DISCOVERY

An improved energy-efficient data clustering in UAV-Aided Wireless Sensor Networks for uneven topology

Fagbohunm, Griffin Siji

To Cite:
Fagbohunm, Siji G. An improved energy-efficient data clustering in UAV-Aided Wireless Sensor Networks for uneven topology. *Discovery* 2023; 59: e114d1355
doi: https://doi.org/10.54905/disssi.v59i333.e114d1355

Author Affiliation:
[1]Department of Computer Engineering Abia State University, Uturu, Abia State, Nigeria
fagbhume.griffn@abiastateuniversity.edu.ng

Peer-Review History
Received: 12 June 2023
Reviewed & Revised: 16/June/2023 to 06/October/2023
Accepted: 10 October 2023
Published: 14 October 2023

Peer-Review Model
External peer-review was done through double-blind method.

Discovery
pISSN 2278–5469; eISSN 2278–5450

ABSTRACT

It is not uncommon to see sensor nodes deployed in an uneven or hilly terrain. This can be found in many parts of Nigeria. In the same vain, sensor nodes may be deployed in very hostile areas such as in northern parts of Nigeria where insurgents are heavily present. In areas such as those stated above, the use of unmanned aerial vehicles can be used for energy-efficient data collection from the scattered sensor nodes. Unmanned Aerial Vehicles operating at low altitudes can be used to lower the energy consumption of the wireless sensor network by using an intelligent data collection methodology to distribute the UAVs for data collection from the nodes. This paper proposes an energy-efficient and optimized data aggregation (EEODA) scheme in UAV-assisted wireless sensor network for hilly and uneven terrain is designed, using UAVs as data collection points. This can be achieved through the following steps, firstly, a distributed clustering algorithm based on reinforcement learning was proposed to organize the wireless sensor nodes, secondly, a mono-objective simulated annealing search method will be used to efficiently distribute the UAVs for optimum collection of data from the various cluster heads in the network, thirdly the city section mobility model will be used to compute the optimum position for the UAVs to each of the cluster heads in the network. Simulation results show that EEODA scheme proposed in this paper outperforms the EFDC, the closest-performing algorithm to it with an average of 12%. It also outperforms the other two compared algorithms, LEACH with UAV and HEED with UAV with between 17% and 36%, respectively with performance metrics such as energy consumption of nodes, scalability, delay in data aggregation and collection, control overhead and number of dead nodes in each round of clustering.

Keywords: Aggregation, data collection, genetic algorithm, reinforcement algorithm, HEED, Unmanned Aerial vehicle, drone, modified simulated annealing, wireless sensor network.

1. INTRODUCTION

The study of wireless sensor networks has been on the increase in recent time due to its application in many data distribution and dissemination scenarios. Their

DISCOVERY SCIENTIFIC SOCIETY

applications include but the following, home automation, industry automation, environmental monitoring, forest observation, monitoring patient's health, underwater communication etc. As sensor nodes are usually battery powered, and therefore power constrained as batteries will eventually run out of energy (Ogundile and Alfa, 2020). For this purpose, an energy-efficient algorithm should be designed so as to extend the lifetime of the wireless sensor networks. Many researchers have developed energy-efficient protocols for routing Nayak and Vathasavai, (2017) and clustering Arafat, et al., (2022) in wireless sensor networks.

Many researchers have designed an optimum medium access control or wireless sensor networks Liu et al., (2018) and optimal positioning of mobile sinks (Dandekar and Deshmukh, 2013). In many instances, WSNs are deployed in hilly and uneven environment where data aggregation and forwarding to sinks will be difficult due to absence of network infrastructure. With this type of difficult scenario, mobile robots can be used for data collection from the wireless sensor nodes, however robot mobility can be impeded by uneven or bumpy surfaces, which is the target topology in this paper. The non-suitability of mobile robots for wireless sensor networks in hilly and bumpy topology has resulted in the study of Unmanned Aerial Vehicles (UAV) for data collection in such scenarios. UAV has the advantage that it can flown towards any sinks in the WSN and hence cannot be disturbed by the hilly and uneven surface topology of the environment.

The investigation of the possible application of mobile sink in gathering data from a network has resulted in the use of unmanned aerial vehicles (UAVs) for data gathering in wireless sensor networks. Due to the possibility of UAVs to flying towards an area of the network and its being controlled by 3D software, make them very suited for the mobile sink scenario (Nagata et al., 2020). UAVs, have the advantage over ground robots in that obstacles on the ground do not impair their mobility. They are therefore able to cover more distance at less time frame and with much less effort (McAuthor et al., 2018). The use of UAVs for data aggregation in wireless sensor networks, therefore is a new paradigm shift in research approach where sensors are located in less accessible areas (Chen et al., 2019). The use of UAVs for data aggregation in WSN therefore reduces the importance of sensors necessarily being in line of sight as in the conventional wireless sensor networks demands. UAV has a position optimization routine in its firmware, enabling it to get a line-of-sight connection between two or more sensor nodes even though they are not in actual sight. From literature, there are two types of UAV, Multi-rotor and fixed-wing UAVs.

Multi-rotor UAV can change its moving direction better than the fixed-wing UAV, so it is more suited for the network scenario proposed in this paper. Wireless sensor networks are usually deployed randomly in a given setting; therefore, they are unlikely to be distributed uniformly in a given network landscape. Due to this, a distributed clustering technique is better than a flat topology wireless sensor networks. In clustered network, the sensor nodes are divided into groups so that a given cluster will not be overwhelmed and ensure equitable node distribution for an even spread of data transmission to the sink. In the case of ad-hoc networks there is no information as to the location of cluster heads. In this situation, the clustering algorithm will be evoked many times, leading to the protocol automatically electing the cluster head after each round. Also, the failure of a cluster head to transmit data to the sink will cause the isolation of the sensed data from that cluster. For these reasons, the use of hierarchical data dissemination is not suited for the application scenario proposed in this paper as there are no access points (APs).

For bumpy topology, other novel techniques are used, including borderline detection Hammoudeh et al., (2017), the prediction of landslide Sumathi et al., (2019), prediction of instantaneous rise in quantity of nodes in a particular region of the network Eckerstorfer et al., (2016) as well as innovative data aggregation scheme. For the scenarios stated above, there is a need for remote sensing as most of these network areas cannot be accessed by humans. The use of conventional data aggregation scheme requires network infrastructure and therefore not feasible in a bumpy terrain / hilly surface proposed in this paper. It should be noted that the presence of static network infrastructure will require a lot of time for data to get to the sink in addition to more energy consumption, especially when communication is done over long range as is bound to be experienced in uneven and hilly surfaces. These situations result in reduced network lifetime of the sensor nodes as batteries usually power them. The use of flying mobile sinks can reduce energy consumption in these scenarios. The use of flying mobile sinks have the effect of reducing the distance between the transmitter and the receiver due to the inbuilt line of sight feature in the UAVs.

However, there is a need for improvement in certain areas of UAV operation, a 3D mobility feature must be included in order to navigate the network topology proposed in this paper. The EEODA protocol proposed in this paper will exploit the 3D mobility capability in the quadcopter's UAV firmware as well as the ability to hover steadily over a given region of interest. The UAVs will also be programmed to hover over the proposed network using a lower altitude than is usually used. Hovering at a lower altitude is necessary to reduce the energy consumption of UAVs as they hover around the network for data collection from the sensors. This paper proposes a low latency and energy-efficient data aggregation (EEODA) scheme in wireless sensor networks well suited for uneven and hilly surfaces. In the proposed design, data will be received from randomly placed sensors in the network by a multi-rotor UAV deployed in an uneven terrain. The contribution made in this paper includes the following: The design of a reduced latency transmission between the cluster heads and the UAVs using Dijkstra's algorithm. The network is deployed in an infrastructure-less network without a base station or network control centre.

The EEODA protocol will therefore operate on an ad-hoc network. This paper is specifically an improvement over the Energy-efficient and fast data collection (EFDC) protocol proposed by researchers in Rezoan and Sangman, (2021), where the UAVs receive sensed data directly from all the sensors in each cluster. The argument in this paper is that such arrangement leads to quick energy depletion by all the nodes in the network at approximately the same time. This paper proposes a hierarchical clustering algorithm based on Q-learning, this is due to the fact that the sensor nodes in the network are battery powered, and also that the nodes may contain an avalanche of data or nodes having similar data to be sensed. In these scenarios, there may be a high volume of data transmission between the sensors and UAVs in the case of the avalanche data scenario or duplicates of data transmitted where similar data are sensed from sensors in a cluster. In the above stated scenario, the energy depletion of nodes in the EFDC protocol will be high, leading to a decline in the expected lifetime of the network. A second shortcoming of the EFDC protocol is its poor latency in a high-density network, this fact that is evidenced in the experimental data and results presented later in the paper.

The proposed EEODA protocol is suited for a network with many nodes and deployed in hilly or uneven terrains. The clustering algorithm does not require many rounds of reorganizing the sensors as a new round is only invoked when the Q-value of the current cluster head reduces below a threshold value, and even in such a case the re-election of cluster heads assume a role-free model where simple comparison computation is done. Another advantage is that a model of the environment is not required. Due to the above reasons, the overhead in transmission is reduced using the reinforcement learning algorithm. It should also be noted that the rounds of clustering are further reduced as the UAV acts as data collection centres for locating the position of cluster heads in the network. In this regard, the UAVs are able to change their positions when the topology of the network change, such as arising from changes in the number of sensors or changes in the distribution of sensed data. The positions of the UAVs are determined by the aggregate of received signal strength (RSS) from the sensor nodes. There is also a need for sensor clustering re-organization when the volume of sensed data in a particular region or part of the network changes considerably.

The use of UAVs as data collector agent is used by researchers in (Khan et al., 2013). Additionally, EEODA exploits the 3D capability of multi-rotor quadcopter UAVs using a modified multi-objective simulated annealing search method in reducing distance of transmission between the UAVs and the cluster heads in the network. It should be emphasised here, that though the EEODA protocol is designed for data collection in high density sensor network, located in hilly and bumpy surfaces, its aggregation capability is comparable to those in flat and low-density sensor networks. This is a very promising capability considering the infrastructure-less nature of the network considered in this paper. This paper has the following contributions to the research community. First, a role-free clustering algorithm based on Q-learning is proposed, which helps to balance the quantity of sensor nodes in each cluster. For this algorithm every node in a cluster elects a cluster head (CH) to send data to the UAV using a combination of its distance to the UAV, the received signal strength, and the remaining battery energy in the node. When the data gets to the cluster head from the different nodes, it aggregates the data and transmit them along neighbouring cluster head to the UAV closest to it.

It should be noted that each cluster has a counter that keeps the record of all data received from each neighbour. The condition here is that a cluster head can only forward data to one of the neighbour clusters with the least message count from that particular cluster head. The amount of messages received by a node is a determining factor used in calculating the node's residual energy. This means that a cluster head with a high number of messages received is assumed to be low in residual energy, so data will not be forwarded to the closest UAV. The method ensures that there is a balance of energy distribution among the cluster heads and routing is not skewed towards a particular part of the network. In EEODA, routing is not only based on the shortest cluster head to a particular one but on a combination of three parameters which includes, (i) shortest path (ii) the neighbour cluster head with lowest number of received data and (iii) the received signal strength to the cluster head. The clustering process in EEODA comprises of two phases, (i) formation of cluster and (ii) propagation of data. The formation of cluster phase is similarly made of two steps, they are start state and the formation of cluster state.

Start state phase is complete when a node can exist in any of three states, namely a sole node representing a single node in a cluster, a cluster head or a cluster member. Formation of cluster phase are formed in the following sequence. Each node first starts as a sole node, at this stage, a node will require high energy to transmit to a neighbour node. This can deplete the remaining battery energy of the node, in this case, the node will be alone in its cluster and becomes a forced cluster head. However, this scenario only plays out when a node does not have a neighbour in the range of 100 meters. The second scenario is for a node to be a cluster head, this occurs if the node has the highest level of Energy among its one-hop neighbour. The third and last scenario is for a node to be a cluster member. This happens when a node joins a neighbour node with a higher residual energy to it.

All these steps are determined using Q-learning protocol. After the cluster head aggregates sensed data in its cluster, the next phase is the transmission of packets to the UAVs. All cluster head has a register that stores the nodes in the cluster and a counter that keeps the amount of times it receives data from its neighbour cluster head. Using the Q-learning protocol, the cluster formation phase will be formed when all nodes have settled either as a lone node in a cluster, a cluster member or a cluster head. Details of the protocol are given in subsequent section 3. Q-learning clustering protocol proposed in this paper is suited for an ad-hoc network with no defined network infrastructure such as a base-station. The cluster heads in each cluster aggregate sensor data in its cluster and subsequently forward through a succession of neighbour cluster heads to the UAVs. To improve the position of the UAVs to the cluster head, a modified simulated annealing search method was used to reduce the distance of transmission from the cluster head to the UAV.

