e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14

Time and Energy Minimized Trajectories for LANs of Drones

Sandaruvan Rajasinghege[1] and Rohan de Silva[2]
[1]Research Student, School of Engineering and Technology, CQ University, AUSTRALIA
[2]Lecturer, School of Engineering and Technology, CQ University, AUSTRALIA

[1]Correspondence Author: s.c.rajasinghege@cqumail.com

ABSTRACT

Controlling UAV movements in a UAV network is a critical but not well-studied research area in UAV network research. In this paper, we consider the problem of finding time and energy minimized trajectories for LANs of Drones (LoDs) by computationally inexpensive method. A LoD is a novel type of UAV network, which uses a minimum number of UAVs to perform any collaborative task. For both criterions of time and energy minimization, we formulate separate nonlinear constrained optimization problems and use Sequential Quadratic Programming method to obtain local optimum solutions. These minimization methods were tested by carrying out a range of simulations in MATLAB environment.

***Keywords*—** LANs of Drones, Time Minimization, Energy Minimization, Determination of Desired Trajectory, Communication Paths

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs), which are commonly known as drones are increasingly gaining its popularity among commercial and civilian applications. As the general public is more and more interested in UAVs, the UAV manufacturing companies develop more reliable, user friendly and affordable UAVs to an increasingly competitive market. Within the last few decades, commercial and civil UAV market has grown exponentially and many privately owned companies have started offering UAV related services. These services include aerial photography, asset inspection, surveying, 3D mapping and thermal imaging. [1-4]. Several major companies are operating UAV based parcel delivery services. DHL Parcel copter [5] and Amazon Prime Air [6] are such ongoing developments.

Even though, the commercial and civilian drone industry has grown to such a vast extent, there are still no UAVs developed to perform tasks cooperatively. For all commercial and civilian applications, the developers still use individually controlled or standalone UAVs. A standalone UAV is a UAV which is controlled via an RC transmitter, smart phone, computer or any other ground station. Therefore, if there are multiple UAVs operating in the same area, they are controlled by individual controllers and there is no communication between the UAVs.

Several research groups have proposed Flying Ad hoc Networks (FANETs) [7], Internet of Drones (IoD) [8] and LANs of Drones (LoDs)[9] for practical applications.

Flying Ad hoc Network (FANET) is an ad hoc network of UAVs. This is a mesh topological network and there is no fixed infrastructure. UAVs in a FANET usually belong to several users and therefore, it is a public network. UAVs of a FANET that are in the communication rage of a ground station may be connected to it but all other UAVs are connected to the ground station via several other relay UAV nodes. When a UAV changes its location, then it should be able to connect to a ground station from its new location. However, for this to be practically possible, there should a ground station or another relay UAV node within the wireless communication range of its new location. Therefore, for uninterrupted communication, there should be a large number of UAV nodes distributed over the geographical area concerned. Many researchers have proposed routing algorithms and protocols to increase the communication efficiency in FANETs, but they all operate on the assumption that there is at least one communication path to a ground station at all times. This means the UAV node density of the FANET should be a high. As they were derived from ad hoc networks, FANETs inherit most of the security issues that are preset in ad hoc networks. Therefore, the users need to have a special concern on network security of FANETs.

Internet of Drones (IoDs) is a cellular network [8]. In an IoD, the whole geographical area is divided in to a specific number of zones. Each zone consists of a ground station. All UAVs flying over a specific zone is directly connected to the ground station of that zone. If any UAV moves from its current zone to another zone, then it disconnects from the ground station of its current zone and establish a new communication link with the ground station of the new zone. This is similar to the mobile phone network. The whole network is owned and operated by several service providers and the UAVs cannot operate in areas where there is no network infrastructure established. Both FANETs and IoDs are public networks and are essentially Metropolitan Area Networks (MANs) and

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14

Wide Area Networks (WANs), depending on the size of their coverage area.

A LAN of Drones (LoD) or a Private UAV network is a relatively new concept and this network requires a minimum number of UAVs to perform any given task compared to a FANET or an IoD [9]. Also as this is a private network, there is less concern about the network security. A LoD consist of a ground station and branches of series connected UAVs. If any UAV connected to any UAV branch moves from its initial location to a different target location, then all UAVs connected to that specific branch will have to move accordingly to maintain communication to the ground station at all times.

When a single UAV of a LoD moves from its initial location to a target location, there are many alternative trajectories that the other UAVs in a branch of the network can make in order to maintain communication to the ground station at all times. Therefore, the trajectories of all UAV nodes should be optimized to obtain a desired maximum performance from the LoD. For instance, in certain applications, the task completion time can be critical than the other performance factors. Alternatively, in certain other applications, the total energy consumption of the UAV network may have to be minimized in order to have the best performance of the network. In this paper, we separately handle the two cases of minimizing the time and energy in a LoD when performing any given task.

The rest of the paper is organized as follows. In Section 2, we discuss a few literature related to the movement control of UAV nodes while maintaining the connectivity to a fixed network or a ground station. In Section 3, we discuss a method of maintaining communication in a LoD while minimizing task completion time. This is followed by the energy minimization method, which is discussed in Section 4. Finally, we conclude the paper in Section 5.

II. BACKGROUND

There are only a few research publications available on controlling UAV movements in UAV networks to maintain communication. Savla et al. [10] proposed a novel method to maintain connectivity among second order agents in an ad-hoc network of robotic agents. They have used double-integrator dynamics to develop distributed algorithms to maintain connectivity in the ad-hoc network. In this method, they have shown that any network node can maintain connectivity by maintaining a spanning tree of the network connectivity graph. Therefore, it is not compulsory to maintain connectivity of a particular node pair at all times. However, in this approach, they haven’t considered the ground station which acts as a stationary node. For UAV network applications it is essential for all UAVs to maintain connectivity to a stationary ground station which sends all control commands to the UAVs and receives data from the UAVs.

Zavlanos et al. [11] proposed a novel method to control the connectivity of mobile networks by studying various connectivity properties of dynamic graphs. Dynamic graphs can closely approximate properties of ground or aerial vehicle networks. They have graph theoretically formulated the connectivity problem by representing agents by vertices and network connections by a time varying edge set. They have maintained k-hop connectivity, where agents are allowed to be connected to the network when they are located less than k-hops away from a node. As complexity of graph connectivity problems grows exponentially with the number of nodes, this method becomes computationally expensive, when it comes to large networks.

Zavlanos et al.[12] presented another control method based on connectivity of dynamic graphs to control connectivity in multi-agent systems. In this method, they have performed the motion control of agents in the continuous state space and the topology control in the discrete graph space. In this method, particular links of the network are deleted throughout the movements of the agents. Therefore, gossip algorithms and distributed market-based algorithms are used to avoid any disturbances that can occur due to link deletion.

