JJEE Volume 9 | Number 3 | September 2023 | Pages 450-465

Jordan Journal of Electrical Engineering

ISSN (print): 2409-9600, ISSN (online): 2409-9619

Homepage: jjee.ttu.edu.jo

Development of Autonomous Hexacopter UAVs for Smart City Air Quality Management

Abdel Ilah N. Alshbatat*

Department of Computer and Communications Engineering, Tafila Technical University, Tafila, Jordan
E-mail: a.alshabatat@ttu.edu.jo

Received: January 11, 2023 *Revised: February 13, 2023* *Accepted: February 18, 2023*

Abstract— Monitoring the pollution index in smart cities has piqued the interest of researchers in designing and developing unmanned aerial vehicles (UAV) capable of carrying several sensors. Recent advancements in drone technology, as well as rapid expansion in air pollution sensor technologies, have presented valuable alternatives for air quality monitoring and management in smart cities. Fixed stations are now used in smart cities to measure air pollution and collect precise data on air quality. However, such data is highly sufficient in making decisions that can improve people lives; monitoring stations require an adaptable and large communication network that is capable to handle such huge data. Instead of having such an expensive and complex network, drones could be considered an easy and cheap alternative to the current systems. In this regard, an aerial system that is equipped with off-the-shelf low-cost micro-sensors is designed and implemented to monitor air quality at a specific location inside a smart city. The behavior of the aerial system is controlled by our proposed Air Quality-Driven Control Algorithm (AQDCA). Hexacopter Drone, in particular, will fly up to a predetermined height, measure air pollutants, activate the on-board AQDCA, and then return to its ground location. The entire system was developed, implemented, and tested in a real-world flight test. The testing results corroborate the system's practicality and demonstrate that the prototype may be simply implemented to provide an added-value service to smart city citizens.

Keywords— Autonomous unmanned aerial vehicles; Smart cities; Air quality.

1. INTRODUCTION

According to a report provided by the World Health Organization [1], air pollution in low and middle-income countries has become the greatest threat to economic and human health. The data shows that nine out of ten people now breathe polluted air, which kills seven million people every year. Without a doubt, the rising levels of air pollution in such countries indicate that they are not on track to either monitor or arrest carbon emissions. From this perspective, the health effects of air pollution are serious and require an urgent and immediate effort through implementing modern technologies.

Structural health of buildings, waste management, air quality, noise monitoring, traffic congestion, city energy consumption, smart parking, and smart lighting are some of the important applications that can be directly integrated into smart city infrastructure [2-4]. Among these services, monitoring air quality requires that sensors should be deployed across the city and its data should be made available to authorities and citizens. Instead of deploying such a huge number of sensors, Drones can provide a means to monitor air quality and get a clear view of the number of pollution levels in the city and offer better services to citizens.

* Corresponding author
Article's DOI: 10.5455/jjee.204-1673429561

The Internet of Things (IoT) is one of the current solutions that network air pollution sensors. This solution needs to deploy a huge number of sensors on smart city infrastructures [5]. In addition, the sensors should be connected to microcontrollers and transceivers. The data collected from the system is then processed by a suitable protocol stack which makes it available to the nearby systems and users. Since these sensors are installed once at a dedicated location within a city, they cannot track the change of air pollution at different elevations. Therefore, the realization of the IoT still lacks the best practice due to some technical difficulties and also due to its complexity.

Improving the process of monitoring air quality necessitates the development of new technologies that combine air pollution sensors with Unmanned Air Vehicles (UAV), such as Drones. This novel technology may have several advantages over IoT systems, including the potential to measure key air contaminants with high sensitivities and temporal resolutions. In this regard, difficulties as flight endurance, city safety, and sensor data synchronization with GPS data should be addressed [6]. Path planning is another challenge that autonomous UAVs encounter. To address the path-planning problem, the A-star algorithm [7] and the genetic algorithm [8] have been devised. We addressed this issue in our study and proposed the Air Quality-Driven Control Algorithm (AQDCA). The AQDCA technique is based on the Breadth First Search (BFS) algorithm, which is appropriate for solving the problem of real-time Drones. Meanwhile, an aerial system will be conceived and built to monitor air quality autonomously and improve the quality of services provided to inhabitants. The system is made up of three modules: the Hexacopter Drone, the air quality monitoring device, and the ground station. The drone will fly to a predetermined height, measure the contaminants in the air, and then return to its original spot.

The core objective of this research is to design and develop a Drone capable of carrying multiple air pollution sensors, monitor air quality and provide a mean to manage community services. Our contribution to this research is to design an Air Quality-Driven Control Algorithm (AQDCA) that enables our aerial system to be capable of providing a higher coverage within the periods defined by the Drone flight time. The rest of this paper is organized as follows: section 2 provides an overview of current research on the technologies used to monitor air quality with drones. Section 3 provides a summary of the suggested method. System architecture, hardware design, and software design are described in section 4. This section describes in detail the proposed AQDCA for autonomously monitoring a certain urban region. Section 5 presents the results and discussions on the experiments. Section 6 presents the main conclusions.

2. LITERATURE REVIEW

Many studies have been conducted to highlight the potential applications and challenges arising from integrated drones in smart cities [9, 10]. The authors in [9] reviewed the technical and non-technical issues facing such integration. They concluded their investigation by stating that drones will bring intelligent solutions to many problems and will have a good impact on the society of smart cities. Environmental monitoring is one of these alternatives, since it allows for the early detection of dangerous compounds and lowers the costs of providing health emergency services. The author in [10] introduces a framework with heterogeneous smart UAVs to monitor several services in smart city.

Multiple environmental drones are utilized in order to independently detect and eliminate contaminants that may be present in the surrounding environment [11]. They used to measure the O_3, CO, NO2, CO_2, SO_2, NH_3, and PM. The authors used custom software that generates an Air Quality Health Index (AQHI) map of the region under inspection. The map is then used to conduct both short-term and long-term environmental analyses.

The authors in [12] proposed a system for measuring atmospheric pollutants using an Unmanned Aerial Vehicle (UAV). The data are processed in a real-time through a metaheuristic algorithm. This technique uses the method of simulated annealing to create navigational coordinates. The coordinates are then transmitted to the quadcopter's flight controller to direct its search for the source of air pollution. In the meantime, acquired data are transmitted to the ground station.

The authors in [13] proposed the use of UAV equipped with off-the-shelf sensors to perform air pollution monitoring tasks. They used the Pixhawk Autopilot for UAV control, and the Raspberry Pi for sensing and storing environmental pollution data. In this research, UAVs are guided by the proposed Pollution-driven UAV Control (PdUC) algorithm, which is based on a chemotaxis metaheuristic, a local particle swarm optimization strategy, and an adaptive spiraling technique. The PdUC algorithm also helps in obtaining a complete and detailed pollution map of the most polluted zones.

The authors in [14] designed and developed a consumer UAV-based air quality monitoring system with off-the-shelf components. They used the UAV's communication module to send the aggregated sensor data to the ground station. Their preliminary field test results show that the onboard devices did not affect the UAV's power consumption and flight time. The main problem is that the UAV operations influence the sensor readings to some extent due to the electronic interference from the UAV. To overcome this problem, they propose the use of a separate power source for the UAV and the onboard devices.