Simulated annealing search method was also modified to determine the number of UAVs needed for the network size such that the Number of UAVs, the cluster size, and the number of clusters in the network were optimized. Using the various UAV positions optimized for data collection using the multi-objective simulated annealing search method, a reinforcement learning technique using Q-learning was used to reduce the time required for the UAVs to locate an optimized 3D location for data collection from the cluster heads. The use of Q-learning algorithm helped to determine the optimized energy consumption of both UAVs and the cluster head for network routing. The rest of this paper is arranged in the following order. Section 2 reviews the literature. In section 3, the model of the system was developed, section 4 describes the operation of the EEODA algorithm. Section 5 describes the mobility model used for the movement of the UAV. Results and analysis of the experimental procedure were done in section 6, while the conclusion was given in section 7.

Related Works

In the work of Ali et al., (2021), they developed a model that shows that, by optimizing the speed of the UAV with a constant altitude, the rate at which the UAV will not acknowledge the transmitted data can be minimized. A tri-rotor UAV was used for their design. The shortcoming in their plan was that it is only applicable in a linear topology wireless sensor network with low sensor density. The forward and backward mobility technique utilized for the UAV in the paper is limited to the coverage area of the UAV radio. Liu and Zhu, (2019), designed three transmission modes: waiting mode, conventional transmission mode of the sensor node-sink, and the UAV-sensor node transmission. The transmission modes were used to reduce the energy consumption of the WSN. Dynamic programming was used to design an optimal policy to select the transmission mode to be used. This was done by using a recursive random search algorithm to optimize the UAV's trajectory. However, the shortcoming of their work is its limitation of the data collection to a static infrastructure scenario. In other words, it is not suitable for an infrastructure-less or mobile infrastructure scenario. EEODA protocol designed in this paper is particularly suited to an infrastructure-less network scenario.

In the work of Ebrahimi et al., (2018), a joint optimization problem was developed, using optimization of node's clustering and UAV trajectory in a high-density sensor network. The purpose of their design, was to lower the sensor's energy drainage when aggregating the data sensed by the sensors using a compressive data-gathering technique. The technique enabled them to lower the amount of transmissions needed to send data to the UAV. They formulated a forwarding tree from the cluster members to the cluster heads. The network was designed in a hierarchical form while sensed data for the sensor nodes were aggregated in each level of the hierarchy. The shortcoming in their design is the requirement that the compressed data was retransmitted along the network hierarchy before getting to the cluster head. The need for aggregating data from the various levels in the order will result in an increased energy depletion of the WSN. In EEODA algorithm, the aggregation of data was done by the cluster heads. In the work of Zhan et al., (2020), a joint optimization method was used for optimizing energy drain of the sensor nodes in a network.

The researchers designed the nodes to be in two states, the transmitting state and sleep state. To transition between the sleep states and the transmit state, a wake-up schedule was designed to optimize the timing of this transmission for an energy-efficient transmission to the UAV. The introduction of the wake-up procedure in their design resulted in reduced energy depletion in the nodes. However, the shortcoming is its limitation to an infrastructure-based network. The work of the researchers in Sun et al., (2017), they designed an optimization model for UAV trajectory involving multi-parameter constraints. Researchers in Hu et al., (2020), developed a UAV trajectory for WSN pesticide control using a solar energy harvesting technique for powering WSN to reduce the over-dependence on the sensor's battery power, thereby increasing the network lifetime of the WSN. The limitation of this work is its suitability for only cellular wireless communication and the inability to be deployed in an infrastructure-less network as proposed in this paper.

In the work of Say et al., (2016), a different priority-based algorithm for medium access control (MAC) in WSN was proposed. Their design aimed to lower the amount of duplicate transmissions. The selection process, which depends on the size of each transmission window used in the design, took advantage of the mobility feature of the UAV. In addition to the MAC protocol, a routing algorithm was proposed to reduce the time lag for transmission between the sensors and UAVs. The shortcoming in their design is the use of fixed-wing UAVs, which are not appropriate for data dissemination in a hilly and uneven network. Fixed-wing UAV needs to be flown at higher latitude compared to the rotary-wing UAV. This results in the sensor's data needing to travel longer distances to get to the UAV, thereby depleting their energy source. Researchers in Poudel and Moh, (2020), designed a protocol for UAV wireless sensor networks by deploying a quick and data aggregation method which results in low energy consumption in non-accessible networks; however, their design involved the use of fixed-wing UAV.

The design also suffers from the high altitude of the UAVs to the sensors, causing quick energy drainage and low life expectancy of the network. Researchers in Poudel and Moh, (2019) surveyed on MAC protocols suitable for UAV wireless sensor networks. The scope of this work is limited to an energy-efficient clustering algorithm, so the optimization of the medium access protocol wasn't considered. In the work of the researchers in Ho et al., (2015), particle swarm optimization was used to design a combination of an ideal WSN topology and trajectory of UAV required for reducing energy loss in WSN. A performance comparison analysis was done with the low energy adaptive clustering hierarchy (LEACH). The radio communication model in their design was modelled on a flat topology. However, the same radio model can be used with diverse types of topology.

This work combines of the work in Rezoan and Sangman, (2021) and the use of UAV to act as an aggregation point for receiving sensor data from the cluster heads instead of all sensors in the cluster as proposed in (Rezoan and Sangman, 2021). In the work of the researchers in Popescu et al., (2019), an experimental test bed was designed such that sensed data from a sensor was transmitted first to its closest base station. The base station acted as a tunnel used to aggregate the sensed data for onward transmission to the UAV. The shortcoming of the design was its high reliance on infrastructure. The design, though leads to a reduction in energy consumption for the nodes; it however requires an expensive network configuration. The work of the researchers in You and Zhang, (2019), they employed a technique to reduce the propagation signal fading. An obstacle-aware 3D trajectory was designed for communication between the UAV and WSN. The design led to optimum energy consumption of the nodes, though a flat topology was assumed.

The researchers in Mi et al., (2019) designed a WSN clustering protocol using K-means. The nodes were deployed in an uneven and random topology. The cluster head was elected based on the nodes within a region with the highest remaining battery energy and storage capacity. Exponential function based on fuzzy logic was used to get the value function for the combination of the remaining energy of a node sensor and its storage capacity. In the work of the researchers in Pang et al., (2014), they proposed a data aggregation scheme for UAV wireless sensor network, where the network area was divided into clusters. Cluster head was elected based on the remaining battery energy of a node and its frequency of data transmission within the group. Data aggregation scheme was designed for an optimal UAV trajectory using direct future prediction model. The shortcomings of the model in Mi et al., (2021) and Pang et al., (2014) was the fact that they were optimized for flat network topology.

2. MODEL OF THE SYSTEM

This section describes the assumptions, model for communication and the mobility model of the UAV. Lists for these assumptions are made separately for the application scenario, which involves UAV, WSN and MAC protocol. The communication model describes the network topology, radio communication, states of nodes and the network's overall protocol. The model for the mobility of the UAV is described at the end of this section.

Assumptions Made

The network topology is assumed to be uneven and hilly. This means that obstacles are expected in the network area. It is, however assumed that the transmission between the ground sensors and the UAV is not obstructed in any way. The model used in this paper is also applicable to a flat topology. The network is devoid of static infrastructure such as a base station or network control center. This is due to the assumption that the network is in an inaccessible area where these infrastructures are absent. The work of past researchers under this framework, assumed that the data aggregation nodes have high level of energy with a possible external energy source. This type of node is not realistic for the model scenario presented in this paper because it needs constant supervision. This will not possible due to its network inaccessible status. For this type of scenario, a UAV will be better suited as the data collection device, this is because it is mobile and can fly over many obstacles present in the network.

Wireless Sensor Network Assumption

The sensor nodes are assumed to be location-aware. This means that they are installed with a global positioning system module, which makes them capable of determining the location of nearby sensors and the UAVs (Hayes and Ali, 2016). Since the nodes are static, using the global positioning feature only once to determine the position of nearby nodes and UAVs is sufficient. This reduces the incremental energy loss due to the repeated use of the GPS functionality. The nodes are assumed to be static. This implies that their position does not change after deployment. This assumption is applicable to most network scenarios because wireless sensor nodes are not expected to migrate from their initial position in most instances. Many mobile nodes occur in the mobile ad-hoc systems, and this represents an insignificant percentage of the wireless network family. However, mobility-enabled sensors are reserved for future areas of deployment. The nodes in the network are assumed to be homogenous. This implies they all have the same processing speed, power, storage capabilities, and radio propagation characteristics.

The network is assumed to have two types of nodes, namely cluster members and cluster heads. The class of a node is determined through the clustering algorithm. As stated earlier, the nodes are equipped with a global positioning system; hence, they are assumed to be aware of the cluster they belong to. Furthermore, the positions of neighbour nodes and the initial positions of the Unmanned Aerial Vehicles (UAVs) are also assumed to be known in advance. It is also assumed that nodes within the cluster are a few hops from one another i.e., a maximum of three hops. Nodes are also aware of their direct neighbours (i.e., one hop away nodes) so that the number of hops to the cluster head within the cluster will be updated through the Q-values assigned to each node using the agent’s learning algorithm. Finally, nodes have no prior information on sensed data from other nodes in the network, i.e., no knowledge of data packets sensed by other nodes is known.

The UAV Assumption

The UAVs are assumed to have adequate power to be sustained throughout a round of data collection from the sensors. UAV’s power can be replenished after every round of data aggregation from its base. The assumption is justified since the UAV’s battery can be changed after each round of data aggregation as their mobility is not affected by the network environment's hilly, uneven, or hostile nature. It should be realized here that research is currently ongoing for a solar-powered UAV. However, this is beyond the scope of this work. UAV used in this work is the quadcopter; hence, it requires little bend angle to migrate from one position to another. It is also capable of prolonged stay in one position. The UAV is assumed to have adequate memory to aggregate all sensor readings from the network coverage area. The assumption is realistic because the UAV stores aggregated data only for a round of routing after transmitting all aggregated data to the network’s base station or network control center (NCC) for further processing.

At the end of every successful transmission to the network’s NCC, the sensed data in the UAV are purged to get it ready for another round of data aggregation from the sensors. The UAVs are assumed not to have obstacles. This is because they are equipped with many sensors, which enable them to communicate only through line of sight. All sensors experiencing obstacles will be excluded from aggregating data in that particular region (Joukhadar et al., 2019). In the UAV-WSN communication model proposed in this paper, a line-of-sight communication was assumed because the UAVs are expected to have many sensors wherein at least one of the sensors will have a line-of-sight communication with the target wireless sensor node. With the assumption that many UAVs are deployed in the network area. This is a very reasonable assumption. EEODA algorithm proposed assumed a high number of UAVs and the sensor nodes. Sensor nodes are divided into clusters, while the UAVs serve as the data collection center.

The use of many UAVs equipped with many sensors is to enables them to be positioned in such a way that data collection from the cluster heads will be energy-efficient and devoid of long-range transmissions even in the presence of obstacles. UAVs are equipped with embedded firmware capable of calculating the RSSI power. This enables each UAV to determine the cluster from which sensor readings will be aggregated. It was assumed that each UAV is capable of determining the lowest height of flight above the ground for energy-efficient radio transmission between it and the wireless sensor nodes using its embedded sonar or LDR sensor. This assumption is closely linked with the UAV capability of having line-of-sight connectivity with the wireless sensor nodes using its many sensors.

MAC Protocol

The EEODA protocol employs carrier sense multiple access with collision avoidance (CSMA/CA) to transmit the hello packet to the wireless sensor nodes. Time division multiple access (TDMA) was used for data packet transmission between the wireless sensor nodes (cluster heads) and the UAVs. The multiple access control scheme used in this paper is taken from that used in the article by researchers in (Ho et al., 2018). However, this paper assumed that all sensor nodes had equal processing abilities, which is at variance with the different priority values given to nodes in their report.