Stump et al. [13] has taken a similar approach to manage connectivity among mobile robot teams. They have developed a method to maintain communication between a stationary node and an exploring node. The exploring node is operated in an obstructive environment and, therefore, it is unable to maintain a line-of-sight communication with the stationary node. As such, there should be additional nodes in between these nodes in order to relay communication. They have used k-connectivity matrix to find the nodes, which can be connected by less than k number of nodes and the movements of the nodes are planned in a way that they satisfy the maximum hop count constraints. The second-smallest eigen value of Laplacian, which is also known as the Fiedler value is used to measure the connectivity of the mobile network.

Chiu et al. [14] proposed a bio-inspired, distributed control algorithm called *TENTACLES* to create a communication path between two stationary entities. The main concept behind their approach is to grow ‘tentacles’ from two entities which are situated apart from each other until they can establish a communication path. These ‘tentacles’ are made of self-organizing and self-healing robotic networks and these can operate in unknown environments with obstacles. Individual nodes of these ‘tentacles’ are controlled by the *TENTACLES* algorithm which is developed by the research team. *TENTACLES*

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14

algorithm consists of four main functions, Tentacle Building, Tentacle Rebuilding, Radio Guided Exploration and Local Flow Optimization. Using these main functions 'tentacles' grow in length in the direction of the other entity to be connected and select their positions in order to maximize the data flow rate.

All the node control methods discussed above consist of certain drawbacks in practical applications. Computational complexity can be identified as the most critical drawback, which may cause computational delays. Computational delays can be a problem when the UAVs operate at high speeds. In addition, these methods have not attempted to minimize the total energy consumption of the UAVs. Therefore, simple ways to maintain connectivity in LoDs under two optimization criteria, time and energy are required.

III. TIME MINIMIZATION

UAVs are used in crowd surveillance, natural disaster monitoring and many other critical applications, where delays in task completion can result in huge losses. Therefore, task completion time is one of the most critical factors, when it comes to UAV network applications. This section elaborates the method of minimizing the task completion time of a LoD.

1.1 Minimization Problem Formulation

A LoD consists of a ground station and branches of series connected UAVs. The number of UAV branches and the number of UAVs in each UAV branch can be varied depending on the requirement. Consider a LoD, which has single UAV branch with n+1 number of UAV nodes. When this UAV branch moves from its initial location to a given target location, then the initial and the final locations of the i^{th} nodecan be represented by $P_{iI}(x_{iI}, y_{iI}, z_{iI})$and $P_{iF}(x_{iF}, y_{iF}, z_{iF})$ (Figure 1).

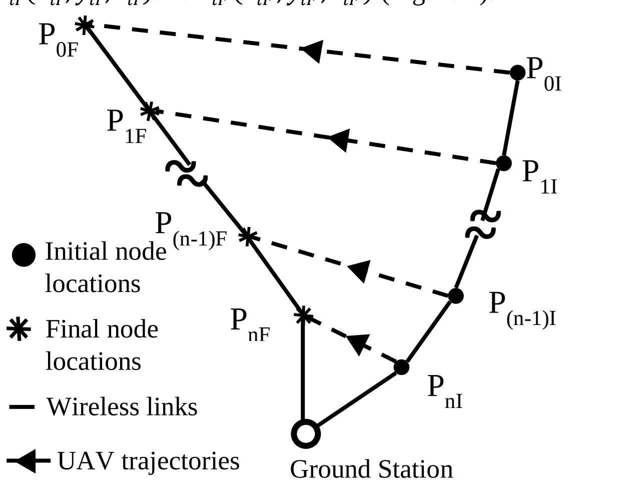


Figure 1: Mathematical notation for UAV node locations

Consider the i^{th} node of the given LoD in Figure 1. For a given task, the i^{th} node moves from its initial location of $P_{iI}(x_{iI}, y_{iI}, z_{iI})$ to the final location of $P_{iF}(x_{iF}, y_{iF}, z_{iF})$. The distance travelled by the node is given by the Euclidean distance between the point P_{iI} and point P_{iF} (Figure 2). Ground station location does not change over time, therefore consider the coordinate $P_{(n+1)F}(x_{(n+1)F}, y_{(n+1)F}, z_{(n+1)F})$ as the ground station location.

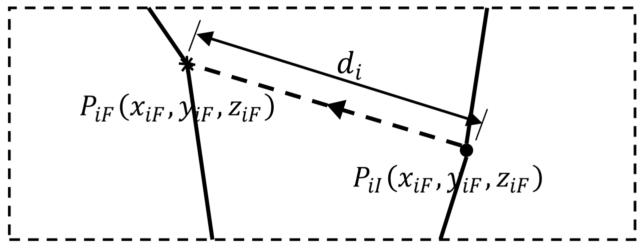


Figure 2: Trajectory of the i^{th} node

Thus, we can derive the distance travelled by the i^{th}node by following equation.

$$d_i(x_{iF}, y_{iF}, z_{iF}, \mathrm{x_{iI}}, \mathrm{y_{iI}}, \mathrm{z_{iI}}) = \sqrt{(x_{iF} - x_{iI})^2 + (y_{iF} - y_{iI})^2 + (z_{iF} - z_{iI})^2} \tag{1}$$

To simplify the notation, we introduce,

$$\boldsymbol{X_{iI}} = [x_{iI}, y_{iI}, z_{iI}]^T \text{for i} = 0,1, \dots, \text{n} \tag{2}$$

$$\boldsymbol{X_{iF}} = [x_{iF}, y_{iF}, z_{iF}]^T \text{for } \text{i} = 0, .1, \dots, \text{n} + 1 \tag{3}$$

Now, equation(1) can be rewritten in the following compact form.

$$d_i(\boldsymbol{X_{iI}}, \boldsymbol{X_{iF}}) = \sqrt{[(\boldsymbol{X_{iF}} - \boldsymbol{X_{iI}})^{\boldsymbol{T}}(\boldsymbol{X_{iF}} - \boldsymbol{X_{iI}})]} \tag{4}$$

While the leading UAV in a branch of a LoD moves from its initial location to a target location, all the UAVs should be able to maintain communication to the ground station at all times. Therefore, the distance between the consecutive UAV nodes in the branch should be less than the wireless communication range (R) of the UAV nodes. In order to maintain communication to the ground station continuously, the distance between the consecutive UAV nodes should satisfy the following inequality constraint.

$$\boldsymbol{D_{i-1,i}}(\boldsymbol{X_{iI}}, \boldsymbol{X_{iF}}) = \sqrt{\left[\left(\boldsymbol{X_{(i-1)F}} - \boldsymbol{X_{iF}}\right)^{\boldsymbol{T}}\left(\boldsymbol{X_{(i-1)F}} - \boldsymbol{X_{iF}}\right)\right]} \le R \tag{5}$$

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14

for all $i = 1, 2, \ldots\ldots, n+1$

Using equations(4) and (5), minimization problem can be formulated as:

Minimize,

$$\sum_{i=1}^{n} d_i(\boldsymbol{X_{iI}}, \boldsymbol{X_{iF}}) = \sum_{i=1}^{n} \sqrt{[(\boldsymbol{X_{iF}} - \boldsymbol{X_{iI}})^T(\boldsymbol{X_{iF}} - \boldsymbol{X_{iI}})]}$$