A survey on collaborative Drones and IoT was conducted in [15]. The survey indicates that there are so many attempts to show how the collaborative Drones and IoT have the capability to improve the smartness of smart cities. It also indicates that; a combination of advanced technologies could be used to perform the following tasks: data collection and dissemination, pollution monitoring, security, and surveillance. The authors also provided some challenges that need to be addressed before making such collaboration fully functional. The most pressing issues include the battery lifetime of Drones, communication protocols between the Drones and the base stations, data security, route planning, and lastly; the capability to deal with a wide variety of heterogeneous devices and sensors.

The authors in [16] presented an IoT-based 3D air quality sensing system. The architecture of the proposed system is divided into four layers: sensing layer, transmission layer, processing layer, and presentation layer. The sensing layer is used to gather data; the transmission layer is used to enable bidirectional communication between the UAV and the ground station; the processing layer is used to analyze and process the data; and the last layer is used to provide a graphic interface for the users. One problem raised in this research is the hovering time in which the UAV stays at each selected position. The influence of sensing interval against the total power consumption was verified with the power control proposed strategy. According to this study, the ideal UAV hovering time decision is around 5 s to balance the error of air sampling and the number of sensing positions.

The characteristics of UAV's and the types of air quality monitoring sensors were reviewed in [17, 18]. It was observed that, different types of UAVs are used such as quadrotor and hexacopter. Meanwhile, air quality monitoring sensors to sense parameters like CO, SO2, NO2, O3, PM2.5, and PM1.0 are also used. The method to collect data and the type of sensors were also presented in [19]. The others show different examples of sensors and their characteristics. The measured concentrations were limited with a precision of 1 mg/m^3 and vary according to the type of emission and the required response time.

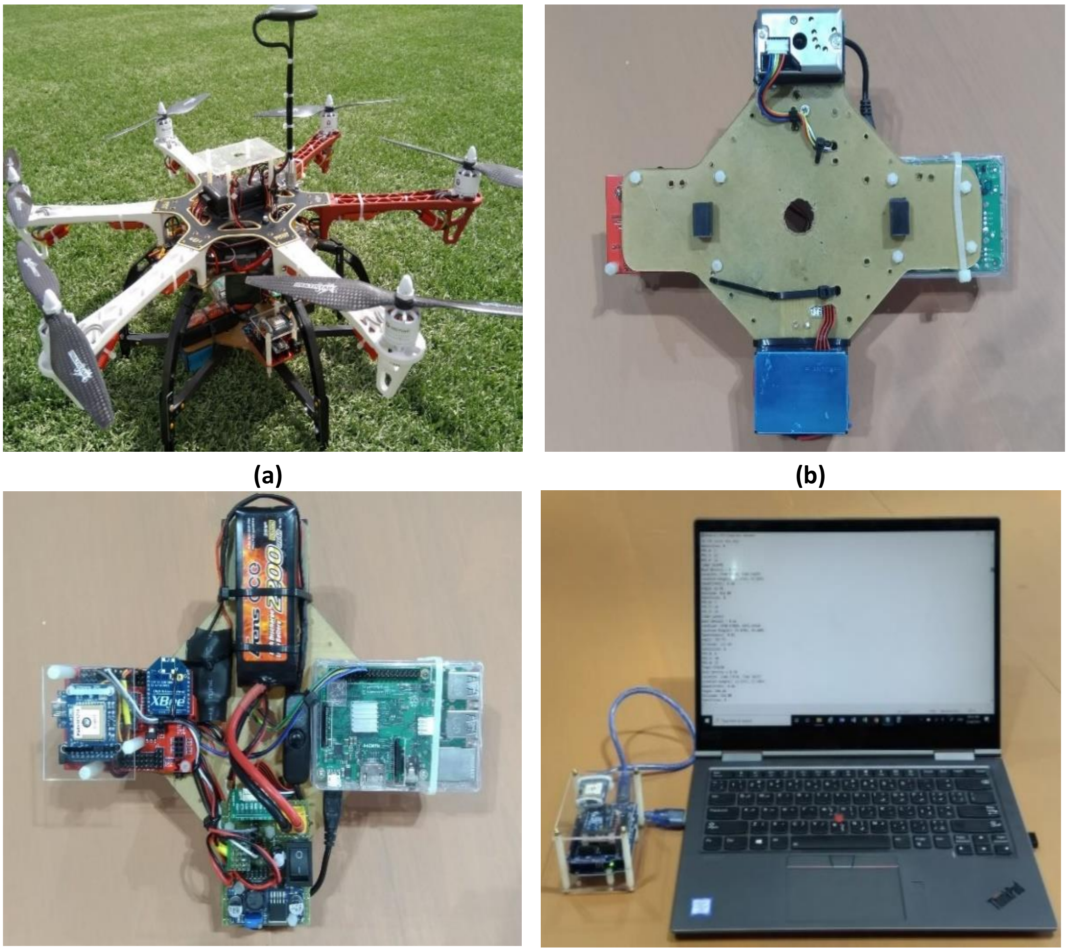
The effects of speed and altitude on the air pollution measurements using Hexacopter Drone were conducted in [20]. Several challenges were investigated in this research, in particular, the effect of wind-generated from UAVs propellers rotation on the efficiency of gas sensors. The authors studied the effect of changing altitude and speed on the measurements and presented their system. As stated there, the system can measure gas concentrations at speeds less than or equal to 6 m/s. On the other hand, speed of above 8 m/s has significant impact on the accuracy of the measurements while sensing the targeted gases.

The author in [21] conducted several experiments aiming to identify the best location for the gas sensor. It was noted that, the active gas transport approach was most effective at reducing the propeller dilution effect. The author in [22] presented a UAV system and conducted two tests: the aim of the first test was to solve the sensor mounting point issue to avoid turbulence and air mixing effect of the propellers, while the aim of the second test was to quantify the propeller downwash effect on the sensor readings. The results of the first test showed that the best mounting point for the sensors to be alongside the UAV, while the results of the second test demonstrated that the UAV propellers cause a dispersion effect shown by the decrease of gas and PN concentration measured in real time. In this research, we propose the use of an autonomous Hexacopter Drone carrying two types of air quality sensors. The holder of the sensors was designed to allow precise measurements.

3. OVERVIEW OF THE PROPOSED SYSTEM

As shown in Fig. 1, the proposed system consists mainly of three major parts: i) Hexacopter Drone, ii) an air quality monitoring system and iii) and ground station. Hexacopter Drone was built on the F550 frame as shown in Fig. 1(a). The Drone is equipped with a landing gear to hold the air quality monitoring system. Figs. 1(b) and (c) show the air quality monitoring system. All sensors and modules including the GPS, and Xbee were connected to the Arduino board. Communication between the air quality monitoring system and Drone is achieved through a Raspberry Pi. Fig. 1(d) shows the ground station, which consists mainly of a laptop and a router. The station provides both live data from the sensors and GPS data. Fig. 2 shows the functional block diagram of the aerial system, where Pixhawk flight controller is responsible for flight stabilization and Raspberry Pi is in charge of the guidance system. Both modules are connected through a serial port.

As soon as the Drone is off the ground, data is collected and saved to external storage, meanwhile, data is processed within the Raspberry Pi, and also the aggregated part is sent to the ground station via the Xbee module. The aggregated data includes the sensor data and the UAV data. Initially, the GPS will guide the Drone to a predefined waypoint, and then the flight controller will receive its command from the Raspberry Pi. Raspberry Pi will implement the on-board AQDCA and process the data obtained from the air pollutant sensors. The processed data will be used to control the behaviour of the aerial system. Communication between Raspberry Pi and Pixhawk flight controller will be accomplished using the MAVLink protocol over a serial connection. Once the last predefined waypoint is reached, Raspberry Pi will command the Drone to fly back down to its original location on the ground.



