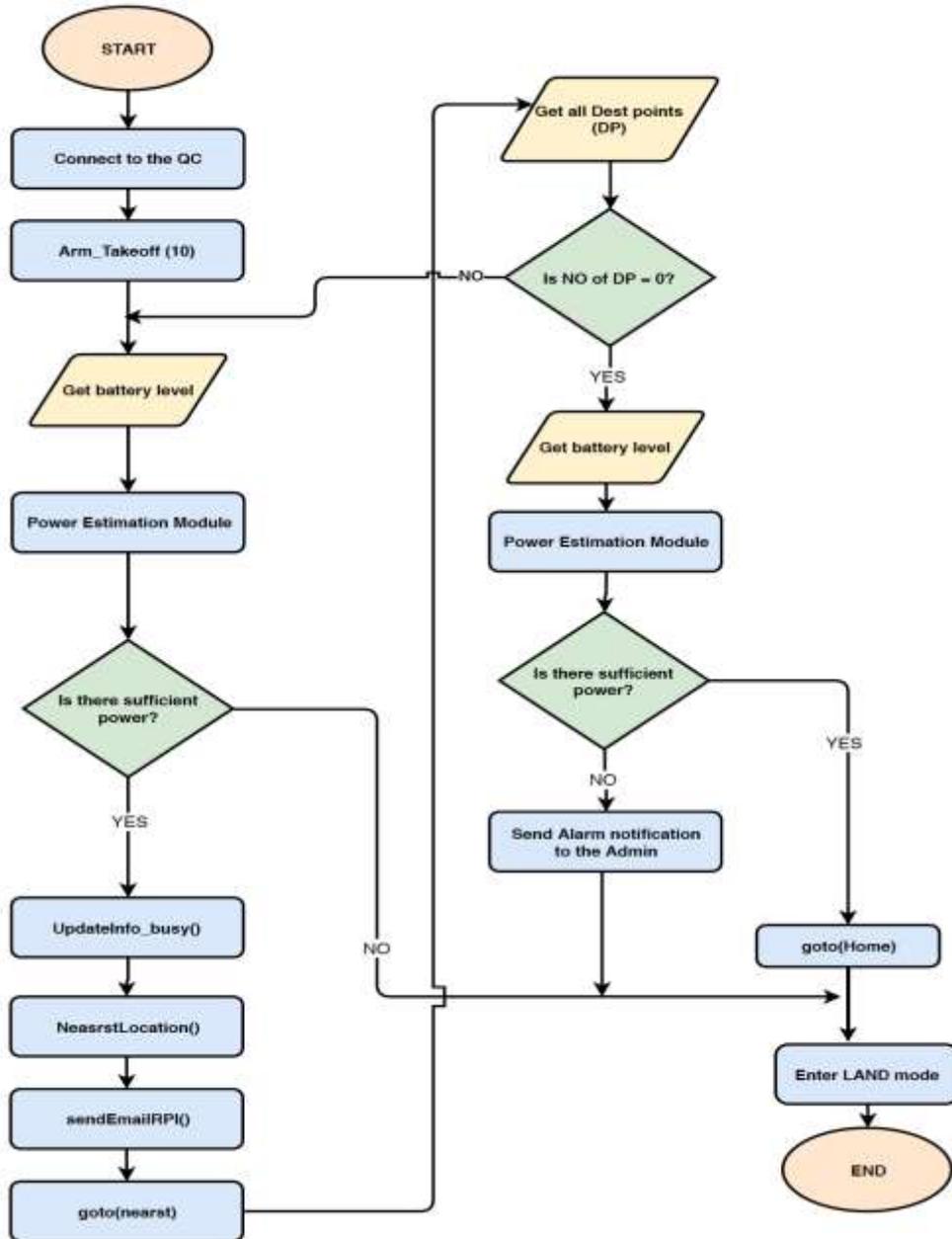


Fig. 5 Autonomous flight main algorithm.

## 5. CONCLUSIONS

In this paper, a prototype system for a drone-based delivery service has been presented. The design of the proposed system has first been discussed from the hardware point of view. Secondly, the software framework that is used to implement the autonomous flight has been discussed in details. A great effort is devoted to the simulated environment, to test and analyzed the code which is developed to the drone before deploying it on real drones. Particular attention has been paid to implement path planning when there are multiple destination points delivery. NNLF algorithm was introduced for path planning to let the quadcopter fly always to the next nearest destination point. The web interface has been developed to be easy to use by the end-user using google map. This interface has two levels of authentications. The first one is on user registration and the second one on the approval of destination points. The power estimation has been taking into consideration during the implantation of this system. Additionally, this work can be used in many services not only for delivery service. It is useful for monitoring and video transfer using autonomous flight.