Subject to constraints,

$$g_i(\boldsymbol{X_{(i-1)F}}, \boldsymbol{X_{iF}}) = \sqrt{\left[(\boldsymbol{X_{(i-1)F}} - \boldsymbol{X_{iF}})^{\boldsymbol{T}}(\boldsymbol{X_{(i-1)F}} - \boldsymbol{X_{iF}})\right]} - R \leq 0 \quad (6)$$

for all $i = 1, 2, \ldots\ldots, n+1$

Introducing parameters λ_i, $\mathrm{i} = 1,2, \ldots, \mathrm{n}+1$, the Lagrangian function for this minimization problem can be written as:

$$L(\boldsymbol{X_{1F}}, \ldots \boldsymbol{X_{nF}}, \boldsymbol{X_{1I}}, \ldots \boldsymbol{X_{nI}}) = \sum_{i=1}^{n} d_i(\boldsymbol{X_{iI}}, \boldsymbol{X_{iF}}) + \sum_{i=1}^{n+1} \lambda_i g_i(\boldsymbol{X_{(i-1)F}}, \boldsymbol{X_{iF}}) \quad (7)$$

This is a constrained nonlinear optimization problem, where all constraints are nonlinear. This can be solved by numerical methods. We can use Sequential Quadratic Programming (SQP) method to iteratively find the optimum solution for nonlinear optimization problems given in equation (7).

1.2 Sequential Quadratic Programming (SQP)

SQP is one of the most effective methods of solving nonlinear constrained optimization problems [15]. This method generates linearly constrained optimization sub-problems of the nonlinear constrained optimization problem. These linearly constrained optimization sub-problems can be solved by Quadratic Programming methods.

Let's introduce,

$$x = [x_{1F}, y_{1F}, z_{1F}, x_{2F}, y_{2F}, z_{2F}, \ldots, x_{nF}, y_{nF}, z_{nF}] \quad (8)$$

Objective function can be reformulated as,

$$f(x) = \sum_{i=1}^{n} d_i(\boldsymbol{X_{iI}}, \boldsymbol{X_{iF}}) \quad (9)$$

Constraints can be reformulated as,

$$g(x) = [g_1(\boldsymbol{X_{0F}}, \boldsymbol{X_{1F}}), g_2(\boldsymbol{X_{1F}}, \boldsymbol{X_{2F}}), \ldots, g_{n+1}(\boldsymbol{X_{nF}}, \boldsymbol{X_{(n+1)F}})] \quad (10)$$

Thus, our nonlinear constrained optimization problem can be written in the following form.

$$\underset{x}{minimise} \quad f(x)$$

$$subject\ to \quad g(x) \leq 0 \quad (11)$$

The two term Taylor expansion of the objective function and the constraint functions around the pointx_k is given by

$$f(x) = f(x_k) + [\nabla f(x_k)]^T d + {}^1/_2\, d^T \nabla^2 f(x_k) d \quad (12)$$

$$g(x) = g(x_k) + [\nabla g(x_k)]^T d + {}^1/_2\, d^T \nabla^2 g(x_k) d \quad (13)$$

Notation$d$ is the size of one iterative step $(x - x_k)$.

Using equations(12) and (13), linearly constrained quadratic sub program can be formulated in the following form.

$$\underset{d}{minimise} \quad f(x_k) + [\nabla f(x_k)]^T d + {}^1/_2\, d^T \nabla^2 f(x_k) d$$

$$subject\ to \quad g(x_k) + [\nabla g(x_k)]^T d \leq 0 \quad (14)$$

This sub problem can be solved iteratively by Quadratic Programming until the iterative step $(x - x_k)$ become negligibly small and x converges to a local optimum. Following worked out example problem elaborates the use of SQP method in this specific type of application.

1.3 Numerical Example

Consider a LoD branch with two UAV nodes located on a two-dimensional 300m × 200m field (Figure 3). Ground station of the network is located at the coordinates (150, 10) and the UAV branch is initially located over left half of this field. Initial distances between the nodes are less than the wireless communication range (100m) and therefore, each node is initially connected to the ground station either directly or via a neighbouring UAV node. The task of the LoD is to send UAV0 to the target coordinates (250, 160). When executing this task, UAV1 also has to move in a way that both UAVs can maintain its connection to the ground station. This numerical example elaborates the method of finding the final location of UAV1, while minimizing the task completion time.

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14

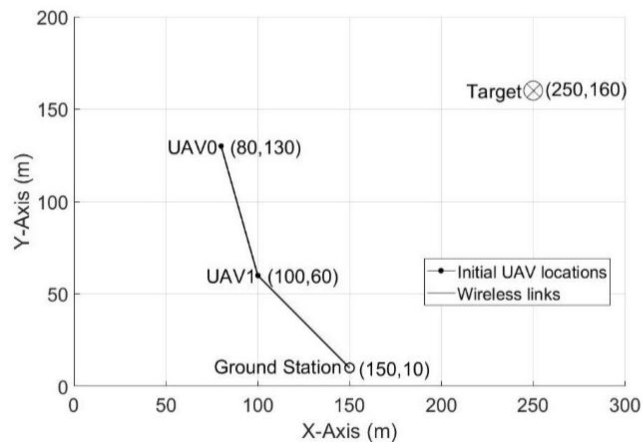


Figure 3: Initial UAV node locations

For given scenario, let's define,

$$x = [x_{1F}, y_{1F}] \quad (15)$$

In order to simplify the minimization problem, following objective and the constraint functions are squared to eliminate the square root terms in the functions. As all the functions consist of only non-negative distance values, this operation does not have any impact on the final optimized solution. Accordingly, the optimization problem for the given scenario is given by,

Minimize

$$f(x) = (x[1] - 100)^2 + (x[2] - 60)^2 \quad (16)$$

Subject to

$$g_1(x) = (250 - x[1])^2 + (160 - x[2])^2 - 100^2 \leq 0 \quad (17)$$

$$g_2(x) = (x[1] - 150)^2 + (x[2] - 10)^2 - 100^2 \leq 0 \quad (18)$$

Differentiating equation(16) with respect to$x$

$$\nabla f(x) = \begin{bmatrix} 2(x[1] - 100) \\ 2(x[2] - 60) \end{bmatrix} \quad (19)$$

$$\nabla^2 f(x) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \quad (20)$$

Differentiating equation(17) with respect to x,

$$\nabla g_1(x) = \begin{bmatrix} -2(250 - x[1]) \\ -2(160 - x[2]) \end{bmatrix} \quad (21)$$

From equation(18),

$$\nabla g_2(x) = \begin{bmatrix} 2(x[1] - 150) \\ 2(x[2] - 10) \end{bmatrix} \quad (22)$$

Let's use the initial location of UAV1 as the initial value of x for first iteration.

$$x_0 = [100,60] \quad (23)$$

The linearly constrained quadratic sub program of equation(14) to be used for the first iteration step is given by,

$$\underset{d}{minimise} \ {}^1/_2 \, d^T \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} d$$

Subject to,

$$150^2 + \begin{bmatrix} -300 \\ -200 \end{bmatrix}^T d \leq 0 \quad (24)$$

$$-5000 + \begin{bmatrix} -100 \\ 100 \end{bmatrix}^T d \leq 0$$

Iterative step size$d$solved using equation(24) is given by

$$d = [51.923, 34.615] \quad (25)$$

The value$x_1$for the next iteration is given by

$$x_1 = x_0 + d \quad (26)$$

Substituting values ofx_0 and$d$from equations (23) and (25)to equation(26)

$$x_1 = [151.923, 94.615]$$

This process is carried out iteratively until the x_k value converges to the final solution. TABLE 1 shows the results of each iteration of the optimization problem.