(a) (b)

(c) (d)

Fig. 1. Architecture of the proposed system: a) hexacopter drone with air quality monitoring system payload; b) air quality monitoring system-back side; c) air quality monitoring system-front side; d) ground station.

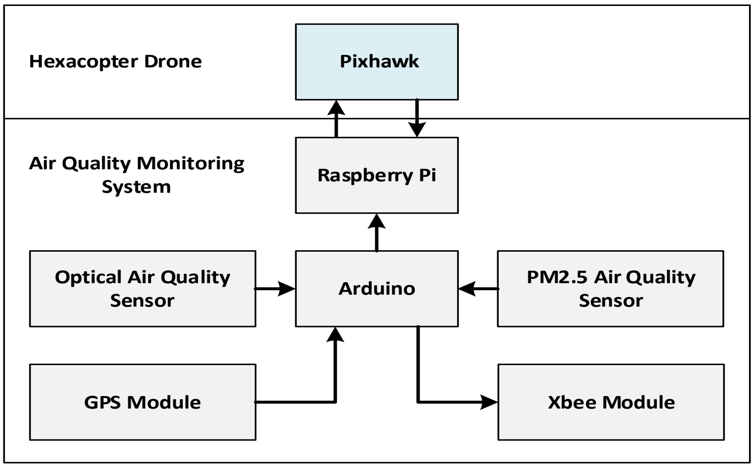


Fig. 2. Functional block diagram of the aerial system.

4. SYSTEM ARCHITECTURE

In the following subsections, we describe the proposed system as well as the hardware and software components. The hardware was constructed with off-the-shelf parts, while the software was built specifically for this project. In terms of the software, we have developed an algorithm for autonomous navigation called Air Quality-Driven Control Algorithm (AQDCA).

4.1. Architecture of the Hardware

4.1.1. Hexacopter Drone

As shown in Fig. 1(a), Hexacopter Drone was built on the F550 frame [23]. F550 is a light weight Hexacopter frame and has a sufficient load space for carrying larger payloads such as camera systems and other needed electronic components. The space is used to hold the air quality monitoring system. The frame is equipped with the following off-the-shelf components: Pixhawk flight controller, six electronic speed controllers (ESC), six BLDC motors, six (9 X 45) inch propellers, RC receiver/ transmitter, PPM encoder, buzzer, inertial measurement unit (IMU), safety switch, telemetry radio, GPS module, and LiPo battery.

The Hexacopter Drone is simply used as a carrying platform and driven using a Pixhawk flight controller. Pixhawk is an advanced autopilot that features the most advanced processor technology. It has a set of sensors and these include gyroscopes, accelerometers, and barometer. The autopilot is attached to six Turnigy Plush 30 amp Electronic Speed Controllers, and six brushless three-phase Tiger motors (MN2214) rated for 920KV / 251W. The GPS and compass modules were mounted separately and attached to the autopilot. The whole system is powered by a Turnigy Nano-Tech 6400mAh 3S 30C LiPo rechargeable battery to support a maximum flight time of around 25 minutes.

4.1.2. Air Quality Monitoring System

As shown in Fig. 3, air quality monitoring system consists of Arduino nano microcontroller, Xbgee module, GPS module, PM2.5 air quality sensor, optical air quality sensor, Raspberry Pi 3 B+, Lipo rechargeable battery (2s) 25C 2200mah, and a regulator (12-5 V). All sensors including the GPS and Xbee modules were connected to the Arduino nano microcontroller. The Raspberry Pi is also connected to the Arduino Nano and to the Pixhawk flight controller through a serial port. Since the Drone consumes a very large current than the air quality monitoring system, the assembly was powered by a separate rechargeable Lipo battery (2200mah). Fig. 1(a) shows the system where it mounted between the landing gear.

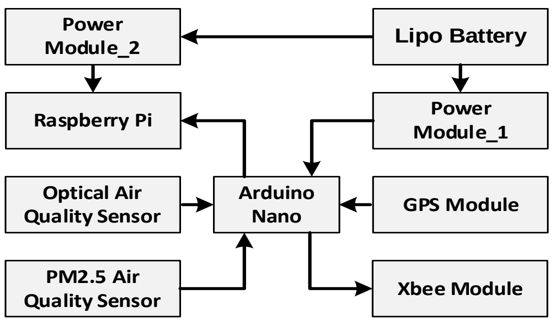
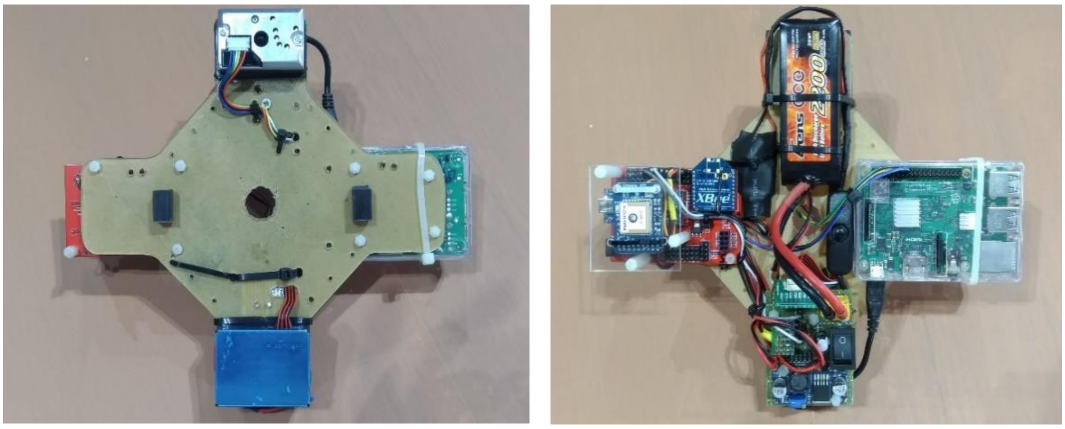


Fig. 3. Functional block diagram of the air monitoring system with the proposed Arduino nano microcontroller.

Fig. 4 shows the working prototype of the air quality monitoring system. The sensors, shown in Fig. 4(a), are both low weight and suitable to be carried on a Drone. Both sensors collect the concentration of pollutants and send the data to the Arduino Nano microcontroller. The data from the GPS is also sent to the Arduino Nano microcontroller. The aggregated data is then passed to the Raspberry Pi and to Xbee module. The Raspberry Pi implements the AQDCA and process the aggregated data obtained from the Arduino Nano microcontroller. The results will be used to control the aerial system motion. Meanwhile, the aggregated data were also sent to the ground station via the Xbee module in a real time manner for present and long-term environmental analysis.