Communication Model

The network area assumed in this paper is composed of hilly and uneven surfaces, and as a result of this, the sensor nodes will take different heights from one another. The nodes are scattered randomly in the network area. The network is infrastructure-less i.e., without network control centre or base station. It can therefore be regarded as an ad-hoc network comprising sensor nodes and the UAVs acting as the data collection centre, which transmits the aggregated data from the cluster heads to the processing center. It should be stated here that the protocol proposed in this paper will equally be applicable in other types of infrastructure-less networks as well as in most network topology. The set of UAV nodes that covers the network area is modeled as H number of nodes where H = 1,2,3,…..H, each node has a three-dimensional coordinate sensing agent (m1, m2, m3,…….mH), to cater or the uneven topology resulting in different heights of the wireless sensor nodes within the network area.

The EEODA Communication System

The EEODA communication system comprises five stages: initialization, identification of neighbour nodes, routing, data aggregation, and forwarding aggregated data to the sinks (UAVs). In the initialization stage, the wireless sensor nodes within the network area are divided into clusters using the EEODA Q-learning clustering algorithm. The second stage involves each node identifying its neighbours. This stage is necessary for the formation of clusters with the eventual election of cluster heads (CHs) within each cluster. This is achieved through the geographical positioning system embedded in each of the sensors. GPS is capable of measuring the RSSI values from nodes to determine the nodes closest to it. The third stage is the election of cluster heads.

To achieve this, a modified simulated annealing search method was used to derive an optimum position of the cluster heads to the UAV such that energy consumption due to radio communication between the cluster heads (CHs) and the UAVs is minimized. The cluster heads are used as data aggregating agents for transmitting the combined sensed data from the sensor nodes in each cluster to the UAVs. The third stage is routing, through which all nodes in a cluster determine the path through which their sensed data would be sent to the UAVs through a cluster head. The routing protocol is embedded in the Q-learning clustering algorithm described in section 4.3. The fourth stage is the forwarding of the data sensed by each wireless sensor using the path selected in stage three to the elected cluster heads, while the fifth and the last stage is forwarding of the aggregated sensed data from the clusters to the UAVs which acts as the collection points of the aggregated data for onward transmission to the processing center.

At the start of network formation, the UAVs are unaware of the cluster heads position, which is at variance with the assumptions in most routing methodologies used in existing literature. It should be stated that the modified simulated annealing search method is used to fine-tune the position of the UAVs in relation to the cluster heads in the entire network. Simulated annealing heuristics were used in comparison with the metallurgical operation of calculated annealing temperature to get an alloy in its most stable state. it was used in this paper to get the best position of the UAVs to the cluster heads the details of the modified simulated annealing algorithm is shown is section IV.I.

IV Cluster Head Election using Q-learning

In electing cluster heads, each node attempts to route sensed data directly to a nearby UAV, thereby acting as a cluster head for sending the data to better suited neighbour node which requires less energy and acting as a cluster member. In order to make this decision, the UAVs send HELLO packets (DATA REQUEST) to the wireless sensor nodes.

Algorithm 1 EEODA for electing cluster heads

Input: {L sen L1 sen L2 sen L3 sen L4 sen…….. Ln sen: denotes location of 1 hop neighbour to sensor L

Output: Li selected as Cluster head

Initialization

1. L sen ← {one hop neighbour list}
2. Hello broadcast packet to all nodes ∈ L sen
3. Divide network into virtual grid of 25m2
4. Each sensors in cluster replies UAV with RSSI value
5. RSSI value for each sensor in a cluster stored in a flag in its data packet
6. Each sensor in cluster compares its RSSI value with all one-hop neighbour
7. Initializes cluster head counter J = 1
8. If RSSI value <= to all RSSI of one hop neighbour
9. Assigns CHj to L sen
10. else

11. Assigns CH to sen i
12. increment J = J + 1
13. do: is CHi != CHfinal
14. while (True)
15. goto 4
16. end do.. while
17. end

Figure 1 Pseudocode for EEODA clustering algorithm

The sensor nodes in each cluster measure its RSSI value and compare it to that of its neighbour. If its own RSSI value is less than that of its neighbour, it acts as the cluster head, otherwise the neighbour node acts as the cluster head. This operation let all nodes to know the different paths data can be routed to the UAVs. The pseudocode for the clustering protocol is shown in (Figure 1). From Figure 1, it can be seen that a wireless sensor node will make the decision that leads to reduced energy radio consumption to act either as a cluster member or cluster head.

The Modified Simulated Annealing algorithm

Now, after the Q-learning protocols determine the number of clusters and the cluster heads in the network, the modified simulated annealing protocol is used to determine the optimum cluster head position for the UAVs. To do this, the algorithm initializes the cluster head position from the Q-learning as the initial position from which the UAVs receive aggregated data. The modified simulated annealing algorithm aims to compute an optimum position for the UAVs, leading to minimum energy consumption for the cluster heads in transmitting aggregated data packets to the UAVs. The starting point location of both the cluster head and the UAV serves as the current solution and required temperature. Continuous iteration is performed to select another location for the UAV such that the energy for radio communication will be minimized.

The energy required for radio communication between the UAV at its new position and the fixed cluster heads is then applied to an energy evaluation procedure to determine if the new position will be used to replace the earlier position. The modified simulated annealing protocol used in this paper comprises two nested loops. The inner loop includes computing an alternative position for the UAVs in succession to get an optimum solution to the algorithm. This iteration is performed until a better position with respect to the Received Signal Strength Indicator, which indicates the energy consumption of radio communication between the UAV and the cluster head, cannot be achieved.

The outer loop comprises adjusting the distance of the UAV’s new position to that of its previous position to be within a certain range so as not to alter the equilibrium of the protocol. The loop is, controlled by an equilibrium value that determines the maximum number of iterations to be performed on each UAV. The modified simulated annealing algorithm is shown in (Figure 2). From the aforementioned, two factors that determine the modified simulated annealing protocol. These are generating a new position for the UAV and controlling the distance between successive generations to be within a certain range. It should be stated here that the outer loop controls the inner loop, and it represents a set of conceptual methodologies aimed at controlling variation in the UAV’s mobility for generating alternative solutions.

The Mobility Model for UAV

The City Section mobility model was used in controlling and computing the UAVs' mobility to generate a new position used in the modified simulated annealing algorithm shown in (Figure 2). In this model, the network is modeled as a grid in which the UAVs are initially stationed at the corners of the network. The UAV starts the search at its initial position while it randomly selects the destination, which is a cluster head in the network. UAVs send a HELLO packet to the wireless sensor network nodes, to which the cluster heads report their identity through an identity flag in the data packet replied to the UAV. City section mobility model is used to iteratively move the UAV until the best received signal strength indicator (RSSI) is received, after which the iteration stops, and another UAV is selected for similar operation to other cluster heads in the network. The operation is repeated until all cluster heads in the network have been successfully covered.

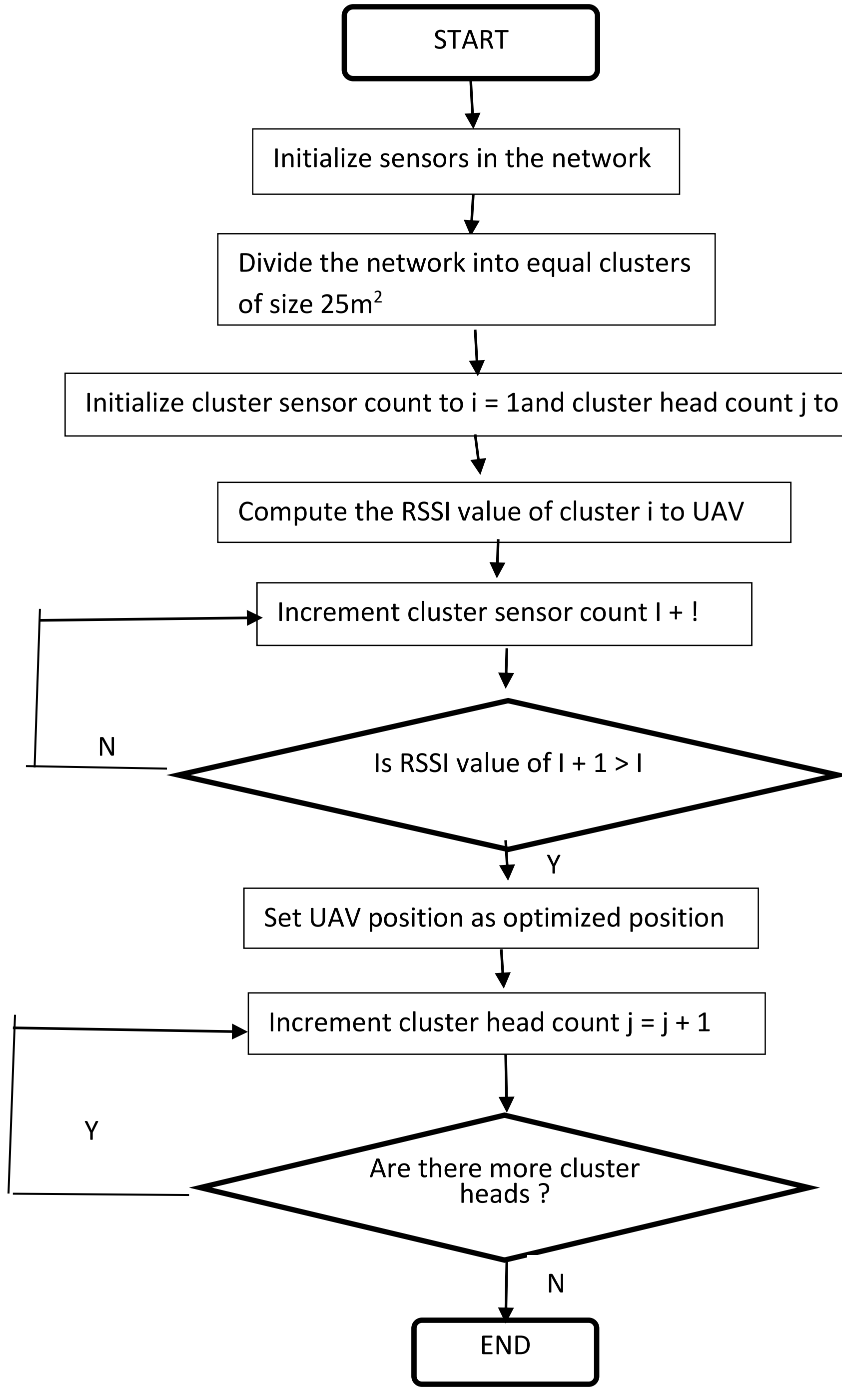

Figure 2 Flowchart for the Modified Simulated Annealing Algorithm

The UAV commences the search for the optimum RSSI from an initial position denoted by

$$\text{UAV commence} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (1)$$

Where UAV commence denotes the initial position of the UAV. The UAV then follows the city section mobility model to compute its new 3D position for the UAV. The last position of the UAV after the complete iteration is denoted by

$$\text{UAV last} = \begin{pmatrix} Fn_x \\ Fn_y \\ Fn_z \end{pmatrix} \quad (2)$$

Where UAV last represents the UAV's final position, while Fnx, Fny and Fnz represents the optimal 3D coordinate of the UAV along the region of interest (ROI). From the mobility model used in this paper, the values of Fnx, Fny and Fnz determine the limit of mobility used for the iteration in the modified simulated annealing protocol. For each UAV mobility during the simulation, the relative value of Ty (change in distance along the y-axis) with respect to TX (change in distance along the x-axis) is computed as shown in Equation 3.

$$TY = \begin{cases} T_Y \text{ Sensors} * 1.5, & T_{y\ \text{Sensors}} \leq T_z * 0.9 \\ T_Y \text{ UAV} * 1.5, & T_{y\ \text{UAV}} \leq T_{z\ \text{sensors}} * 1.1 \end{cases} \quad (3)$$

Where TY Sensors and Tz sensors represents the movement of the UAV along the y-axis and z-axis respectively, with respect to the cluster heads in each cluster used for the determination of the RSSI value. The position of the Ty and Tz sensors is fixed as the sensors in the network are assumed to be in a fixed location. The position of the Ty sensors can be derived from Equation 4,

$$\text{Tysensors} = \sqrt{\delta_{sens}^2 - T_Z^2} \quad (4)$$

Where δ_{sens}^2 represents the transmission range of the sensor. Tz can be computed from Equation 5

$$\text{Tz} = \sqrt[2]{(T_z - T_x)^2)} + \sqrt[2]{(T_z - T_y)^2} \quad (5)$$

where Tx, Ty, and Tz represents the 3D coordinate of the cluster heads.