TABLE 1
OPTIMIZATION RESULTS OF SQP

Iteration No.	x_k	d
1	[100.000,60.000]	[51.923,34.615]
2	[151.923,94.615]	[13.744, 9.163]
3	[165.667, 103.778]	[1.120, 0.747]
4	[166.787, 104.524]	[0.007, 0.005]

In fourth iteration, both values of$d$ < 0.01 and therefore, final value of x converges to
$x = [166.79, 104.53]$

Above numerical example is based on the simplest case where there are only two UAV nodes in a two-dimensional environment. However, the calculation becomes more complex with higher number of UAV nodes in a three-dimensional environment. Therefore, a computer algorithm is essential for more complex problems. MATLAB *fmincon* function has an inbuilt SQP algorithm, which can be used for constrained nonlinear optimization problem solving. We use this function in all the following simulations.

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14

1.4 UAV Movements in Two-dimensional Space

In order to gain an in-depth understanding of the UAV movements, an initial study in a two-dimensional space would be useful. Therefore, the first few simulations are carried out in two-dimensional space.

Simulation I: We considered a two-dimensional 600m × 500m field in this simulation. A network with single UAV branch was initially located over right half of this field (Figure 4).The ground station of the network was located at coordinates (300, 20). The UAV branch consists of five UAV nodes and all consecutive nodes were initially located in each other's wireless communication range. The task of the simulation was to move UAV0 from its initial location to the target coordinates (100, 450) and observe the movements of all UAVs of the network. Figure 4shows the final locations of the UAVs after the completion of the task. UAV0 has successfully reached the target location and the distances between all the consecutive nodes did not exceed the specified wireless communication ranges. As a result, all the UAV nodes have maintained communication to the ground station successfully throughout the task.

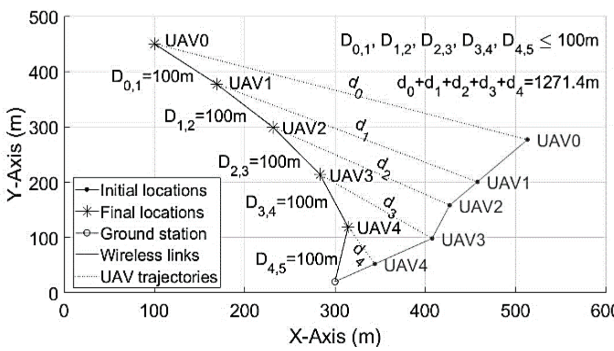


Figure 4: UAV movements in Simulation I

Simulation II: In any practical LoD application, the UAVs should be initially located on the ground closer to the ground station. Therefore, in this simulation, all the UAV nodes of the network were initially placed very close to each other in a 5m × 5m area next to the ground station (Figure 5). Then one of the UAVs (UAV0) was sent to the target coordinates (200, 700) and the movement of other UAVs of the branch was observed. In this simulation, a 2000m × 1000m two-dimensional field was used. Throughout the task, all the UAVs connected to the branch moved along with the UAV0 in order to relay the communication to the ground station. The distances between all consecutive nodes were maintained less than the wireless communication range.

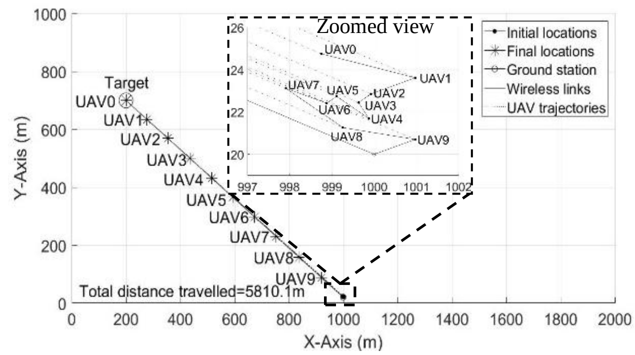


Figure 5: UAV movements in Simulation II

1.5 UAV Movements in Three-dimensional Space

In real world UAV applications, the UAVs operate in three-dimensional space. Therefore, simulations carried out in three-dimensional space are better approximation of real world UAV applications.

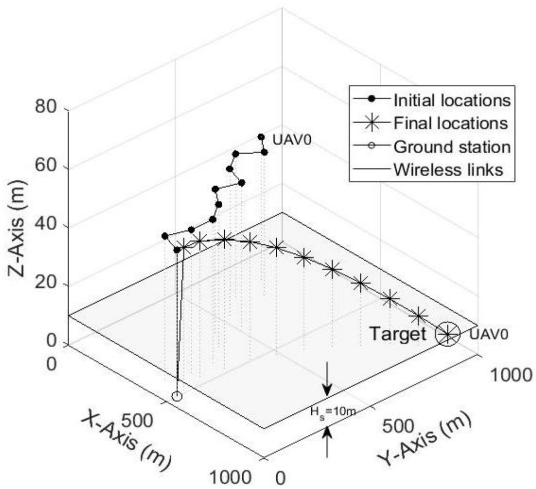
The UAVs that perform tasks in a given area of land may encounter obstacles such as humans, animals or vehicles. These kinds of obstacles are mobile in nature and cannot predict in advance. Therefore, it is appropriate to fly at a safe height from the ground level at all times.

Let the safe height from the ground be H_s to avoid any small-scale obstacle. Then the UAV movements should satisfy the following constraint in addition to the constraints given in equation(6). It is assumed that the z-coordinate of the ground level is zero.

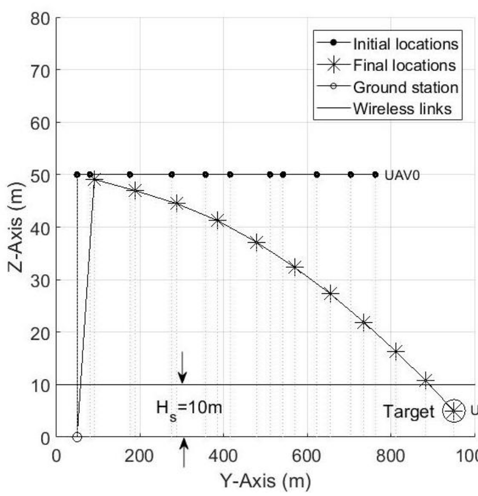
$$z_{iF} > H_s \text{ for all } i = 1, 2, \dots, n \qquad (27)$$

Simulation III: In this simulation, we considered a land with 1000m × 1000m area. A LoD consisting of a single branch of 11 UAV nodes was initially located 50m above the land (Figure 6).