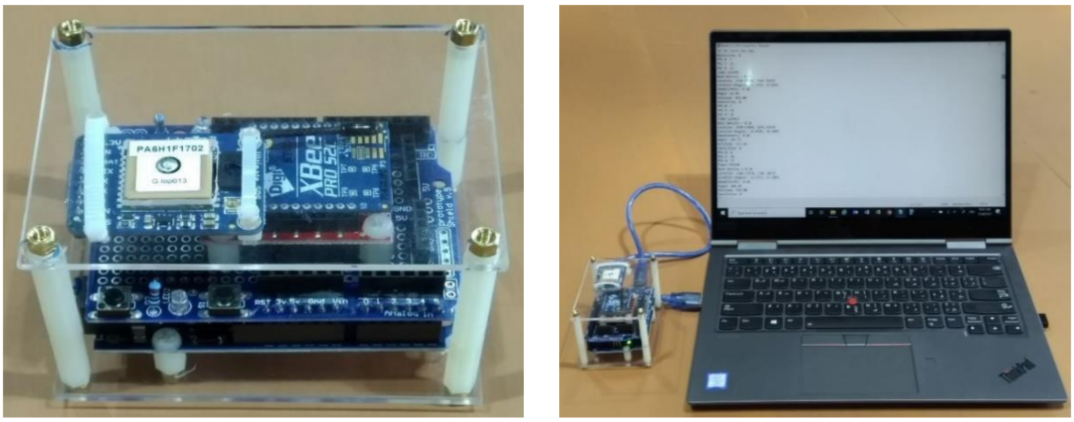


(a) **(b)**

Fig. 4. Air quality monitoring system: a) back side; b) front side.

4.1.3. Ground Station

As shown in Fig. 5, ground station consists of Arduino Uno microcontroller, Xbgee module, GPS module, and a laptop. The primary function of the ground station is to receive the aggregated data and then transfer it along with the GPS data to a laptop. The laptop is then analyzing the data and makes it available to authorities.



(a) **(b)**

Fig. 5. Ground station: a) ground station module; b) ground station attached to the laptop.

4.2. Architecture of the Software and Control Strategy

In this research we have developed an algorithm called Air Quality-Driven Control Algorithm (AQDCA). The AQDCA is partially based on the Breadth First Search (BFS) algorithm and is used for controlling the behavior of the drone to search an area with highest pollution concentration levels. BFS is an initialization phase that is performed by the system before processing the AQDCA algorithm. As depicted in Fig. 6(c), the entire targeted region is modeled as an undirected graph G(V, E), where a list of vertices $V = [\, r_0, \dots, r_k \,]$ represents all roundabouts in a smart city and a list of edges $E = [\, s_{00}, \dots, s_{xy}]$ represents all streets linking them. To ease the process, the entire region has been separated into Sector 1 and Sector 2 as shown in Figs. 6(a) and (b). The location of roundabouts and street lengths in each section are known beforehand.

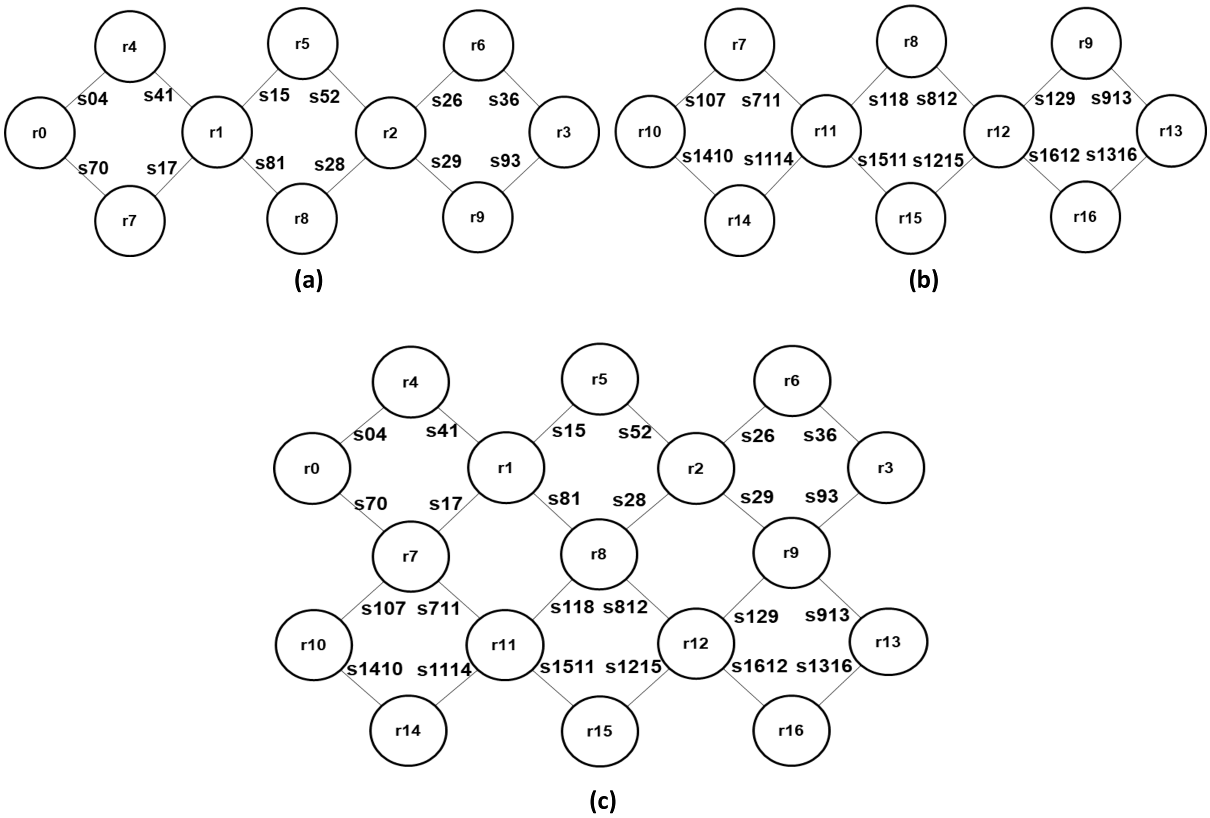


Fig. 6. Undirected graph $G(V, E)$: a) vertices and edges, Sector 1; b) vertices and edges, Sector 2; c) vertices and edges (targeted area).

According to the BFS algorithm, the graph is traversed from a given starting vertex and then the neighbors of this vertex are visited first before moving to those that are two edges away [24]. Algorithm 1 shows the pseudo-code of the BFS and works as follow: First in First Out (FIFO) queue is used to store the vertices that need to be visited, and a visited list is used to store the vertices that have already been visited. With the current case, the entire targeted area is divided into sectors, where one of the $[\, \mathrm{r}_0, \mathrm{r}_{10}, \mathrm{r}_3, \mathrm{r}_{13} \,]$ is assigned as a starting vertex, and inserted into the queue. Then it is extracted from the queue and inserted in the visited list. Adjacent vertices are then inserted into the queue and marked as visited. This process is repeated until a target vertex (ry) is reached. BFS algorithm then returns the shortest path between a start vertex (rx) and an end vertex (ry). The shortest route is the link that the Drone should initially follow to reach its objective. This link has been updated to reflect AQDCA.

Algorithm 1. Optimal flight path using BFS

Input: A graph G (V, E) for the targeted area, start (r_x) vertex, and end (r_y) vertex.

Output: Flight path F[i] from the starting vertex to the end vertex.

```
BFS (G, r_x, r_y)        // r_x and r_y could be one of the [ r_0, r_10, r_3 r_13 ].
    let q be a queue.
    let v be a list of visited vertices
    q.enqueue( r_x )
    mark r_x as visited
    while queue. Length > 0 do
        v = q.dequeue( )
        if v is r_y then
            return v
        for all neighbours u of v in Graph G
            if u is not visited
                q.enqueue( u )
                mark u as visited
            end if
        end for
    end while
```

The AQDCA is composed of two phases that are different from the BFS phases: visiting and exploration phases. In visiting phase, Drone flies up to a predetermined height above start vertex (rx), measures air pollutants and puts it in a buffer. At this stage, Drone is required to visit another vertex that is reported by BFS algorithm, measures air pollutants and puts it also in its buffer. The exploration phase is based mainly on the variation in samples, if the sampling variation between any two vertices is negative, then Drone follows the BFS algorithm and the next vertex reported by the flight path F[i] will be visited. Otherwise, if the sampling variation is positive, then all vertices adjacent to the original should be visited.