The range of the broadcast of the hello packets from the UAV is such that each UAV's transmission range should cover all wireless sensor from which it acts as the data collection agents for a minimum of one round of transmission. In order to fulfil this, the time of the hello broadcast is as shown in equation 6

$$\text{dbef} \leq \text{daft} \leq (\text{dbef} + (\frac{UAV_W^{T_{y/3}}}{LV_{UAV}})) \quad (6)$$

where dbef represents the time the previous hello control packet was broadcast while daft represents the time the following hello control packet will be transmitted. $UAV_W^{T_{y/3}}$ Represents the transmission range in which the UAV will travel on level ground after one-third the distance of the Ty with respect to the y-axis while LV_{UAV} is the simulation speed of the UAV. Parameter $\frac{UAV_W^{T_{y/3}}}{LV_{UAV}}$ represents the range of time that a wireless sensor node will be within the transmission radius of the UAV. It should be noted here that the distance of the sensor nodes is one-third of Ty from the city section mobility model. The distance $T_{y/3}$ given in Equation 6 represents the farthest distance of a node from the UAV. For the hello control packet broadcast from the UAVs to get to all sensor nodes in their domain, the expression in Equation 6 should be met.

Identifying UAV Optimum Data Aggregating Position

In the EEODA protocol, the UAVs are placed at a specified altitude above the center of each cluster. It should be noticed here that the Q-learning algorithm determines the position of the cluster head, and it is fixed. City section mobility model is used in iterating the positions of the UAVs with respect to the cluster heads to arrive at an optimum UAV position for forwarding the aggregated sensor data to the processing centre.

Algorithm 2 to locate the optimal position of UAVs to the cluster heads

Output: {U1, U2, U3……………. Um | contains the suboptimal data gathering positions}

Initialization:

1. broadcast_U_{RSSI}^i search_message ()
2. if (receive (CH info))
3. $U_{RSSI(X)}^i$ ← initial x-axis position of UAV
4. $U_{RSSI(y)}^i$ ← initial y-axis position of UAV
5. $U_{RSSI(z)}^i$ ← initial z-axis position of UAV
6. CHx ← discovered cluster head x-axis value
6. CHy ← discovered cluster head y-axis value
7 Ux,y,z = RSSImax // Initialize suboptimal UAV position as optimal and final position
8. Ui = CH_{dist}^j // Initial position of UAV i.e. at a distance CHdist from CH

Iteration:

9. while CHj != {φ} // where there still exist cluster head
10. while (U_{RSSI}^i >= RSSIthreshold): // controls the number of iteration
11. i = i + 1

12. If $U^{i+1}_{RSSI(X)}$ >= $U^{i}_{RSSI(x)}$ x axis geo-position of UAV & $U^{i+1}_{RSSI(y)}$ >= $U^{i}_{RSSI(y)}$ y axis geo-position of UAV & $U^{i+1}_{RSSI(z)}$ >= $U^{i}_{RSSI(z)}$ z axis geo-position of UAV
13. U^{i+1}_{RSSI} = RSSImaxUV
14. end if
15. end while(10)
16. j = j + 1
17. end while (9)

Figure 3 Pseudocode for Obtaining Optimum UAV Position

Locating the Optimum position of UAVs to the cluster Heads

The city section mobility model is used to iterate the position of the UAVs until an optimum position for the UAV to each of the cluster heads is received. Modified simulated annealing algorithm were used to compute the optimum position of the UAVs. The pseudocode for iterating the positions of the UAVs is shown in (Figure 3). From the pseudocode, $U^{i}_{RSSI(X)}$, $U^{i}_{RSSI(y)}$ and $U^{i}_{RSSI(z)}$ denote the initial position of the UAV in the x, y and z axes respectively. The RSSI value of the cluster head sensor node to the UAV is measured until the best signal strength is received. However, a limit was placed on the number of iterations possible by using an equilibrium value of RSSI from the modified simulated annealing algorithm.

Once the difference between this value and the RSSI is within a threshold value, the iteration is stopped. UAVs broadcast a hello packet to the wireless sensors in the network. When a cluster head response is received (Two flags in the hello packet indicate the identity of a wireless sensor node as a cluster node or a cluster member while another flag denotes the node's geographic position), it begins its iterated search for an optimum RSSI value to each of the cluster heads. The pseudocode in Figure 3 contains a nested loop. The inner loop finds the optimum path to each cluster head, while the outer loop was used to iterate the entire cluster heads in the network. The range of the hello packet transmission is given in Equation 6.

Description of Algorithm Runtime

The runtime complexity of the pseudocode in Figure 3 is dependent on that of (Figure 1). The cluster head search function in Figure 3 is invoked in (Figure 1). Figure 3's running time is controlled by the threshold value of the RSSI denoted by RSSI threshold. Once the difference between the RSSI value of the wireless sensor cluster head is less than the threshold value, the iteration will stop, and the program continues with the next cluster head until all the cluster heads in the network have been exhausted. The modified simulation annealing search technique was used iteratively to improve the data-gathering position of the UAV. The equilibrium position is fixed at a threshold value from a limiting RSSI value.

This is shown on line 10 of (Figure 3). The essence of this equilibrium is to fix an upper bound to the search of an optimum position to the UAV. When this condition is met, the iteration is terminated. To effectively use this limiting value, it is imperative to define the number of sensors in each network cluster. To compute how the UAV's position is progressively iterated in 3D geo-location, to arrive at an optimum poinr for certain number of clusters in the network, (the number of clusters used in this paper was 3). UAV uses the lower and upper boundary positions for every three contiguous clusters in the network. This computation is necessary so that UAVs do not overlap the cluster boundaries. The expression for computing the iterated position is shown in Equation 7

$$\begin{pmatrix} U_{uppX} \\ U_{uppY} \\ U_{uppZ} \end{pmatrix} - \begin{pmatrix} U_{lowX} \\ U_{lowY} \\ U_{lowZ} \end{pmatrix} = \begin{pmatrix} U_{rangeX} \\ U_{rangeY} \\ U_{rangeZ} \end{pmatrix} \qquad (7)$$

Where UrangeX, UrangeY, and UrangeZ denote the limits to the range in which the UAV can move to get an optimum position for aggregated data delivery to the processing centre. UuppX, UuppY and UuppZ denote the upper boundary for the selected three clusters i.e., upper boundary to the right, while UlowX, UlowY and UlowZ denotes the lower boundary of the three clusters i.e., lower boundary to the left. The mobility for each step in the iteration of the UAV (to arrive at the optimum position) is calculated using Equation 8.

$$\begin{pmatrix} U_{rangeX} \\ U_{rangeY} \\ U_{rangeZ} \end{pmatrix} + \begin{pmatrix} \beta_x \\ \beta_y \\ \beta_z \end{pmatrix} z = \begin{pmatrix} Iter_X \\ Iter_Y \\ Iter_Z \end{pmatrix} \qquad (8)$$

Where βX, βY and βZ denote the fraction of the range that would be used to iteratively move the UAV in search of the optimum position, as shown in (Figure 3). The limit of iteration is dependent on βX, βY, and βZ values. If the value of β were increased, the number of iterations will be lower and vice-versa, however a high value of β may result in suboptimal value. The value of

β used in the simulation was 12.5% of the range. The computation of an optimal value for β is, however beyond the scope of this paper.

To compute the next iteration point, the step sizes shown in Equation 7 and Equation 8 are added and subtracted as delivered in the respective equations. The need for both addition and subtraction are for the UAV to explore the 3D coordinates of the x, y, and z axis, respectively. The total number of combinations for the position of the UAV from the range used in the simulation is 27. However, this has been reduced to 8 after using the expression given in Equation 11. Careful observation of the 27 points in the combination will show that the excess number (27-8) is just duplicate of the eight positions. This is because the RSSI values measured from these bloated points are redundant values as they are duplicates of the eight positions computed from Equation 8.

Algorithm 3 to locate alternate positions of UAV

Output: {U1, U2, U3……………. Um ǀ contains the suboptimal data gathering positions}

Initialization:

1. Broadcast_U^{i}_{RSSI} search message()
2. do while CH infor != {Φ}
3. CHcounter ← 0
4. if (receive (CH info))
5. $U^{i}_{RSSI(X)}$ ← initial x-axis position of UAV
6. $U^{i}_{RSSI(y)}$ ← initial y-axis position of UAV
7. $U^{i}_{RSSI(z)}$ ← initial z-axis position of UAV
8. CHx ← discovered cluster head x-axis value
9. CHy ← discovered cluster head y-axis value
10 Ux,y,z = RSSImax // Initialize suboptimal UAV position as optimal and final position
11. Ui $= CH^{j}_{dist}$ // Initial position of UAV i.e. at a distance CHdist from CH
12. end while (2)

// Iteration to locate three cluster heads with highest RSSI

13. do while CHj != {φ} // where there still exist cluster head
14. while (U^{i}_{RSSI} >= RSSIthreshold): // controls the number of iteration
15. i = i + 1
16. If $U^{i+1}_{RSSI(X)}$ >= $U^{i}_{RSSI(x)}$ x axis geo-position of UAV & $U^{i+1}_{RSSI(y)}$ >= $U^{i}_{RSSI(y)}$ y axis geo-position of UAV & $U^{i+1}_{RSSI(z)}$ >= $U^{i}_{RSSI(z)}$ z axis geo-position of UAV
17. U^{i+1}_{RSSI} = RSSImaxUV // Cluster head with hghest RSSI value
18. Set I = I – 1
19. while CHj != {φ} // where there still exist cluster head
20. If $U^{i}_{RSSI(X)}$ >= $U^{i-1}_{RSSI(x)}$ x axis geo-position of UAV & $U^{i+1}_{RSSI(y)}$ >= $U^{i}_{RSSI(y)}$ y axis geo-position of UAV & $U^{i+1}_{RSSI(z)}$ >= $U^{i}_{RSSI(z)}$ z axis geo-position of UAV
21.. U^{i}_{RSSI} = RSSImaxUV // cluster head with second highest RSSI value
22. Set i = i – 2
23. while CHj != {φ} // where there still exist cluster head
24. If $U^{i}_{RSSI(X)}$ >= $U^{i-1}_{RSSI(x)}$ x axis geo-position of UAV & $U^{i+1}_{RSSI(y)}$ >= $U^{i}_{RSSI(y)}$ y axis geo-position of UAV & $U^{i+1}_{RSSI(z)}$ >= $U^{i}_{RSSI(z)}$ z axis geo-position of UAV
25. U^{i}_{RSSI} = RSSImaxUV // cluster head with third highest RSSI value
26. end if
27. end while(22)
28. end while (18)
29. Ui $= CH^{j}_{dist}$ // Initial distance to the three cluster heads
30. RSSI sum $= U^{i+1}_{RSSI} + U^{i}_{RSSI} + U^{i-1}_{RSSI}$
31. do while I <= 7
32. i = 1
33. Ug(i) $= CH^{j}_{dist}$
34. i = i + 1 // Iterate using city section mobility
35. If Ug(I + 1) > Ug(i)

36. Ug(i + 1) = CH_{dist}^{j}
37. end while (32)
38. end while (13)
39 end while (12)

Figure 4 Pseudocode for computing alternative positions of UAV

To illustrate this point, it was assumed that the initial position of the UAV is given by Ux, Uy, and Uz, in contrast, the optimum position was assumed to be within the 27 points, which can be computed from Ux + IterX, Uy + IterY and Uz + IterZ. From the assumption stated earlier, the UAV is made to explore both positive and negative directions of x, y, and z axis, respectively. With this assumption, the first iterated step will be computed as Ux + IterX, Uy + IterY and Uz + IterZ. After this, the UAV may then explore the negative direction calculated as Ux - IterX, Uy - IterY and Uz - IterZ. Now it is important that the UAV is not made to iterate to the same position twice. This is made possible by the use of Equation 9

$$\begin{pmatrix} U_x \\ U_y \\ U_z \end{pmatrix} + \begin{pmatrix} Iter_x \\ Iter_y \\ Iter_z \end{pmatrix} = \begin{pmatrix} U_x^h \\ U_y^h \\ U_z^h \end{pmatrix} \qquad (9)$$

Where, U_x^h, U_y^h and U_z^h denotes the three likely positions the UAV may be iterated to from its previous position of Ux, Uy, and Uz. The alternative position can be computed using Equation 10.

$$\begin{pmatrix} U_x \\ U_y \\ U_z \end{pmatrix} - \begin{pmatrix} Iter_x \\ Iter_y \\ Iter_z \end{pmatrix} = \begin{pmatrix} U_x^l \\ U_y^l \\ U_z^l \end{pmatrix} \qquad (10)$$

Where U_x^l, U_y^l, and U_z^l denotes the alternative three likely positions the UAV will be iterated to. From these equations, the total number of possible iterated locations can be obtained by multiplying the two matrices shown in Equations 9 and 10. This multiplication leads to Equation 11.

$$\begin{pmatrix} U_x^h \\ U_y^h \\ U_z^h \end{pmatrix} \text{X} \begin{pmatrix} U_x^l \\ U_y^l \\ U_z^l \end{pmatrix} = \begin{pmatrix} U_x^h\, U_x^l \\ U_y^h\, U_y^l \\ U_z^h\, U_z^l \end{pmatrix} \qquad (11)$$

The matrix above Equation 11 will be used in Figure 4 to derive how to get the alternative positions for the UAV in the search of getting an optimum position for it. The purpose of Figure 4 is to get the number of iterations necessary to move the UAVs to their optimum position. From Figure 4, the best position for the UAV is selected for every iteration based on an objective function. The objective function here is to reduce the number of iterations required to get the UAV at its optimum position.