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14



a. Three-dimensional view

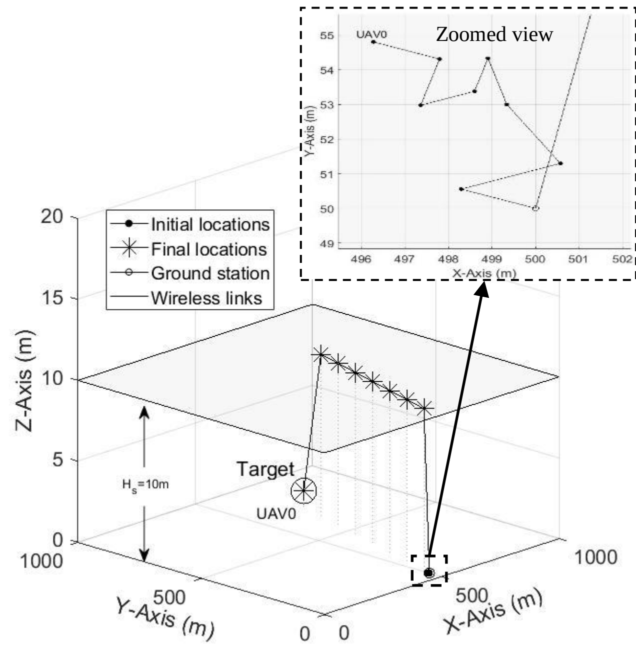


b. YZ-plane

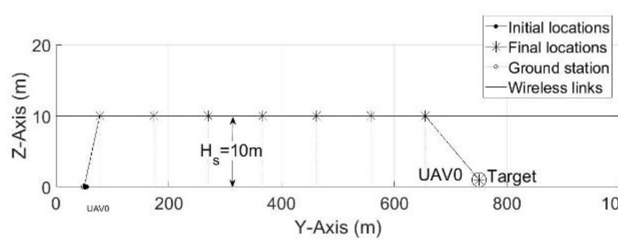
Figure 6: UAV movements in Simulation III

Then the farthest UAV of the network (UAV0) was given a command to move to the target location (900, 950, 5), which was located just 5m above the ground. A constraint for the safe height from the ground in the rest of the area of the land of 10m ($H_s = 10m$) was imposed. In Figure 6 b, it can be clearly seen that all the UAVs have maintained their heights above safe height after completion of the task, except UAV0, which has successfully reached the target, which was located below the safe height. The network has maintained the distance between each consecutive node to be less than the wireless communication range (100m) and therefore, the communication to the ground station was successfully maintained throughout the task.

Simulation IV: In this simulation, a 1000m × 1000m land was used and the UAV branch consisted of eight UAV nodes. All the UAV nodes of the network were initially placed on the ground very close to each other in a 5m × 5m area adjacent to the ground station (Figure 7).



a. Three-dimensional view



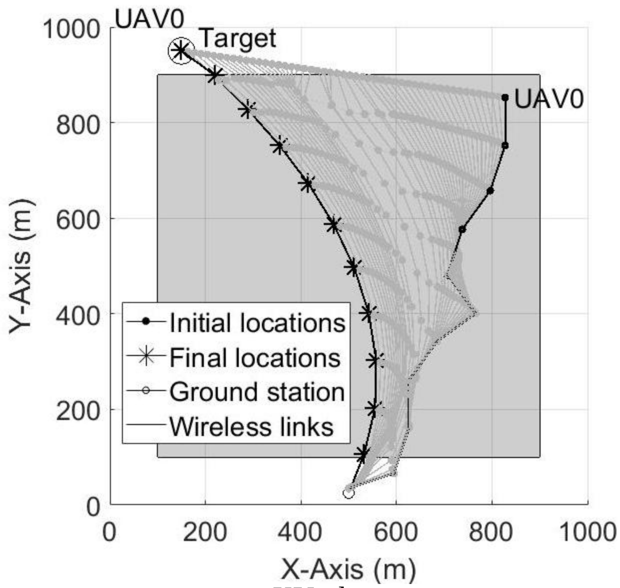
b. YZ-plane

Figure 7: UAV movements in Simulation IV

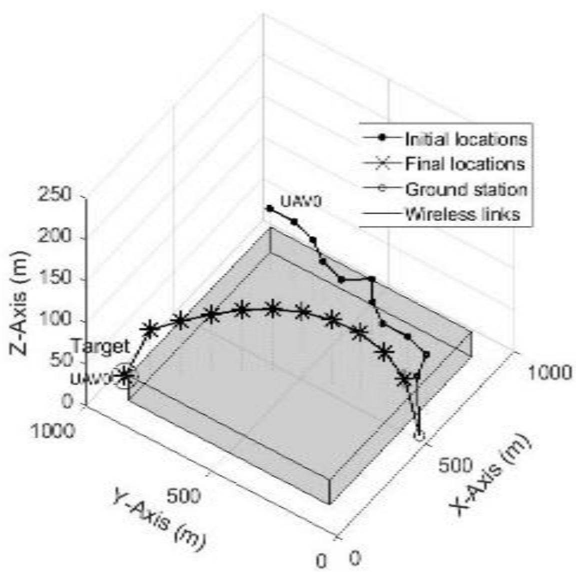
Then one of the UAVs (UAV0) was sent to the target coordinates (700, 750, 1), which was located just 1m above the ground level. After completion of the task, all UAVs have cleared the safe height ($H_s = 10m$) except UAV0. At the end of the task, UAV0 has successfully reached the target, which was located at 1m height from the ground while maintaining the communication path to the ground station.

Simulation V: In this simulation, we considered a land with 1000m × 1000m area with a large (800m × 800m × 30m) cuboid obstacle at the center (Figure 8).

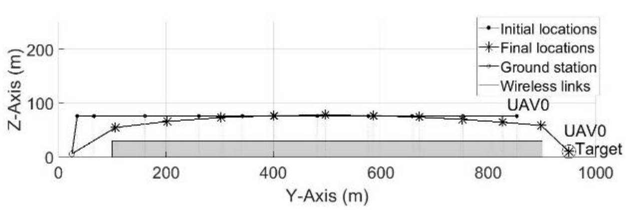
e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14



a. XY-plane



b. Three-dimensional view



c. YZ-plane

Figure 8: UAV movements in Simulation V

An 11-node UAV branch was initially located 70m above the land and the leading UAV was given a command to move to the target location (150, 950, 10). Figure 8a shows the UAV locations after each iteration. In Figure 8c, it can be clearly seen that all the UAVs have avoided the obstacle and the consecutive UAV nodes have maintained the line-of-sight communication by adhering to the constraints.