The AQDCA pseudo-code is shown in Algorithm 2. Because the position of the Drone and all vertices, including a start vertex (rx) and an end vertex (ry), are known at the start, the method performs the BFS and calculates the optimal flight path F[i]. The desired flight command is then calculated, and the Drone will fly to a predefined height F[0]. Two stages will be undertaken for each vertex (v) near to the F[0]: visiting and exploring. Two positions must be visited during the visiting phase. Drone requests current sensor information and assesses the concentrations of air contaminants from F[0]. Then, from F[1], the drone requests sensor data and measures the concentrations of air contaminants. The desired flight command is computed based on the difference between the readings, and the Drone will fly either to an adjacent vertex that is not listed in the flight path or to an adjacent vertex that is listed in the flight path. Meanwhile, sensor data is collected and relayed to the ground station.

The mobility models used in this research is based on the AQDCA, where the Drone is set to start at a random vertex $[\, r_0, r_{10}, r_3\ r_{13}\,]$ within the targeted area and reacts to air pollution values. Drone is restricted to follow the programmed edges $E = [\, s_{00}, \dots, s_{xy}]$, in which the movement of the drone between any two vertices is divided into steps, for each step, it collects a sample and compares it with the previous one. Based on the results, Drone adjusts its heading. After the whole area is covered, Raspberry Pi will command the Drone to return to the point where it was lunched. Fig. 7 shows the closed-loop control scheme of the

Drone, where the Pixhawk flight controller is responsible for maintaining a self-stabilizing flight and the AQDCA is in charge of the guidance system.

Algorithm 2. Air Quality-Driven Control Algorithm (AQDCA)

Input: Flight path F[i] from algorithm 1, edges S[i].
Output: Air pollution concentration pMax[i].

```
procedure AQDCA (drone, F[i], S[i])
visiting= true
exploration= false
i=0
pMax=[ ]
home= Drone.getCurrentPosition()
Drone.flyTo (F[0])
for each vertex v adjacent to F[i] do
        while visiting is true do
                P1=Drone.getCurrentSensorReading()
                Drone.flyTo(F[i+1])   // Heading as recorded by edges S[i]
                P2=Drone.getCurrentSensorReading()
                visiting=false
                exploration= true
        end while
        while exploration is true do
                Δp=p2-p1
                If Δp>0
                        Drone.flyTo(F[i]+1)   // (F[i]+1) adjacent but not listed in flight path
                        pMax[i]= p2
                else
                        Drone.flyTo(F[i+1])    // (F[i+1]) adjacent and listed in flight path
                        i=i+1
                end if
                visiting= true
                exploration= false
        end while
end for
return pMax
Drone.flyTo (home)
end procedure
```

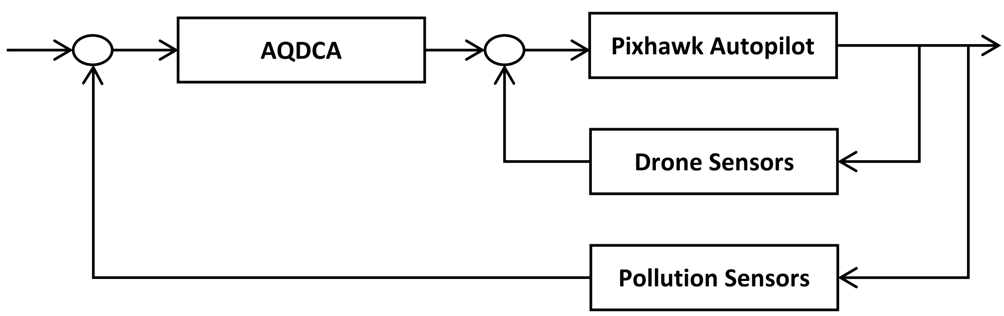


Fig. 7. Closed-control loop system of the drone.

Finally, Raspberry Pi module is used to implement the Air Quality-Driven Control Algorithm (AQDCA) and command the Drone using DroneKit-Python API accordingly. In addition, three software packages are used in this paper: autopilot software, ground station software, and air quality monitoring software. ArduPilot software is used in this paper to monitor the drone, reading sensors inputs and control the drone accordingly. Air quality monitoring software as well as ground station software which is differ from mission planner software were implemented using Arduino platform.

5. EXPERIMENTAL RESULTS AND ANALYSIS

In this research, several experiments were conducted to demonstrate the system's ability to perform automated aerial pollution detection. The sensors used in this research are PM2.5 air quality sensor (PMS5003), and optical air quality dust sensor (GP2Y1010AU0F). PM2.5 air quality sensor allows the measurements of particulate materials of 1, 2.5, and 10 µm with a serial protocol communication access. Optical air quality dust sensor outputs an analog voltage proportional to the measured dust density, with a sensitivity of 0.5 V/ (0.1 mg/m^3). Table 1 shows more details about the sensors.

Table 1. Specifications of the air quality sensors.

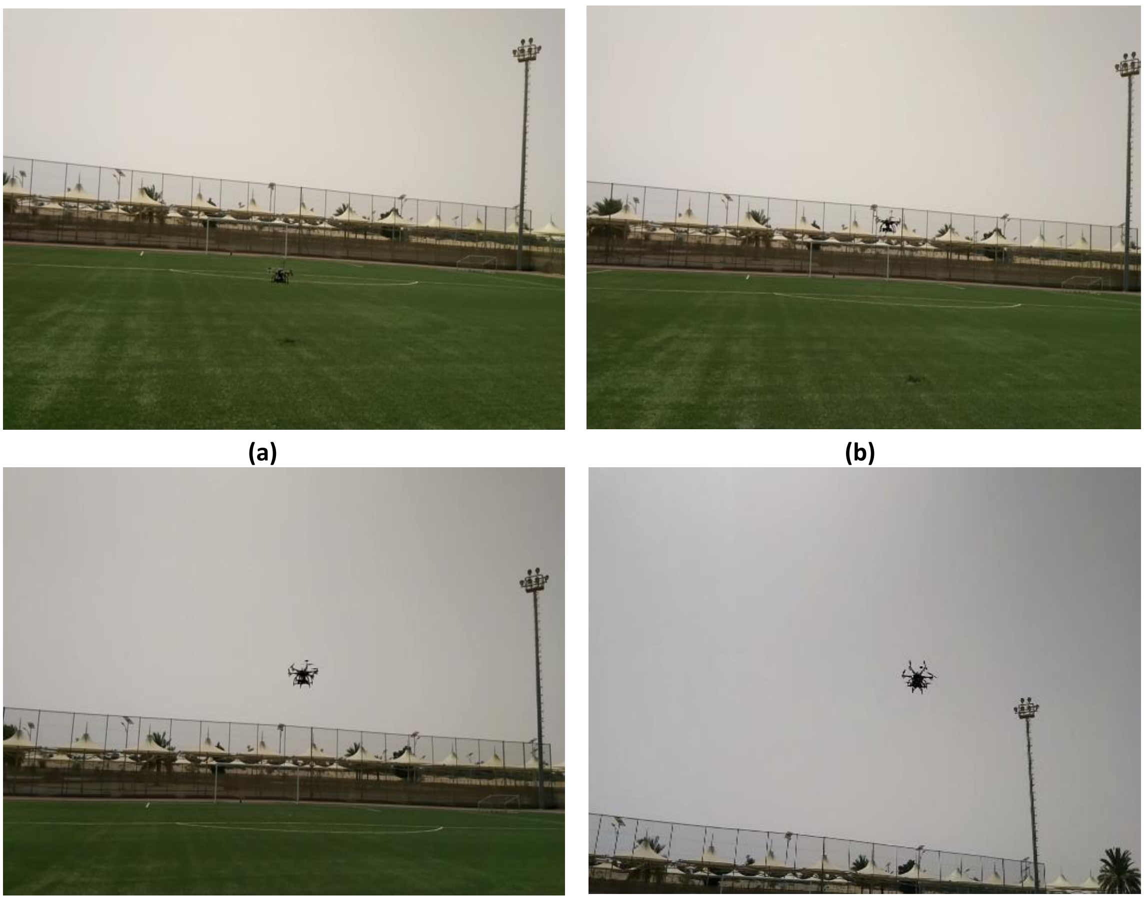
Parameter	PMS5003	GP2Y1010AU0F
DC power supply	4.5 V to 5.5 V	2.5 V to 5.5 V
Operating current	≤ 100 mA	20 mA
Resolution	1 µg/m^3	100 µg/m^3
Net weight	42 g	16 g
Communication protocol	UART	Analog
Working temperature range	-10 °C to 60 °C	-10 °C to 65 °C
Dimensions	50.0 × 38.0 × 21.0 mm	46.0 × 30.0 × 18.0 mm