Computing the Alternative Position for UAV Data Forwarding

The objective function employed by the Q-learning algorithm is to use the value of RSSI of the cluster head to the UAV in progressively iterating the UAV's position. Using city section mobility model, the UAV's position is moved iteratively toward getting an optimum position for it. At each step, the modified simulated annealing search method is used to obtain the UAV's next position based on its RSSI value from the cluster head. At each point the reward value signifying the RSSI value from the cluster heads (which in this case is 3) is compared to the previous value. If the current value is higher, it updates the UAV position with the current position, while if otherwise, keeps the previous RSSI value corresponding to the previous UAV position. This procedure is continued until an optimum position for the UAV is obtained. The operation is then repeated until all the clusters have been utilized. To compute the reward, which in this case is the received signal strength indicator value, the expression in Equation 12 was used.

$$Sr = St * \left(\frac{1}{\zeta}\right) \mu` \qquad (12)$$

Where Sr denotes the strength of the received power of the UAV, St is the strength of the transmitted power of the wireless sensor. ζ represents the distance between the UAV and the cluster head, and μ denotes the path loss exponent. The range of the value of μ is 1 to 8 (Miranda et al., 2013). The authors in Tang et al., (2019) performed a test-bed experiment, it was found out that when using the near-ground communication model, the range of μ is between 2.5 and 3.45 in an obstacle-laden outdoor environment. CC2420 transceiver was used by the author in Tang et al., (2019) to carry out the propagation experiment. In this paper, it was assumed that the value of μ was 2.5. This can be attributed to the fact the likelihood of line-of-sight communication between the UAV and the wireless sensor nodes is high. The RSSI value of the cluster heads in the network will be sensed by the UAV's sensor, using Figure 4 to compute its value. Now, the power content of the signal is computed by taking the logarithm of both sides of the equation. This will result in the expression in Equation 13.

$20\log Sr = 20\log St - 20\mu \log \zeta$ (13)

From Equation 13, the value of Sr is expressed in dB to obtain the power content of the received signal. This value is denoted as the received signal strength indicator (RSSI). The path loss is given by 20μ log ζ. Equation 13 can be rewritten as shown in Equation 14

$RSSI = St - Sloss\ (\mu)$ in dBm (14)

Where Sloss represents the path loss given in dBm. Equation 15 gives the log-distance path loss

$Sloss\ (\mu) = S\ loss\ (\mu o) + 20\zeta \log\left(\frac{\mu}{\mu_o}\right)$ (15)

Here, Sloss (μ) denotes the path loss with the distance between the UAV and the cluster head given as μ. The path loss at a reference distance μo is represented as Sloss (μo). If the value of Sloss (μ) is replaced, Equation (18) can be rewritten as shown in Equation 16

$\left(S_t - (S_{loss}\ \mu_o\ ) + 20\zeta \log\left(\frac{\mu}{\mu_o}\right)\right)$ (16)

μo normally represents a unit of distance. If the value of μo in Equation 19 is updated, the expression shown in Equation 17 is obtained.

$RSSI = S_t - Sloss\ (\mu o) - 20\zeta \log(\mu)$ (17)

From Equation 17, the perceived power from a receiver (UAV) at reference distance μ_o is given by Equation 18

$G = S_t - Sloss\ (\mu o)$ (18)

Here, G represents the perceived power from reference distance μo. From here, the equation for RSSI can then be written as shown in Equation 19.

$RSSI = G - 20\zeta \log(\mu)$ (19)

By increasing the distance between the transmitter and the receiver, the value $Sloss\ (\mu)$ will increase while the value of RSSI decreases. This shows that RSSI is dependent on the 3D location of the UAV. Now, using the energy model in (Nayak and Devulapalli, 2019), the energy consumption of the wireless sensor thus depends on its distance from the UAV. From the aforementioned, the UAV's objective function for optimum search position is shown in Equation 20

$F(u,v,w) = \max\ (\frac{2}{RSSI_{sH}} \sum_{H=1}^{G_k} RSSI_H + (2 - \frac{\gamma_D}{Y_{DH}})))$ (20)

Where γ_D represents the standard deviation for the node's energy consumption in the cluster, while Gk represents the cluster's node size. RSSIH represents the value of the received signal strength indicator from each of the cluster head to the UAV in its horizon. The constraints used in determining the UAV's starting position shown in Equation 21 is represented by Equations 22, 23 and 24.

$RSSIH \leq \beta RSSI$ (21)

$Gminx \leq UAVx \leq Gmaxx$ (22)

$Gminy \leq UAVy \leq Gmaxy$ (23)

$FRUAV \leq \theta UAV \leq DRUAV$ (24)

The first constraint in Equation 24 illustrates that the position of the UAV at any instant must be greater than or equal to the threshold value RSSIH. This also ensures that the constraint $\gamma_D$should be less or equal to Y_{DH}. The expression for RSSIH is shown in Equation 25.

$RSSIH = \min\ (RSSI_{ini}^{G_k})$ (25)

where $RSSI_{ini,}^{G_k}$ denotes the possible position for the starting location for the RSSI value in each cluster. Constraint (25) shows that the UAV's coordinate on the x-axis cannot be outside the x-axis boundary of the assigned cluster denoted by Gminx and Gmaxx. Consequently, constraint (23) shows that the UAV's coordinate on the y-axis cannot be outside the x- axis boundary of the assigned cluster denoted by Gminy and Gmayy. Lastly, constraint (24) is to ensure that θUAV must be in the range FRUAV and DRUAV. FRUAV denotes the lowest possible height of the UAV above the ground. This is determined by the UAV's internal height sensor.

To derive the objective function for locating the various locations of the UAV, the algorithm is divided into two parts. The first position is to determine the position of the UAV for which its RSSI value from the cluster head is maximum. Secondly, the determined position would make the sum of the RSSI from each cluster member increase progressively. The standard deviation Gk, which denotes the summation of energy consumptions of the nodes in the cluster, should be minimized. This is because since all nodes except the cluster heads in the network are responsible for data transfer to the UAVs, the energy of all nodes will be nearly the same except the cluster head. UAV measures the RSSI value from the cluster heads in the network using Equation 26. By maximizing RSSI, the consumed energy of the nodes will be reduced. Also, minimizing the value of γ_D will make for an even distribution of energy consumptions from all the nodes in the cluster. The values for RSSI can be computed using Equation 26 below.

$$\sum_{H=1}^{G_k} RSSI_H = \sum_{H=1}^{G_k}(\mathrm{D} - 20\zeta \log(\mu)) \quad (26)$$

As stated earlier, ζ denotes the distance between the UAV and the concerned cluster head. From the EEODA protocol, the RSSI value was measured from the UAVs to the cluster heads. In other words, it is possible to express ζ in terms of Euclidian distance between the UAV and the cluster heads, this is an improvement over the latest researcher's work in Rezoan and Sangman, (2021) where the Euclidian distance of the UAV was measured from all the wireless sensor nodes in the cluster. It is argued here that the method in Rezoan and Sangman, (2021) would lead to redundant reading, which would result in quick energy depletion of the nodes due to the constant transmissions from all the nodes. In the algorithm proposed here, only the cluster heads will need to communicate with the UAVs, so that the energy of the other nodes in the cluster can be reserved if they may be used as a future cluster head. Overall, this will lead to a prolonged energy lifetime for the wireless sensor nodes and subsequently increased lifetime for the network. The Euclidian distance of the UAV from the cluster heads is given in Equation 27

$$\sum_{H=1}^{G_k}(D - 20\zeta log\left(\frac{\sqrt{(x_H = UAV_x) + y_H = UAV_G\ ) + \ z_H = UAV_z}}{\alpha_0}\right) \quad (27)$$

Forwarding Aggregated Data to Cluster Head

In the EEODA communication system, the fifth and last stage is the forwarding aggregated data from all the sensors in the cluster by their respective cluster heads to the UAVs. However, before the UAVs can receive the data, it will first calculate the shortest path to all the cluster heads in the network. In the assumption given earlier, the UAV will first transmit a HELLO packet to all the nodes in the network. The nodes will then return acknowledgment packets to the UAV. The data packets from the wireless sensor network have a bit dedicated to identifying the status of the node. This bit is referred to as the status bit in the sensor's data packet. This bit is 0 when a sensor is a cluster member and 1 when it is a cluster head. Now, the UAVs aim is to calculate the shortest distance from it to all the cluster heads in the network. This stage is necessary so the UAV can select the three cluster heads with the shortest distance to itself. Dijkstra's algorithm is used to calculate the shortest distance from the UAVs to the cluster head.

The Dijkstra's algorithm is modelled as follows:

Assume F = (A, B) to be a weighted graph, where the weight h: B→R is a function that maps the edges of the graph (displacement between two points or edges in the diagram) to a real value weight. Now, if d = (g, h), then the displacement between edges g and h represented by e(g, h) can be rewritten as h(d). The displacement of a path in the network given as ή = (s1, s2, s3, ……….. sm) is the addition of the displacements of the direct neighbour in the network. The displacement ή is given by Equation 28.

Displacement (ή) = $\sum_{i=1}^{m} h(s_{i-1,} s_i)$ (28)

The displacement between g and h represented by φ (g, h) is the distance of the least path length if such path exists. If not the distance is assumed to be infinite.

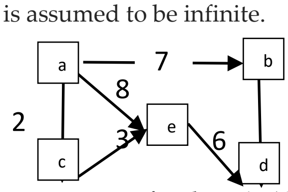

Figure 5 Description of Dijkstra's Algorithm

In the first place, it should be realized that a sub-path to a shortest path must be a subset to a larger shortest one. For example, in the digraph on Figure 5, path (a, c, e, d) is the shortest path between a and d, this also makes the path (a, c, e), the shortest path between a and e. Therefore, by extension, one can identify the presence of a path in an undirected graph with the least between any two nodes.

The Dijkstra's algorithm modified for the work in this paper can be explained as follows:

1. The UAVs receive the displacement value for all the cluster heads in the network for a given round. This is then sorted in ascending order.
2. A tree of the shortest path from the UAV to each of the cluster head is constructed. This is done by increasing the edge value by one after traversing to a new cluster head.
3. Compute the displacement represented by h(d) for the shortest distance (e (g, h)) from the UAV to each of the cluster heads.
4. The displacement h(d) ≥ φ (g, h) where h(d) is the displacement along the path between g and h, and if such path does not exist, the displacement is given as infinite

5. At the start of the algorithm, h (s) = 0, signifies that the length of a vertex to itself is zero, while the rest h (d) to all other cluster heads is set to infinite (∞). The actual length is then computed according to the value in the displacement bit of the control packet. The problem here is to find a single–source shortest path from the UAV to a cluster head.

6. To solve this problem, an undirected graph with positive edge weight is constructed. i.e., F = (A, B) with a known source node n i.e., the UAV. Where n Є A. it is expected to find the shortest path from the UAV to all the cluster heads in the network. Note the path to the cluster head will involve member nodes as well. It is then required to design an effective algorithm to achieve this aim.