IV. ENERGY MINIMIZATION

The weights of the batteries have a huge effect on the payloads of the UAVs. Due to the size and weight constraints of the batteries, UAVs have limited battery capacities. Therefore, optimal usage of battery energy is one of the essential requirements of UAV applications. This section elaborates the methods of minimizing the total battery energy consumption of the UAVs of a LoD.

2.1 Minimization Problem Formulation

First, a mathematical equation should be developed to estimate the energy consumption of each individual UAV of a LoD. In ideal conditions, a UAV consumes battery energy in three basic forms, kinetic energy, potential energy and the energy required for hovering. In ideal conditions, energy dissipation due to drag forces and air turbulences of atmosphere are assumed to be negligible in comparison to these three basic forms of energy consumptions.

The total energy consumption of the i^{th} UAV node can be derived by adding the kinetic energy (E_{ik}), potential energy (E_{ip}) and the hovering energy (E_{ih}) together,

$$E_i = E_{ik} + E_{ip} + E_{ih} \tag{28}$$

Consider a UAV hovering at a constant location. If this UAV is given a task to move from its initial location to a target location, then it has to accelerate and reach a specific velocity. At this point, the battery energy is utilized to increase the thrust of its motors to gain its intended velocity and the battery energy is converted to kinetic energy of the UAV. Then the UAV move at the direction of its target location. When the UAV gets closer to the target, it should apply brakes to decelerate and gradually settle at the target location while hovering.

Consider the i^{th} UAV node of a LoD branch. For simplification, assume the acceleration and the deceleration times of the UAV are negligible and the UAV node moved from its initial location to the target location at a constant velocity of v_i. Assuming there is no regeneration of energy when applying brakes, the kinetic energy loss of the i^{th} node having a mass m is given by,

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14

$$E_{ik} = {}^{1}/_{2} \times m \times {v_i}^2 \tag{29}$$

Consider a UAV hovering at a certain location. If this UAV is given a task to move to a target location, which is at a higher latitude than its current latitude, then the propellers of the UAV have to work against the gravitational force to reach the target. This task converts battery energy to the potential energy of the UAV. Therefore, the energy consumption can be simply determined by calculating the potential energy difference of the UAV.

Consider the i^{th} UAV node of a LoD branch and it is initially located at a height ofz_{iI} from ground level. If this UAV is moved to a target located at z_{iF} height, then the potential energy loss of the i^{th} node having a mass m is given by,

$$E_{ip} = m \times g \times (z_{iF} - z_{iI}) \tag{30}$$

Regardless of the location or the velocity of a UAV, it always consumes approximately a constant amount of energy to maintain its altitude. This energy consumption rate is the hovering power P_h of that UAV. This value is dependent of the properties of that specific make and model of the UAV. If the i^{th} UAV node stays in air for T_i amount of time to perform a task, then the energy dissipated due to hovering of the i^{th} node during the accomplishment of the task is given by,

$$E_{ih} = P_h \times T_i \tag{31}$$

The flying time of the i^{th} node (T_i) can beexpressed in terms of its velocity (v_i) by following equation.

$$T_i = \frac{\sqrt{[(\boldsymbol{X}_{\mathbf{i}F} - \boldsymbol{X}_{\mathbf{i}I})^T(\boldsymbol{X}_{\mathbf{i}F} - \boldsymbol{X}_{\mathbf{i}I})]}}{v_{\mathrm{i}}} \tag{32}$$

Substituting value of T_i from equation (32) to equation (31), we obtain,

$$E_{ih} = P_h \times \frac{\sqrt{[(\boldsymbol{X}_{\mathbf{i}F} - \boldsymbol{X}_{\mathbf{i}I})^T(\boldsymbol{X}_{\mathbf{i}F} - \boldsymbol{X}_{\mathbf{i}I})]}}{v_{\mathrm{i}}} \tag{33}$$

Substituting E_{ik}, E_{ip} andE_{ih} values from the equations(29), (30) and (33) to equation (28),

$$E_i = \left({}^{1}/_{2} \times m \times {v_i}^2\right) + \left(m \times g \times (z_{iF} - z_{iI})\right) + \left(P_h \times \frac{\sqrt{[(\boldsymbol{X}_{\mathbf{i}F} - \boldsymbol{X}_{\mathbf{i}I})^T(\boldsymbol{X}_{\mathbf{i}F} - \boldsymbol{X}_{\mathbf{i}I})]}}{v_{\mathrm{i}}}\right) \tag{34}$$

In real word UAV applications, energy utilized askinetic energy (E_{ik}) is negligible compared to energy utilized for hovering (E_{ih}).

$$\left({}^{1}/_{2} \times m \times {v_i}^2\right) \ll \left(P_h \times \frac{\sqrt{[(\boldsymbol{X}_{\mathbf{i}F} - \boldsymbol{X}_{\mathbf{i}I})^T(\boldsymbol{X}_{\mathbf{i}F} - \boldsymbol{X}_{\mathbf{i}I})]}}{v_{\mathrm{i}}}\right) \tag{35}$$

Therefore, v_i should be maximized, in order to minimize the total energy consumption of UAVs. Accordingly, all UAVs should travel at their highest possible velocity to minimize total energy consumption of the LoD. However, each UAV node of the LoD travels different distances and therefore, if they travel at equal velocities, some UAVs might finish the task earlier than others. This can lead to communication link failures between consecutive UAV nodes. In order to maintain communication to the ground station at all times, all consecutive UAV nodes should maintain a distance less than their specified wireless communication range. Therefore, all UAV nodes of the LoD branch should start and end their task together to avoid neighboring nodes moving far-off from each other. As such, the flying times (T_i) for all UAV nodes should be made equal.

$$T = T_i \text{for all } i = 1, 2, \ldots, n \tag{36}$$

As discussed in the time minimization method, in a LoD branch, the maximum distance is travelled by the farthest UAV of the network (UAV0). Assuming that this UAV travelled at its highest possible velocity ($v_{\max}$), the time consumed by this UAV to perform the task can be calculated by,

$$T = \frac{\sqrt{[(\boldsymbol{X}_{\mathbf{0}F} - \boldsymbol{X}_{\mathbf{0}I})^T(\boldsymbol{X}_{\mathbf{0}F} - \boldsymbol{X}_{\mathbf{0}I})]}}{v_{\max}} \tag{37}$$

As all the other UAVs have to start and end their tasks together, we can calculate the velocities of all other UAVs using the following equation.

$$v_i = \frac{\sqrt{[(\boldsymbol{X}_{iF} - \boldsymbol{X}_{iI})^T(\boldsymbol{X}_{iF} - \boldsymbol{X}_{iI})]}}{T} \tag{38}$$

Substituting value of T from equation (37) to equation (38)

$$v_i = \frac{\sqrt{[(\boldsymbol{X}_{\mathbf{i}F} - \boldsymbol{X}_{iI})^T(\boldsymbol{X}_{iF} - \boldsymbol{X}_{iI})]}}{\sqrt{[(\boldsymbol{X}_{\mathbf{0}F} - \boldsymbol{X}_{\mathbf{0}I})^T(\boldsymbol{X}_{\mathbf{0}F} - \boldsymbol{X}_{\mathbf{0}I})]}} \times v_{\max} \tag{39}$$

The minimization function to minimize the total energy utilized in a LoD can be constructed using equations(34),(37)and(39).