Initially, to validate the functionality of the onboard sensors; two experiments were conducted to read data from the GP2Y1010AU0F and PMS5003 sensors. In the first experiment, two sensors of the same model (GP2Y1010AU0F) are used. The sensors are attached to two microcontrollers and placed close to a pollutant source. Measurements were recorded and cross-checked to ensure the validity of data from both sensors. Due to the lack of a standard calibration protocol, a linear calibration coefficient suggested by the manufacturer is then modified and applied to both sensors. In addition, microcontroller is programmed to remove the abnormal values and average the data every two seconds. In the second experiment, the other two sensors were used. The same procedure was applied and the measurements were recorded and cross-checked.

Taking into account the problem related to the effect of electronic interference from Drone, air quality monitoring system was powered by a separate rechargeable Lipo battery. In addition, to mitigate effect of the wind generated by the propellers, the system was attached to the bottom of the fuselage. Air inlet of the sensors was made parallel to the head of the Drone.

Drone was powered by a 6400 mAh 3S LiPo rechargeable battery with a continuous discharging rate of 30C. The LiPo battery is connected to a power module that provides a regulated power supply to the electronic parts on the drone and also supports measuring the battery voltage and current consumption. The autopilot is programmed to manage the consumed power. As soon as the level of the battery is below a predetermined threshold, the autopilot commands the Drone to land on the ground. The total current drawn from the battery is 12.1 A. Thus, the onboard battery will provide a flight time of approximately 25.4 minutes. As mentioned previously. The air quality monitoring system was powered by a 2200mah 2s LiPo rechargeable battery. The output of the battery is connected to a power management unit. The unit regulates the power input for all components, including the GPS module, the Xbee module, the PMS5003 and GP2Y1010AU0F sensors, the Arduino Nano, and the Raspberry Pi.

As shown in Fig. 8, four types of experiments were conducted to collect 10-minute preliminary sensor data and then evaluate the effect of implementing Air Quality-Driven Control Algorithm (AQDCA) on Drone's location. Due to battery restrictions, the experiments were conducted on a 5000 m² coverage area. The area was divided into two sectors and the distance between consecutive vertices was programmed to be 20 m. The position of all vertices including both a start vertex (rx) and an end vertex (ry) were loaded to the Raspberry Pi. The flight test was carried out at heights of 3, 5, 10, and 15 m.



(a) (b) (c) (d)

Fig. 8. Photos obtained during practical tests at height of: a) 3 m; b) 5 m; c) 10 m; d) 15 m.

Fig. 8(a) illustrates the first test at a 3 m height. Drone is hovering for 6 s, then the data is collected and averaged every two seconds. Figs. 9(a) and (b) show the measures obtained using the PMS5003 and GP2Y1010AU0F sensors. The PM10, PM2.5, and PM1 concentrations were about 36, 20, and 11 µg m^{-3}, respectively. The dust density was about 0.17 mg/m^3. It was observed that the ratio between PM10 and PM2.5 is around 1.8. Fig. 8(b) illustrates the second test at a 5 m height. The PM10, PM2.5, and PM1 concentrations were about 37, 20, and 10 µg m^{-3} , respectively. The dust density was about 0.16 mg/m^3. The results obtained from the second test were relatively close to the results obtained from the first test. This led us to infer that the quantitative measurements for the three types of particulate materials are almost the same up to 5 m.

Fig. 8(c) illustrates the third test at a height of 10 m. Figs. 10(a) and (b) show the measures obtained with the PMS5003 and GP2Y1010AU0F sensors. The PM10, PM2.5, and PM1 concentrations were about 47, 22, and 12 µg m^{-3}, respectively. The dust density was about 0.16 mg/m^3. The test indicates that the concentration of PM10 increases eleven points from the first test. Fig. 8(d) illustrates the fourth test at a height of 15 m. Figs. 11(a) and (b) show the measures obtained with the PMS5003 and GP2Y1010AU0F sensors. The PM10, PM2.5, and PM1 concentrations were about 38, 19, 9 µg m^{-3}, respectively. The dust density was about 0.16 mg/m^3. The results obtained from this test indicate that the concentrations of the particulate materials decrease as the elevation increases.