7. Now, going back to Figure 5, to compute the shortest path to vertex d i.e., (a, c, e, d), it is required to compute h(d) for every d Є neigh [e] i.e. (a, c, e). From the afore-mentioned, the shortest path to vertex d is computed by relaxing the shortest path to vertex e. Therefore, h(d) = h (e) + a new edge φ (e, d)

8. To construct a new shortest path from a previous one, again looking at Figure 5, If we consider edge e to d, which is six units, i.e., the weight of the edge w (e, d) = 6, now assuming the shortest path to e is already known as shown by path (a, c, e), now d (e) is represented by h(e). Therefore, when the path from a to e is added to the edge (e, d) another path from a to d is obtained, which have a length of h (e) + w (e, d)

9. Now, if h (e) + w (e, d) < h (d), the new path is then used to replace the old one (a, e, d)

The algorithm for updating the shortest path to a vertex e to get another shortest path to d if d is a neighbour vertex to e is shown in (Figure 6).

```
Update (a, d)
{
If(h[e ] + w(e, d) < h[ e] )
     {   h[d] = h[e ]  +  w( e, d) ;
            neigh[d]  = e;
     }
}
```

Figure 6 Update Path in Dijkstra's Algorithm

Where neigh [d] = e depicts that vertex e is a neighbour to vertex d. This update function can only be activated when this condition on line 1 of figure 6 is met.

The Dijkstra algorithm used in this paper is depicted in Figure 7.

```
Dijkstra (T, a, d)
{
For (each a  Є  V)
    {
         h[b]  =  ∞;
    }
     h[a] =  0;
     neigh[a] =  ∅;
W = Sort all vertices
while (non-empty(W))                                        //   Arrange vertices in ascending order
{
      G  =  Find-Min(w)
      For (every e Є  neigh[d]
            If (h[e]  +   w  (e, d)  <   h ( e)
           {
                h[d]  =  h[ e]  +   w ( e, d);
                 pred [d]  =  e;
            }
}
```

Figure 7 Dijkstra Algorithm

3. EVALUATING THE PERFORMANCE OF EEODA

The Simulation Environment

To evaluate the performance of the EEODA algorithm proposed in this paper, the MATLAB simulation environment was used. The simulation parameters are shown in (Table 1).

Table 1 Simulation Parameters

Parameters	Value
Area	400 X 400 m2
Amount of Wireless Sensor nodes	200
Start-up energy of sensor	1.5J
Length of Data Packet	5KB
Size of Hello Packet	120B
Percentage of Aggregation	12%
Default altitude of flying UAV	60m
Altitude of sensor	0 2m
Default speed of UAV	25m/s
Mobility of sensor	Static
Carrier frequency	2.45GHz
Type of Antenna	Omnidirectional
Mode of Propagation	Nakagami path loss model
Path loss exponent Sensor – UAV (ζ)	2.5
Sensor – Sensor path loss exponent	2.8

The EEODA data collection technique was compared with three other techniques of data collection, namely LEACH Heinzelman et al., (2002) (Low Energy Adaptive Clustering Hierarchy) communication with UAV, HEED (Hybrid Energy Efficient Distributed clustering) communication with UAV, and EFDC. Two of the compared protocols change their cluster heads every round, while EFDC used static cluster heads. However, this paper has shown that fixed election of cluster heads will lead to a quick energy depletion of the node, since the assumption here is that all wireless sensor nodes have equal energy. Similarly, a constant re-election of cluster heads will lead to instability of the clustering protocol.

In this paper, the Q-learning cluster election was used where a change in the cluster head assignment is effected only when the energy is lower than that of any other in the cluster. The Q-learning cluster head election technique also has a limited communication overhead compared to that of the compared protocols. City section mobility model was used for the three compared clustering protocols. It should be realized here that the assumption in the majority of the data collection procedures used for UAV wireless sensor networks is that the UAV knows the location of the cluster head ab initio due to the presence of infrastructures in the network. Although EFDC algorithm does not use this assumption, its static election of cluster heads makes the cluster heads prone to quick energy depletion, as shown in the simulation experiments.

The Energy Consumption Model

The energy consumption model used in this paper is the same as that used in (Nayak and Devulapalli, 2016). According to the researchers in Nayak and Devulapalli, (2016), the primary cause of energy depletion in wireless sensor nodes is due to data transmission and reception. As stated in the paper, the energy required to transmit one bit of data over a distance Γ to the receiver is as shown in Equation 29.

$$\mathrm{ETx}\,(\lambda, \Gamma) = \mathrm{ETx} - \mathrm{elec}\,(\lambda) + \mathrm{ETx} - \mathrm{amp}(\lambda, \Gamma) = \begin{Bmatrix} \lambda * E_{elec} + \lambda * \Delta_{hf} * \Gamma^2, & \Gamma < \Gamma_{th} \\ \lambda * E_{elec} + \lambda * \Delta_{jd} * \Gamma^4, & \Gamma < \Gamma_{th} \end{Bmatrix} \tag{29}$$

Where E_{elec} denotes the initial energy of the wireless sensor nodes. ETx – elec (λ) denotes the transmission energy required of the wireless sensor nodes to transmit Γ bit of data. Similarly, ETx = amp (λ, Γ) denotes the energy required by the amplifier circuit to transmit λ bit of data over a length of Γ. Also Δ_{hf} *and* Δ_{jd} are constants which are dependent on environmental conditions. While Δ_{hf} represents the model for the transmitter amplifier in free space, Δ_{jd} represent the same model for a multipath propagation model. It should be realized that both Δ_{hf} and Δ_{jd} are dependent on the length of the gap between the transmitter and receiver. The value Γ_{th} which represents the threshold distance can be computed as shown in Equation 30.

$$\Gamma th = \sqrt{\frac{\Delta_{hf}}{\Delta_{jd}}} \qquad (30)$$

Now. in the case where absolute displacement of the transmitter to the receiver is higher than $\Gamma_{th,}$ the multipath propagation model is used, otherwise the free space propagation model is used. The energy required for the reception of data packet of length Γ can be computed from Equation 31.

$$ERx(\Gamma) = ERx - elec * \Gamma \qquad (31)$$

It can be seen from Equation 31 that the consumed energy from the reception of Γ bit of data is represented as ERx (Γ). Similarly, ERx - elec represents the energy consumed in the reception of 1 bit of data.

For the measurement of the energy consumption in the EEODA, the data transmission and reception of the wireless sensor nodes with the UAV is considered in five stages, namely initialization, identification of neighbour nodes, routing, data aggregation and forwarding aggregated data to the sinks (UAVs). For the computation of consumed energy of the nodes Equation 29 and 31 was used. For the analysis of consumed energy of the nodes, the simulation time was dependent on the completion of the five phases of the EEODA communication system explained in section 4.2. The simulation time was thus different for the four compared protocols. The energy consumption considered in this paper was only for the wireless sensor node, this is because the UAVs are assumed to use energy harvesting, thus its energy was renewable. The consumed energy of the nodes is given in Equation 32.

$$E\sigma = \sum_{S_{k=1}}^{S_k} \left(\sum E_{TXk} + \sum E_{RXk}\right) \qquad (32)$$

Where Eσ represents sum of the energy consumed by the wireless sensor nodes as a result of both transmission (E_{TXk}) and reception ($\sum E_{RXk}$) of data packets. Sk denotes the sensors in each cluster. The analysis for the consumed energy is done after the completion of the five phases of EEODA communication system.

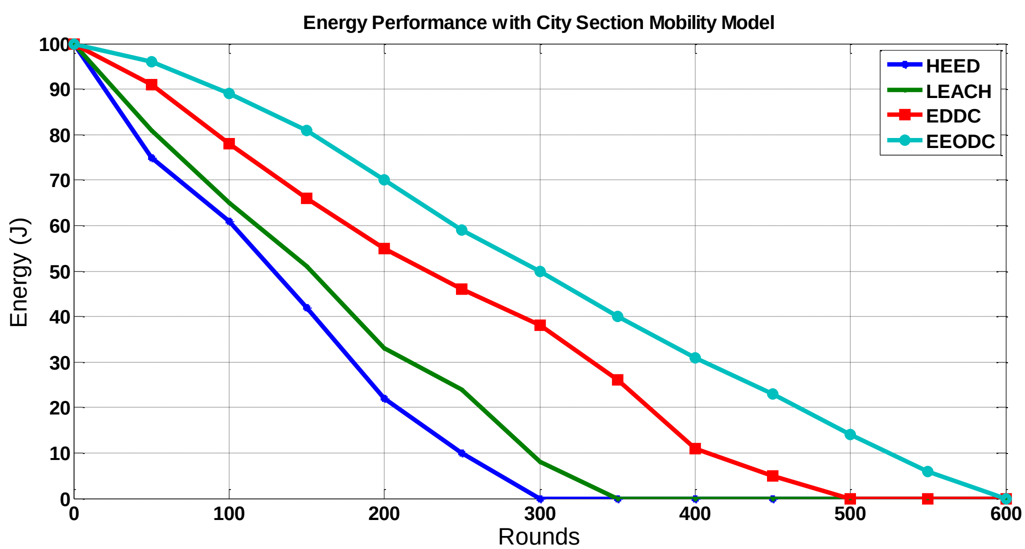

Figure 8 Energy Performance with City Section Mobility Model

Figures 8 shows the analysis for energy consumption for the wireless sensor nodes in EEODA protocol, direct transmission of data packets were only between the UAVs and the cluster heads. This is an advantage over the EFDC algorithm where data packet transmission occurred between the UAVs and all the sensor nodes in the cluster. Similarly in the LEACH and HEED protocol, the constant re-election of cluster heads in each round of clustering causes a huge drainage of the sensor energy, it also incurs huge communication overhead leading to low network lifetime. Also, the EEODA protocol optimizes the distance between the UAVs and the cluster heads for optimum data transmission instead of between the UAVs and all nodes in the cluster, this also reduces data transmission rate in the EEODA as compared to EFDC. The reduction in data transmission rate in EEODA leads to reduced transmission power for the nodes due to a reduction in received packets, this situation leads to an overall improvement in energy consumption and increased network lifetime.

The Model for Iteration Delay

The mathematical model for the delay in iteration of EEODA is given in Equation 33.

$$D\psi = \sum_{g_{v=1}}^{Sc}\left(\frac{j(K)}{EF_{UAV}} + \sum_{j=1}^{n} CH_{time}\right) \quad (33)$$

Where Dψ represents the sum total of the time necessary for the UAVs to get the aggregated data from the cluster heads for every round of data transfer to the processing centre. EFUAV denotes the UAV speed used in the simulation. The different positions for the UAV before obtaining the optimum position for each cluster set is described in section VII. J(K) denotes the sum of the distance covered by the UAV using the city section mobility model. CHtime denotes the time elapsed in getting the aggregated sensed data from the cluster head to the processing center. $\sum_{j=1}^{n} CH_{time}$ represents the total time required by all the cluster heads in the network to get their aggregated data to the processing center or sink. It should be noted the default speed of UAV used in the simulation was kept constant, which implies that speed change arising from wind assisted U movement was not included in the model.

Result and Analysis of Simulation Experiments

This section discusses the results of the simulation experiments conducted using the novel EEODA algorithm designed in this paper. The performance of the EEODA algorithm is compared with three other protocols namely HEED, LEACH and EFDC algorithms. Each of this algorithm was deployed in UAV – WSN environment.

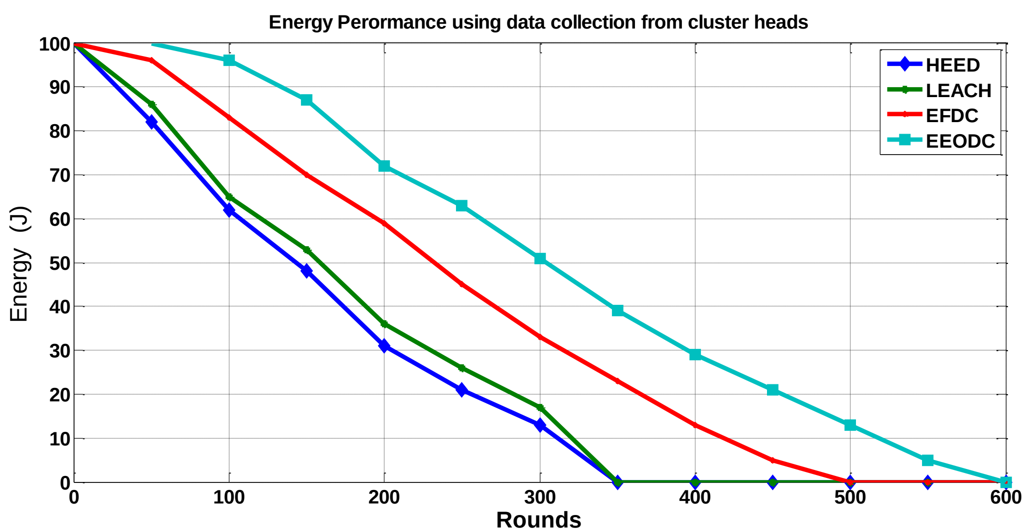

Figure 9 Energy Performance using data collection at cluster heads

In Figure 9 the EEODA's energy consumption was compared to the other three protocols named earlier. All protocols used the city section mobility model for the UAV's movement towards an optimum position for data collection from the cluster heads. It should be noted here that contrary to the data collection from all nodes in the network as deployed in the other three compared protocols, the data collection in this paper was limited to only the cluster heads. This was appropriate because the cluster heads aggregate the data from all sensors in its cluster, and as all sensors are homogenous i.e., they sense the same environmental parameter, gathering data from all the sensors n the clusters would lead to data redundancy. Similarly, the Q-learning algorithm deployed in the algorithm design enable the sensors to seamlessly agree on a cluster head as opposed to the mathematical computation employed in the three compared protocols.