Minimize,

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14

$$\sum_{i=1}^{n} E_i = \sum_{i=1}^{n} \left\{ \begin{array}{c} \left(\begin{array}{c} {}^{1}/_{2} \times m \times \\ \left(\dfrac{\sqrt{[(\boldsymbol{X_{iF}} - \boldsymbol{X_{iI}})^T(\boldsymbol{X_{iF}} - \boldsymbol{X_{iI}})]}}{\sqrt{[(\boldsymbol{X_{0F}} - \boldsymbol{X_{0I}})^T(\boldsymbol{X_{0F}} - \boldsymbol{X_{0I}})]}} v_{\max} \right)^2 \\ +(m \times g \times (z_{iF} - z_{iI})) \end{array} \right) \\ +P_h \times \dfrac{\sqrt{[(\boldsymbol{X_{0F}} - \boldsymbol{X_{0I}})^T(\boldsymbol{X_{0F}} - \boldsymbol{X_{0I}})]}}{v_{\max}} \end{array} \right\} \tag{40}$$

Subject to same constraints applied in time minimization,

$$g_i(\boldsymbol{X_{(i-1)F}}, \boldsymbol{X_{iF}}) = \sqrt{\left[(\boldsymbol{X_{(i-1)F}} - \boldsymbol{X_{iF}})^{\boldsymbol{T}} (\boldsymbol{X_{(i-1)F}} - \boldsymbol{X_{iF}}) \right]} - R \le 0 \tag{41}$$

for all $i = 1, 2, \ldots\ldots, n$

There is no closed form solution for this minimization problem. Therefore, the non-linear programming tool provided in MATLAB software was used to solve the energy minimization problem as well.

2.2 UAV Movements in Two-dimensional Space

The first few simulations were carried out considering only two-dimensional space, because two-dimensional plots are clearer and easier to understand. The wireless communication range of each UAV node was assumed as 100m and the make and model of each UAV node was assumed as AR Drone 2.0. Following property values of AR Drone 2.0 was acquired from a research carried out to model power and endurance of rotorcraft [16].

Maximum velocity ($v_{\max}$) = 11.11 m/s
Mass (m) = 495 g
Hovering power (P_h) = 75w

Simulation VI: We considered a two-dimensional 600m × 500m field in this simulation. A UAV network branch initially located over the left half of the field (Figure 9). Then the UAV0 was given a command to move to the target located at coordinates (600, 300) under the two optimization criteria, minimum energy and minimum time. Figure 9 shows the final locations and the total energy consumption of the UAVs under each criterion. Energy minimized method has the least energy consumption. However, as this simulation is carried out only for two-dimensional field, the potential energy difference has not taken in to consideration. Therefore, the energy difference under the two criteria is insignificant.

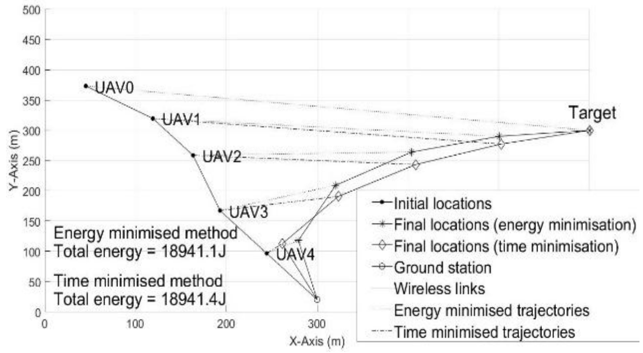


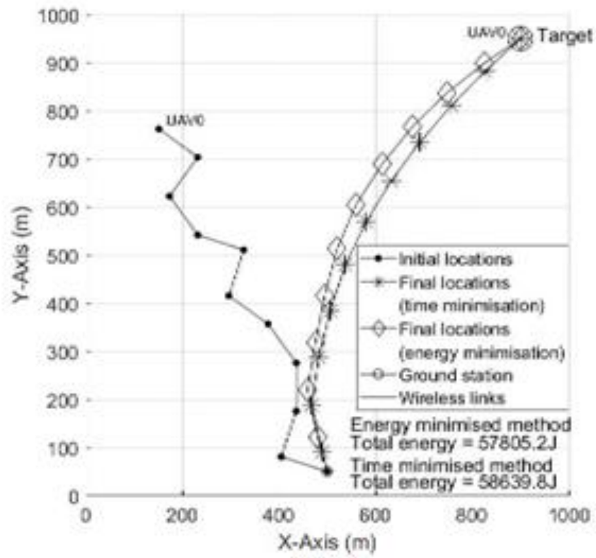
Figure 9: UAV movements in Simulation VI

2.3 UAV Movements in Three-dimensional Space

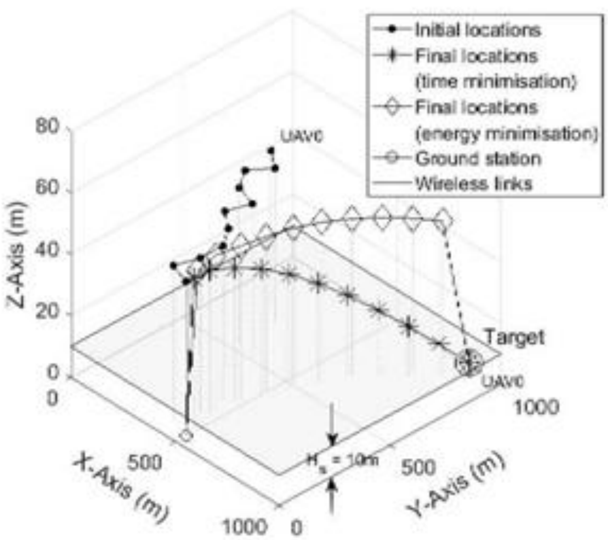
In two-dimensional simulations, the UAV movements in z-direction were disregarded. However, the z-direction movements of UAVs critically affect the energy consumption of the UAVs due to potential energy difference caused by movements on the z-direction. Therefore, for better approximation of real word UAV network applications, three-dimensional simulations will be carried out in this section. As previously mentioned, UAVs should maintain a safe height (H_s) from the ground level in order to avoid any small non stationary obstacles such as humans, animals or vehicles. Therefore, additional constraint of equation(27) has been used in the following simulations.

Simulation VII: We considered the same simulation environment as in Simulation III**.** Eleven UAV nodes were initially located over a 1000m × 1000m land(Figure 10).