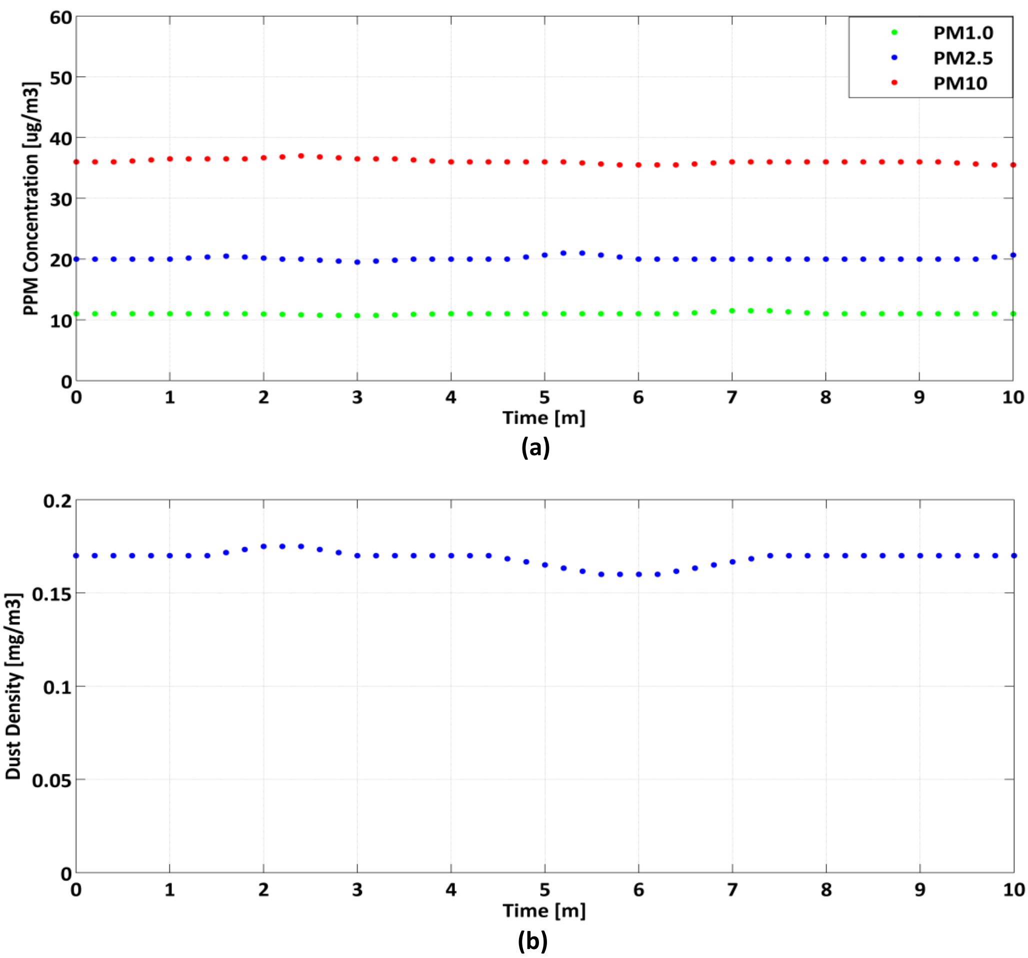
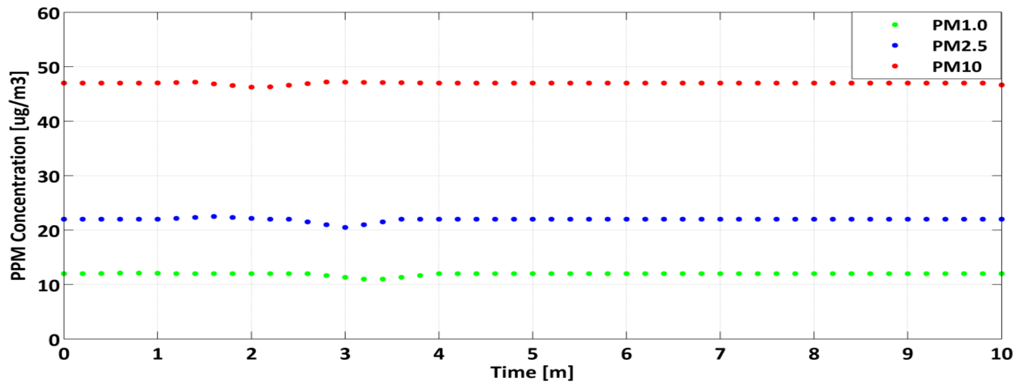
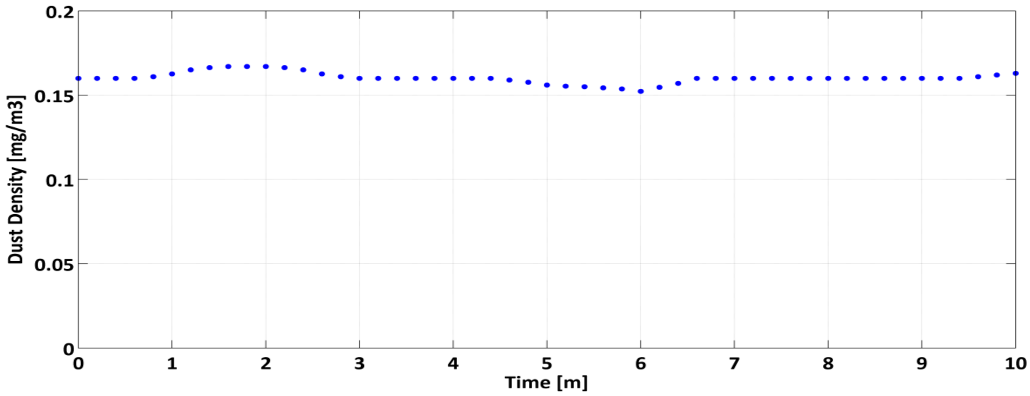


Fig. 9. Measurements obtained by PM2.5 air quality sensor, and optical air quality dust sensor at 3 m elevation: a) PM measurements during the first test; b) dust measurements during the first test.

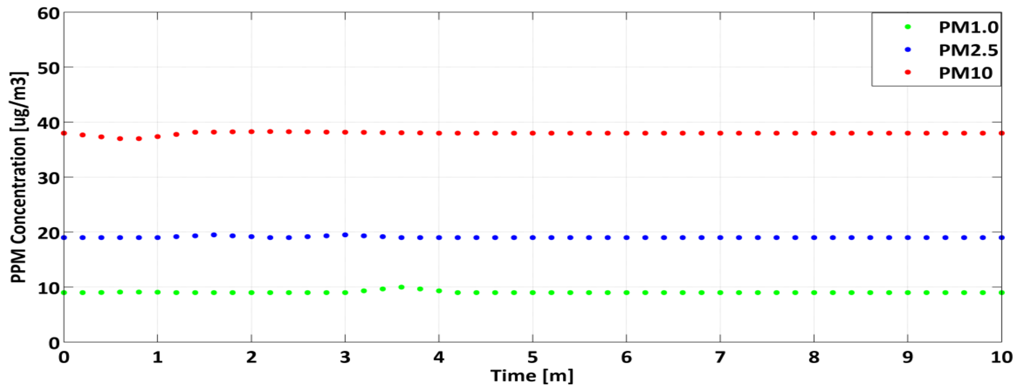


(a)

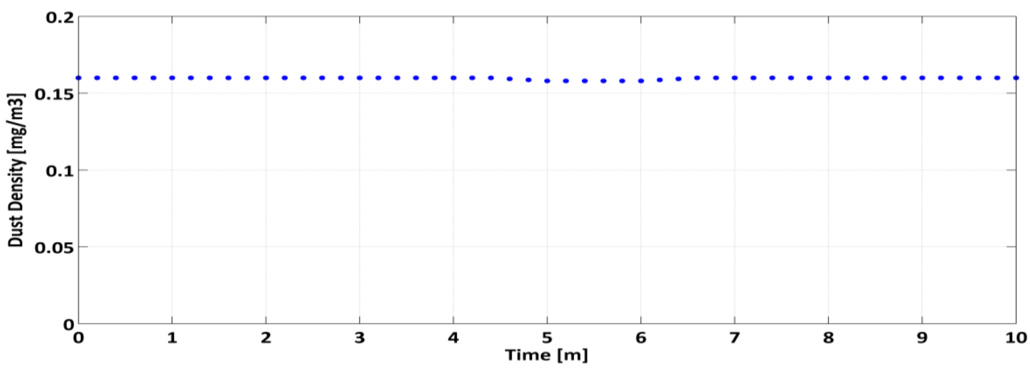


(b)

Fig. 10. Measurements obtained by PM2.5 air quality sensor, and optical air quality dust sensor at 10 m elevation: a) PM measurements during the third test; b) dust measurements during the third test.