From Figure 9, it can be seen that the energy consumption was least in EEODA followed by EFDC, the reduced energy consumption in EEODA is due to the UAV's communicating with only the cluster heads as opposed to all sensors used in other protocols. The EEODA outperforms the EFDC protocols in energy consumption by 12%, while its performance exceeds LEACH and HEED protocols by 26% and 34% respectively. The further performance improvement over HEED and LEACH algorithm is due to the act that distance optimization technique was not employed in them. As is generally known in wireless sensor network, energy depletion of the nodes is mainly due to data transmission from source to destination, data transmission involving all the nodes in the network at all times will lead to higher aggregated energy depletion.

However, through a wise election of cluster heads using Q-learning, the network lifetime will be increased because all nodes will not die at approximately the same time as is the case with the other three compared protocols. Also, the forwarding of Hello packets by the UAVs will incur a high level of energy in the three compared protocols as all the nodes will respond to the packet in contrast to only the cluster heads as used in this paper. It is argued here that single data transmission between the cluster heads and the UAV at every round of clustering will lead to a greater energy conservation as compared to when all nodes in the network are communicating consecutively with the UAVs.

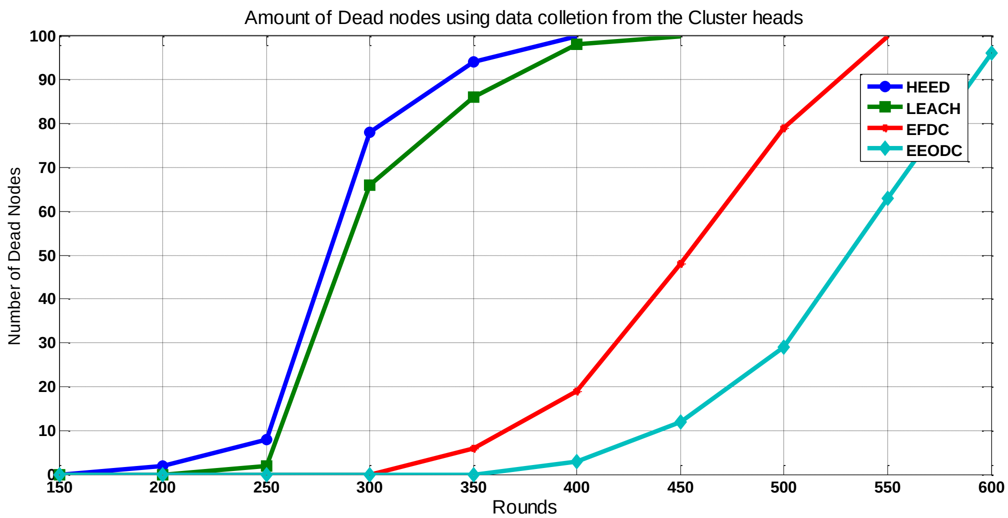

Figure 10 Number of dead nodes using data collection from cluster heads

In Figure 10, the EEODA protocol was compared with the three other protocols using number of dead nodes in each round as the performance metric, however in this simulation, the UAVs were designed to communicate directly with only the cluster heads to receive aggregated sensed data from the nodes in their respective clusters. It can be noticed from the results here that the performance of EFDC was slightly improved and the number of dead nodes in each round for EFDC was marginally increased to 520 rounds. The performance for EEODA protocol still exceeds that of EFDC by 24%. The performance increase is due to the role-free nature of Q-learning clustering algorithm compared to that in EFDC. The Q-learning based EEODA clustering protocol does not involve intensive mathematical computation necessary for the election of cluster heads as is the case in EFDC.

The clustering algorithm for EFDC is cumbersome and requires much computational processing. The performance of both HEED and LEACH were further reduced due to the protocols not employing any optimization techniques in their UAV location. It will be noticed here that the LEACH protocol performance is much improved over the HEED protocol. This is due to the extra communication overhead incurred by HEED due to its repeated re-election of cluster heads as opposed to that of LEACH. EEODA employs a much-reduced re-election procedure due to its simple computation of Q-value. The EEODA protocol uses energy optimization scheme in determining the sequence of positions for the UAVs, this further reduces its energy consumption as compared to the other three protocols. EEODA outperforms the LEACH and HEED protocols by 34% and 46% respectively.

In Figure 11, the EEODA protocol was compared to the other three protocols in terms of the number of dead nodes in every round of clustering using the city section mobility model. Here dead node refers to those sensors whose energy has been depleted as a result of their batteries running down. Their remaining battery energy has then run down below the threshold value required for communication with the UAVs. From the results in Figure 11, it will be noticed that the number of dead nodes was highest in HEED, while EEODA has the best result. In HEED and LEACH all the nodes die in approximately 360 rounds EFDC dies in approximately 540 rounds while it takes 600 rounds for EEODA nodes to die. This is due to the optimization of the UAVs distance to the cluster heads. The EFDC protocol also employ energy optimization, however, in this simulation the UAVs were made to communicate with all the nodes in the network, this accounts for the large difference noticed in the network lifetimes between the EEODA and the EFDC The further reduction in the performance in both HEED and LEACH is due to their not employing optimization techniques for the UAV’s location

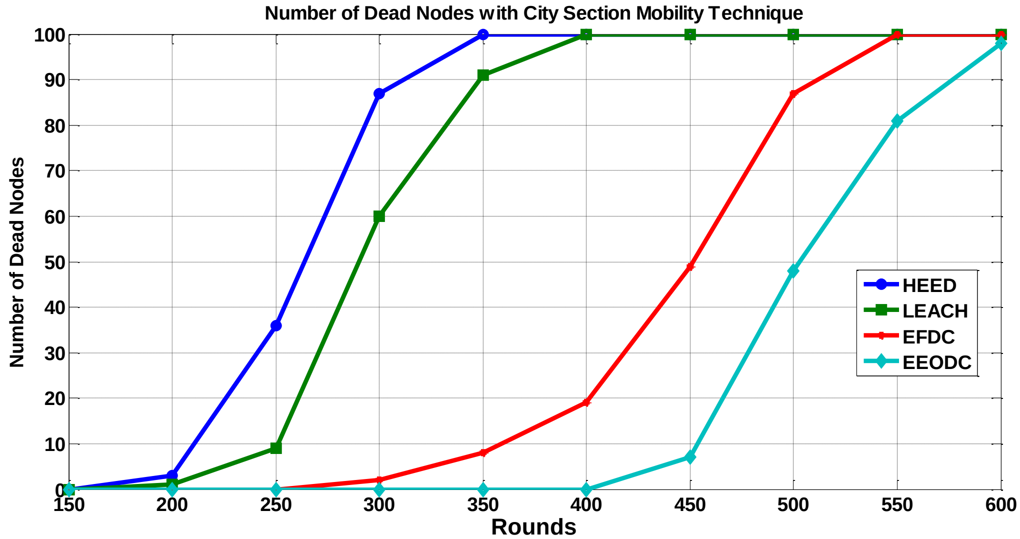

Figure 11 Number of dead nodes with city section Mobility Technique

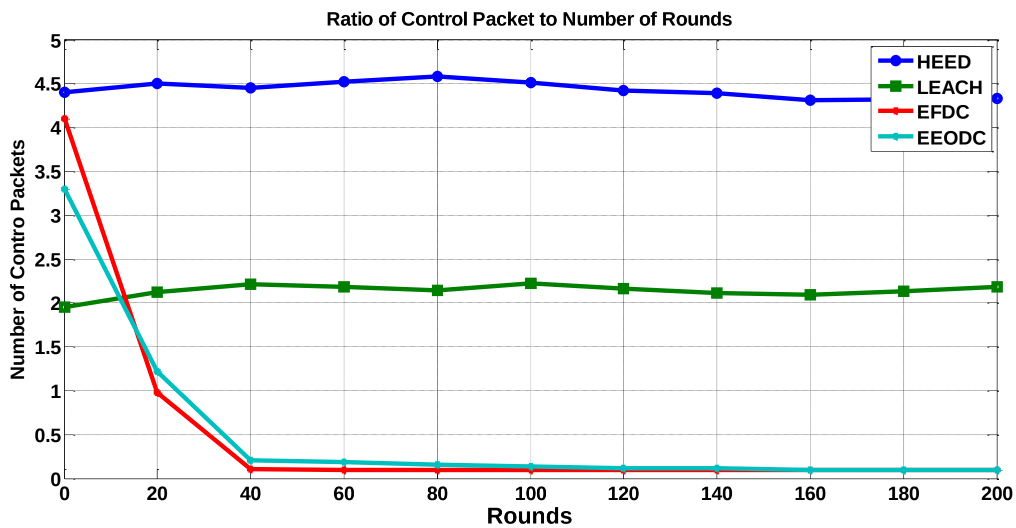

Figure 12 Ratio of Control packets to Number of rounds

In Figure 12, the EEODA protocol was compared with the other three protocols in terms of control packet overhead incurred for every round of clustering. It can be seen from the figure that in EEODA, the number of control packets were smaller overall to the other protocols. The EFDC protocol performance was initially best because it performs the clustering process once, however as time goes on its control overhead increases more than EEODA, this is because as time progresses, the nodes energy reduces below the threshold value, this makes the sensors reply the UAVs with unacknowledged packets.

Hence a high control overhead is required as the protocol searches relentlessly for nodes with enough energy to send sensed data to the UAVs. This causes the nodes to exchange a high number of control packets amongst themselves. The EEODA unlike the HEED and EACH protocols involve control re-election procedures as cluster heads do not change at every round of clustering but only when the Q-values of the cluster head is below a set point. For this reason, it is able to evenly balance the energy of the nodes in each cluster. The EEDOD outperforms the EFDC protocol in terms of control packet by an average of 12%, and 5% and 37% to the HEED and LEACH protocols respectively.

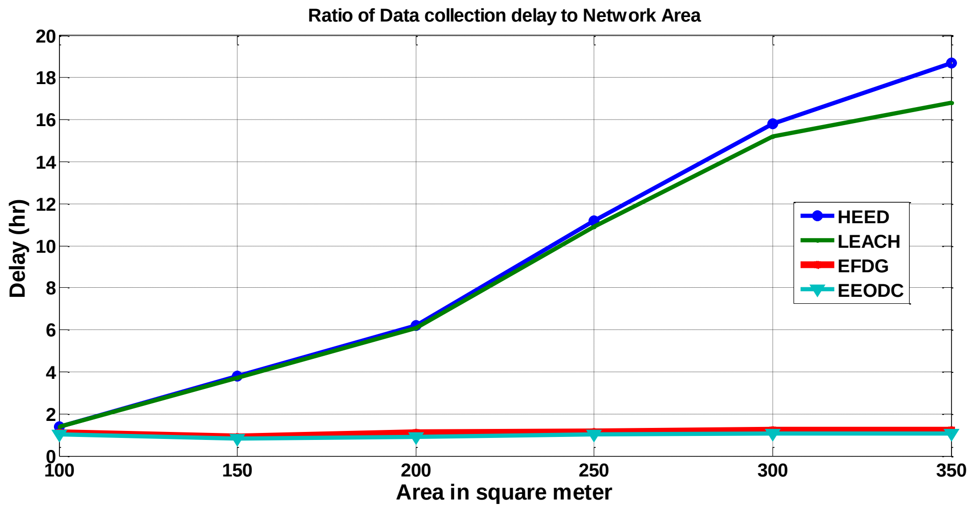

Figure 13 Ratio of Data collection delay to network area

In Figure 13, the ability of the protocols to respond to increasing network area was shown. The scalability performance was done among the four protocols namely EEODA, EFDC, LEACH and HEED. As stated during the clustering analysis, a square cluster area or ROI was assumed. It should be noticed here that the distance between the clusters were increased as a result of network area increase, this invariably increased the energy required for intra-cluster data transmission. It can be seen from the figure that EEODA outperforms EFDC protocol by 9%, while its performance is improved compared to LEACH and HEED by 25% and 38% respectively. The performance increase in EEODA is mainly attributed to the use of city section mobility model in the computation of the UAV's position as well as its role-free clustering technique which always elect wireless sensor nodes with highest battery energy in the cluster to send aggregated data to the UAV. Secondly the cluster heads send only unique data unlike EFDC where a high number of redundant bits are sent to the UAVs. Thirdly the simulated annealing search method proves to be superior to the tabu search algorithm used in EFDC.