## REFERENCES

- [1] Syed Omar Faruk Towaha, Clerk Maxwell “Building Smart Drones with ESP8266 and Arduino”, A Treatise on Electricity and Magnetism, BIRMINGHAM – MUMBAI, 2018.
- [2] M Muhammad, D Swarnaker, and M Arifuzzaman, “Autonomous Quadcopter for Product Home Delivery”, International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT) 2014.
- [3] Alberto Lisanti and Giorgio Venezia , “New Frontiers of Delivery Services Using Drones: a Prototype System Exploiting a Quadcopter for Autonomous Drug Shipments”, IEEE 39th Annual International Computers, Software & Applications Conference 2015.
- [4] Kenzo Nonami , “Research and Development of Drone and Roadmap to Evolution”, Autonomous Control Systems Laboratory Ltd, WBG Marive West 32F, 2-6-1 Nakase, Mihama-ku, Chiba-city, Chiba 261-7132, Japan 2018.
- [5] Eric N. Johnson and Daniel P. Schrage, “The Georgia tech unmanned aerial research vehicle: GTMax,” School of Aerospace Engineering, Georgia Institute of Technology, Atlanta 2003.
- [6] Azade Fotouhi, Ming Ding and Mahbub Hassan, “Understanding Autonomous Drone Maneuverability for Internet of Things Applications”, University of New South Wales (UNSW), Sydney, Australia 2017.
- [7] Matthias Heutger, “UNMANNED AERIAL VEHICLE IN LOGISTICS”, Marketing & Development, DHL CSI 53844 Troisdorf, Germany 2014.
- [8] Prof. Dr. Salih Mahdi Al-Qaraawi, Dr. Aymen Dawood Salman and Majida Saud Ibrahim, “Performance evaluation of Ad-hoc quadcopter monitoring system”, a thesis submitted to the computer engineering department, university of technology 2018.
- [9] Terry Kilby and Belinda Kilby, “Make: Getting Started with Drones”, Elevated Element, Printed in the United States of America 2016.

- [10] Yasir Ashraf Abd Rahman<sup>1</sup>, Mohammad Taghi Hajibeigy<sup>1</sup>, Abdulkareem Shafiq Mahdi Al-Obaidi<sup>1</sup>, and Kean, “Design and Fabrication of Small Vertical-Take- Off-Landing Unmanned Aerial Vehicle ”, MATEC Web of Conferences 152, 02023 (2018).
- [11] P. Patel, M. Patel, R. Faldu, and Y. Dave, “Quadcopter for Agricultural Surveillance”, Advance in Electronic and Electric Engineering, vol. 3, no. 4, pp. 427–432, 2013.
- [12] Cameron Roberts, “GPS Guided Autonomous Drone”, University of Evansville, College of Engineering and Computer Science, April 25, 2016.
- [13] Joshua Shlomo Berman “drone-based delivery of clinical specimens in a rural environment: a feasibility study” A dissertation submitted in fulfilment of the requirements for the degree of Master of Science in Engineering, June 2017.
- [14] Juan A. Besada, Ana M. Bernardos, Luca Bergesio, Diego Vaquero, Iván Campaña, José R. Casar, “Drones-as-a-service: A management architecture to provide mission planning, resource brokerage and operation support for fleets of drones”, Universidad Politécnica de Madrid, Spain, Information Processing and Telecommunications Center, 2019.
- [15] ArduPilot software documentation [Online]. Available: <http://ardupilot.org/about>. [Accessed 2 April 2019].
- [16] DroneKit-Python documentation [Online]. Available: <http://python.dronekit.io/1.5.0/about/overview.html> [Accessed 7 April 2019].
- [17] Heavenly Mathematics: The Forgotten Art of Spherical Trigonometry By Glen Van Brummelen.
- [18] A History of Mathematical Notations: Vol. II By Florian Cajori.
- [19] Oxford English Dictionary. Oxford University Press. 2nd ed. 1989. Cites coinage of term “Haversine” by Prof. Jas. Inman, D. D., in his Navigation and Nautical Astronomy, 3rd ed. (1835).