(a)



(b)

Fig. 11. Measurements obtained by PM2.5 air quality sensor, and optical air quality dust sensor at 15 m elevation: a) PM measurements during the fourth test; b) dust measurements during the fourth test.

From the experiments, it can be observed that the PM10 concentrations varied between 36 and 47 µg m^{-3}, which were detected between two distances; 5 and 10 m. Therefore, Drone was programmed to fly up to a height of 10 m, switch to position hold, and then handover the control to the AQDCA and react to the PM10 measurements obtained with the PM5003 sensor. This is essential to assess the impact of implementing Air Quality-Driven Control Algorithm (AQDCA) on Drone's location. As expected, the path followed by Drone indicates that it starts a visiting and then an exploration phases throughout the scenario until it locates a position with the highest degree of pollution. After the whole area was covered, Raspberry Pi commanded the Drone to return to the point where it was lunched.

6. CONCLUSIONS AND FUTURE WORK

In this paper, an aerial system was built to autonomously monitor air quality, identify and measure air pollution concentrations within smart cities. The system is made up of three components: a Hexacopter Drone, an air quality monitoring device, and a ground station. The Hexacopter Drone is equipped with commercially available air pollution sensors and is piloted by a Pixhawk Autopilot. The aggregated data is processed using the Raspberry Pi module, and the Air Quality-Driven Control Algorithm (AQDCA) is implemented. The Arduino platform is used to collect and store data on environmental degradation. The entire targeted area is modeled using a network of vertices and edges based on the suggested AQDCA method. In a smart city, vertices represent roundabouts and edges represent streets. Drone flies to a predetermined height, measures air contaminants, and then returns to its ground location. In particular, the Raspberry Pi first directs the drone to rise to a specific altitude, after which the AQDCA will direct the drone to the area with the highest pollution levels. The geo-location data is then combined with the collected data and sent directly to the ground station for analysis of the immediate and long-term environmental conditions. The entire system has been developed, put into practicee, and examined in a real flight test. Future work will involve integrating ground stations with IoT systems and running additional experiments to fully comprehend how propeller turbulence affects sensor readings.

REFERENCES

[1] World Health Organization, 9 out of 10 people worldwide breathe polluted air, but more countries are taking action. <https://www.who.int/news-room/detail/02-05-2018-9-out-of-10-people-worldwide-breathe-polluted-air-but-more-countries-are-taking-action, March 7, 2020>

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, "Internet of Things for smart cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22-32, 2014.

[3] M. Mahmoud, "Indoor-lighting system design using simultaneous control of LEDs lighting intensity and roller blinds' opening for economic energy consumption," *Jordan Journal of Electrical Engineering*, vol. 6, no. 3, pp. 179-203, 2020.

[4] A. Daniel, H. Jacson, K. Adnan, "An open source IoT garage real time controller (GarageRTC)," *Jordan Journal of Electrical Engineering*, vol. 6, no. 3, pp. 179-203, 2020.

[5] B. Braem, S. Latre, P. Leroux, P. Demeester, T. Coenen, P. Ballon, "Designing a smart city playground: real-time air quality measurements and visualization in the city of things testbed," *In Proceedings of the 2016 IEEE International Smart Cities Conference*, Trento, Italy, pp. 773–774, 2016.

[6] G. Mayuga, C. Favila, C. Oppus, E. Macatulad, L. Lim, "Airborne particulate matter monitoring using UAVs for smart cities and urban areas," *in TENCON 2018-2018 IEEE Region 10 Conference*, pp. 1398–1402, 2018.

[7] S. Malaek, A. Kosari, "Novel minimum time trajectory planning in terrain following flights," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 1, pp. 2–12, 2007.

[8] K. Ashenayi, R. Wainwright, "Genetic algorithms for autonomous robot navigation," *IEEE Instrumentation and Measurement Magazine,* vol. 10, no. 6, pp. 26–31, 2007.

[9] N. Mohamed, J. Al-Jaroodi, I. Jawhar, A. Idries, F. Mohammed, "Unmanned aerial vehicles applications in future smart cities," *Technological Forecasting and Social Change*, vol. 153, pp. 119293, 2020.

[10] H. Kim, L. Mokdad, J. Ben-Othman, "Designing UAV surveillance frameworks for smart city and extensive ocean with differential perspectives," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 98-104, 2018.

[11] G. Rohi, O. Ejofodomi, G. Ofualagba, "Autonomous monitoring, analysis, and countering of air pollution using environmental drones," *Heliyon*, vol. 6, no. 1, pp. e03252, 2020.

[12] N. Yungaicela-Naula, L. Garza-Castanon, A. Mendoza-dom, L. Minchala-avila, "Design and implementation of an UAV-based platform for air pollution monitoring and source identification," *Congreso Nacional de Control Automatico*, Mexico, pp. 288-293, 2017.

[13] A. Oscar, Z. Nicola, N. Enrico, T. Carlos, "Using UAV-based systems to monitor air pollution in areas with poor accessibility," *Journal of Advanced Transportation*, *Hindawi Publishing Corporation*, pp. 1-14, 2017.

[14] Q. Gu, C. Jia, "A consumer UAV-based air quality monitoring system for smart cities," *2019 IEEE International Conference on Consumer Electronics*, pp. 1-6, 2019.

[15] S. Alsamhi, O. Ma, M. Ansari, F. Almalki, "Survey on collaborative smart drones and Internet of Things for improving smartness of smart cities," *IEEE Access*, vol. 7, pp. 128125-128152, 2019.

[16] Z. Hu, Z. Bai, Y. Yang, Z. Zheng, K. Bian, L. Song, "UAV aided aerial-ground IOT for air quality sensing in smart city: Architecture, technologies, and implementation," *IEEE Network*, vol. 33, no. 2, pp. 14–22, 2019.

[17] V. Lambey, A. Prasad, "A review on air quality measurement using an unmanned aerial vehicle," *Water Air Soil Pollution*, vol. 232, pp. 1-32, 2021

[18] J. Brady, M. Stokes, J. Bonnardel, T. Bertram, "Characterization of a quadrotor unmanned aircraft system for aerosol particle concentration measurements," *Environmental Science and Technology*, vol. 50, no. 3, pp. 1376–1383, 2016.

[19] M. Alvarado, F. Gonzalez, A. Fletcher, A. Doshi, "Towards the development of a low-cost airborne sensing system to monitor dust particles after blasting at open-pit mine sites," *Sensors*, vol. 15, pp. 19667-19687, 2015.

[20] R. Noori, D. Dahnil, "The effects of speed and altitude on wireless air pollution measurements using hexacopter drone," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, pp. 268-276, 2020.

[21] P. Neumann, *Gas Source Localization and Gas Distribution Mapping with a Micro-Drone*, Ph.D. Thesis, Freie Universität Berlin, Berlin, Germany, 2013.

[22] T. Villa, F. Salimi, K. Morton, L. Morawska, F. Gonzalez, "Development and validation of a UAV based system for air pollution measurements," *Sensors*, vol. 16, no. 12, pp. 2202, 2016.

[23] A. Alshbatat, "Adaptive vision-based system for landing an autonomous hexacopter drone on a specific landing platform," *International Journal of Intelligent Systems Technologies and Applications*, vol. 20, no. 3, pp. 245–270, 2021.

[24] T. Cormen, C. Leiserson, R. Rivest, C. Stein, *Introduction to Algorithms*, MIT Press, Cambridge, London, 3rd edition, 2009.