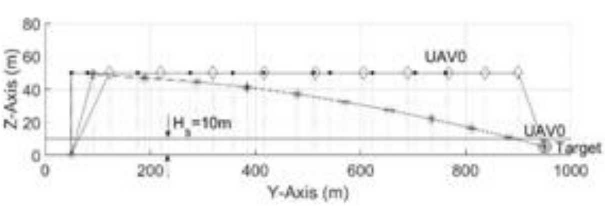
e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14



a. XY-plane



b. Three-dimensional view



c. YZ-plane

Figure 10: UAV movements in *Simulation* VII

All UAVs were initially located 50m above the ground level. In the following simulation, UAV0 was directed to the target location (900, 950, 5), which was 5m above the ground level. Same simulation was carried out in both energy-minimized and time-minimized methods in order to compare energy consumptions under both minimization criteria. UAV0 has successfully reached the target in both minimization criteria. According to the simulation results, energy consumption in energy minimized method was 57805.2 J and that in time minimized method was 58639.8 J. Accordingly, the energy consumption in energy minimized method has reduced by 1.42% (834.6 J) compared to the time minimized method.

2.4 UAV Movements in Three-dimensional Obstructive Environment

As previously discussed, in real word UAV network applications, the UAVs often operate in obstructive environments and therefore should include additional constraints to avoid obstacles. Following simulations were carried out for the energy-minimized case.

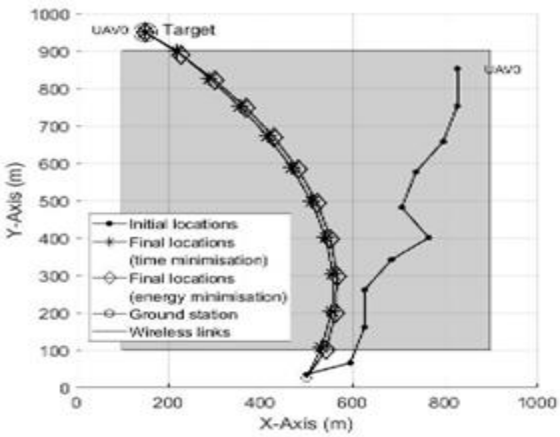
Simulation VIII: We considered the same simulation environment used in Simulation V. A 1000m × 1000m land consisted of a large (800m × 800m × 30m) cuboid obstacle at the center (Figure 11). An 11-node UAV branch was initially located 70m above the land and the leading UAV was directed to the target location (150, 950, 10).Same simulation was carried out in both energy-minimized and time-minimized methods in order to compare energy consumptions under both minimization criteria. According to the simulation results, energy consumption in energy minimized method was 65446.7 J and that in time minimized method was65762.7J. Accordingly, the energy consumption in energy minimized method has reduced by 0.48% (316.0 J)compared to the time minimized method.

V. CONSLUSION

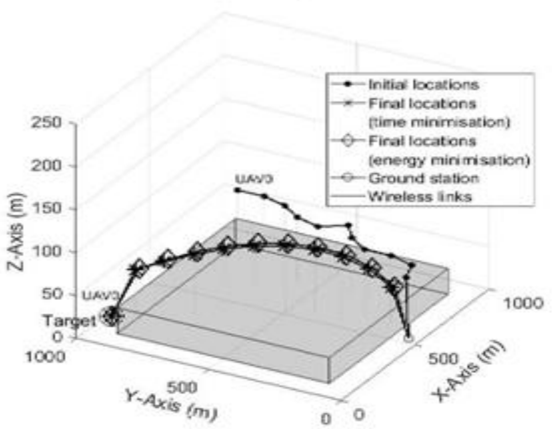
In this paper, we have considered the methods of obtaining optimal trajectories for UAV nodes in a LoD to minimize time and energy to complete a given task. Initially, we developed objective functions corresponding to task completion time and total energy consumption of a LoD. Then both these objective functions were minimized under the constraints of maintaining connectivity between consecutive UAV nodes. These were nonlinear constraints and therefore Sequential Quadratic Programming method was used to calculate optimal solutions numerically. Additional constraints were applied to avoid obstacles and maintain line-of-sight communication in obstructive environments. Both time minimization and energy

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume- 9, Issue- 1, (February 2019)
https://doi.org/10.31033/ijemr.9.1.14

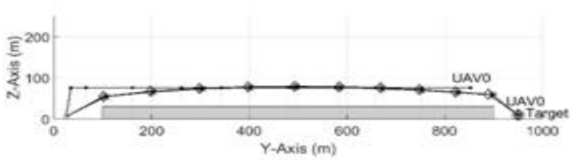
minimization methods successfully performed a range of tasks in obstructive environments while maintaining connectivity to the ground station at all times.



a. XY-plane



b. Three-dimensional view



c. YZ-plane

Figure 11: UAV movements in Simulation VIII

REFERENCES

[1] *Research and markets adds report: Photography drones*. (2015). ed. Jacksonville.
[2] *Omniscan, Australia's provider of UAV asset inspections*. (2015). ed. London.
[3] *Cepton releases lightweight 3D LiDAR sensing solution for UAV Mapping*. (2017). ed. Jacksonville.
[4] Supply And installation of Uav based thermal imaging system. (2019). ed.
[5] B. Stevenson. (2014). DHL to test supply delivery using UAVs.(Flight International)(Deutsche Post)(unmanned air vehicles). *Flight International,* no. 982.
[6] J. Drinan. (2016). I want my amazon prime air. *Planning, 82*(7), 1-2.
[7] İ. Bekmezci, O. K. Sahingoz, & Ş. Temel. (2013). Flying ad-hoc networks (FANETs): A survey. *Ad Hoc Networks, 11*(3), 1254-1270.
[8] M. Gharibi, R. Boutaba, & S. L. Waslander. (2016). Internet of drones. *Access, IEEE, 4*, 1148-1162.
[9] R. d. Silva & S. Rajasinghege. (2018). Optimal desired trajectories of UAVs in private UAV networks. *IMAV2018 Proceedings*, 310-314.
[10] K. Savla, G. Notarstefano, & F. Bullo. (2009). Maintaining limited-range connectivity among second-order agents. *SIAM Journal on Control and Optimization, 48*(1), 187-205.
[11] M. M. Zavlanos & G. J. Pappas. (2005). Controlling connectivity of dynamic graphs. *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, 6388-6393.
[12] M. M. Zavlanos & G. J. Pappas. (2008). Distributed connectivity control of mobile networks. *IEEE Transactions on Robotics, 24*(6), 1416-1428.
[13] E. Stump, A. Jadbabaie, & V. Kumar. (2008). Connectivity management in mobile robot teams. *ed, 2008*, 1525-1530.
[14] H. C. H. Chiu *et al.* (2009). TENTACLES: Self-configuring robotic radio networks in unknown environments. *ed, 2009*, 1383-1388.
[15] P. T. Boggs & J. W. Tolle. (2000). Sequential quadratic programming for large-scale nonlinear optimization. *Journal of Computational and Applied Mathematics, 124*(1), 123-137.
[16] A. Abdilla, A. Richards, & S. Burrow. (2015). Power and endurance modelling of battery-powered rotorcraft. in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 675-680.