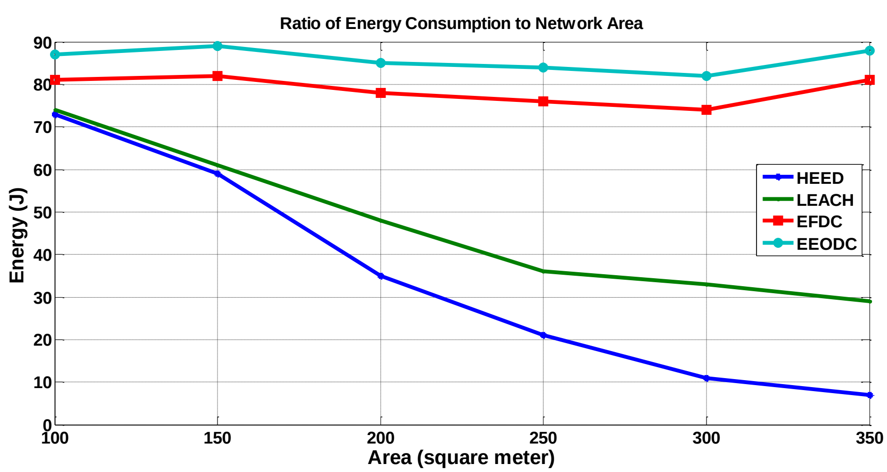

Figure 14 Ratio of energy consumption to Network area

In Figure 14, the energy consumed for data collection only at the CH is analysed in the four compared protocols. It can be noticed in the figure that EEODA outperforms EFDC by 13% and HEED and LEACH by 36% and 28% respectively. The performance increase in the EEODA is due to the fact that the algorithm is originally designed for the UAVs to receive aggregated data directly from the CHs unlike in the other three compared protocols.

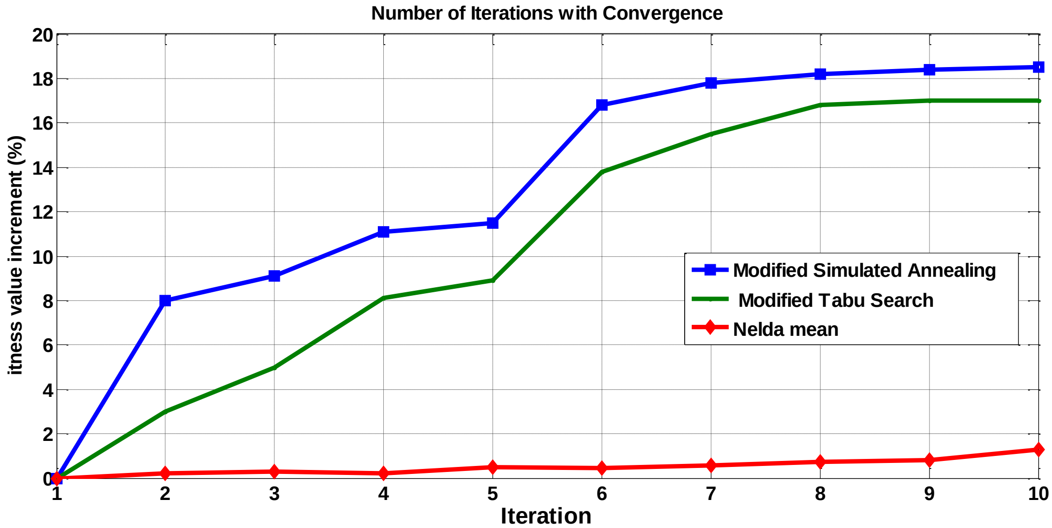

Figure 15 Number of iterations with convergence time

In Figure 15, the convergence time for the search algorithm proposed in this paper, (modified simulated annealing) was compared to two other search algorithms namely modified tabu search and Nelder mead optimization. Data were taken ten times in a single iteration for the entire duration of the experiment. The percentage change in the fitness value was then measured in comparison to their fitness value (each protocol) at the. commencement of the simulation. From Figure 15 it can be seen that EEODA outperforms EFDC protocol by 14% and HEED and LEACH protocol by 23% and 35% respectively. It can be seen in the figure that the fitness value for EFDC does not change after the sixth iteration. This is not a good observation as it cannot react quickly to change in network configurations. Even though a static sensor configuration was assumed an increase in the number of nodes in the network will lead to configuration change which the EFDC protocol will not be able to respond appropriately to.

4. CONCLUSION

This paper proposed a clustering algorithm in WSN-UAV network. The algorithm EEODA is ideal for data collection in hilly and uneven surfaces in which network infrastructures will not be easily deployed. To design an energy efficient data collection algorithm in this scenario, it is required to obtain an optimum 3-D location for the UAV, it is also necessary for such algorithm to converge to this optimum position in a reasonably short time. Q-learning paradigm was used to design an algorithm to obtain appropriate positions for the cluster heads. The CHs were also made to report their identity to the UAVs (by flagging the appropriate bit in the control packet sent by the UAVs) so that the protocol doesn't need to search for the location of the aggregated sensed data.

Also, the protocol reduced energy consumption in the network by eliminating the transfer of redundant sensed data to the UAVs. Modified simulated annealing algorithm was used to compute the sub-optimal positions for the UAVs, control feature in the algorithm was used to limit the number of iterations needed by the UAVs to get to an optimum position for data collection. This control was needed to reduce energy consumptions by the sensors before the protocol converges. Simulation experiments conducted in MATLA*B show that the EEODA algorithm outperforms the compared protocols in energy consumptions number of dada nodes, time of convergence, data collection delay and number of controls. Packets. The future work / modifications to be done in the experiment is assuming mobility of the wireless sensor nodes.

Declarations

We the authors declare that the information contained in this paper is a result of our research findings and is a unique property.

Authors Contribution

The authors worked together for the successful completion of this work.

Acknowledgement

The authors acknowledge Department of Electrical and Computer Engineering, Abia State University for making available the use of their laboratories.

Informed consent

Not applicable.

Ethical approval

Not applicable.

Conflicts of interests

The authors are staff of Abia State University, Uturu, Abia State, Nigeria. They work together to unearth new frontiers in electronic and communication engineering. They collaborate to foster their research interest and individual accomplishments.

Funding

The study has not received any external funding.

Data and materials availability

All data associated with this study are present in the paper.

REFERENCES AND NOTES

1. Ali ZA, Masroor S, Aamir M. UAV based data gathering in wireless sensor networks. J Wirel Pers Commun 2021; 106(4): 1801–1811.
2. Arafat MY, Habib MA, Moh S. Routing protocols for UAV-aided wireless sensor networks. J Appl Sci 2022; 10(12):4077.
3. Chen J, Yan F, Mao S, Shen F, Xia W, Wu Y, Shen L. Efficient data collection in large-scale UAV-aided wireless sensor networks. in Proc. 11th Int Conf Wireless Commun Signal Process. (WCSP) 2019; 1-5.
4. Dandekar DR, Deshmukh PR. Energy balancing multiple sink optimal deployment in multi-hop wireless sensor networks. Proceedings of the 3rd International Advance Computing Conference (IACC ’13), IEEE; Ghaziabad, India 2013; 408–412.
5. Ebrahimi D, Sharafeddine S, Ho P-H, Assi C. UAV-aided projection-based compressive data gathering in wireless sensor networks. IEEE Internet Things J 2018; 6:1893–1905.
6. Eckerstorfer M, Bühler Y, Frauenfelder R, Malnes E. Remote sensing of snow avalanches: Recent advances, potential, and limitations. Cold Reg Sci Technol 2016; 121:126–140.
7. Hammoudeh M, Al-Fayez F, Lloyd H, Newman R, Adebisi B, Bounceur A, Abuarqoub A. A Wireless Sensor Network Border Monitoring System: Deployment Issues and Routing Protocols. IEEE Sens J 2017; 17:2572–2582.
8. Hayes T, Ali FH. Location aware sensor routing protocol for mobile wireless sensor networks. IET Wireless Sens Syst 2016; 6(2):49–57.
9. Heinzelman WB, Chandrakasan AP, Balakrishnan H. An application-specific protocol architecture for wireless micro-sensor network. IEEE Trans Wirel Commun 2002; 1(4):660–67 0.
10. Ho D-T, Grøtli EI, Sujit PB, Johansen TA, Sousa JB. Optimization of wireless sensor network and UAV data acquisition. J Intell Robot Syst 2015; 78(1):159-179.
11. Hu J, Wang T, Yang J, Lan Y, Lv S, Zhang Y. WSN-assisted UAV trajectory adjustment for pesticide drift control. Sensors (Basel) 2020; 20(19):5473. doi: 10.3390/s20195473
12. Joukhadar A, AlChehabi M, Stöger C, Müller A. Trajectory tracking control of a quadcopter UAV using nonlinear control. In Mechanisms and Machine Science. Springer 2019; 58:267–282.
13. Khan MI, Gansterer WN, Haring G. Static vs. Mobile sink: The influence of basic parameters on energy efficiency in wireless sensor networks. Comput Commun 2013; 36(9):965–978.
14. Liu B, Zhu H. Energy-effective data gathering for UAV-aided wireless sensor networks. Sensors (Basel) 2019; 19(11):2506. doi: 10.3390/s19112506
15. Liu Y, Ota K, Zhang K, Ma M, Xiong N, Liu A, Long J. QTSAC: An energy-efficient MAC protocol for delay minimization in wireless sensor networks. IEEE Access 2018; 6 :8273–8291.
16. McArthur DR, Chowdhury AB, Cappelleri DJ. Autonomous control of the interacting-Boom Copter UAV for remote sensor mounting. IEEE Int Conf Robot Autom (ICRA) 2018; 5 219–5224.
17. Miranda J, Abrishambaf R, Gomes T, Goncalves P, Cabral J, Tavares A, Monteiro J. Path loss exponent analysis in wireless

sensor networks: Experimental evaluation. IEEE Int Conf Ind Inform (INDIN) 2013; 54–58.
18. Mi J, Wen X, Sun C, Lu Z, Jing W. Energy-efficient and low package loss clustering in UAV-assisted WSN using K means and fuzzy logic. In Proc. IEEE/CIC Int Conf in. Commun. Workshops China (ICCC Workshops), 2019; 210–215.
19. Nagata SI, Miyagawa J, Murakami K. Flexible flight navigation for quadcopters based on geometric formation using motion sensing. In Proc. Int Workshop Advanced Image Technology (IWAIT) 2020; 1(5):1–4.
20. Nayak P, Devulapalli A. A fuzzy logic-based clustering algorithm for WSN to extend the network lifetime. IEEE Sensors J 2016; 16(1):137–144.
21. Nayak P, Vathasavai B. Energy efficient clustering algorithm for multi-hop wireless sensor network using Type-2 fuzzy logic. IEEE Sensors J 2017; 17(14):4492–4499.
22. Ogundile O, Alfa A. A survey on an energy-efficient and energy- balanced routing protocol for wireless sensor networks. Sensors (Basel) 2017; 17(5):1084. doi: 10.3390/s17051084
23. Pang Y, Zhang Y, Gu Y, Pan M, Han Z, Li P. Efficient data collection for wireless rechargeable sensor clusters in harsh terrains using UAVs. IEEE Glob Commun Conf 2014; 234–239.
24. Popescu D, Stoican F, Ichim L, Stamatescu G, Dragana C. Collaborative UAV-WSN system for data acquisition and processing in agriculture. IEEE International Conference on Intelligent Data Acquisition and Advanced Computing System Technology Application (IDAACS) 2019; 519–524.
25. Poudel S, Moh S. Energy-efficient and fast MAC protocol in UAV-aided wireless sensor networks for time-critical applications. Sensors (Basel) 2020; 20(9):2635. doi: 10.3390/s20092635
26. Poudel S, Moh S. Medium access control protocols for unmanned aerial vehicle-aided wireless sensor networks: A survey. IEEE Access 2019; 7:65728–65744.
27. Rezoan AN, Sangman M. Energy-Efficient and Fast Data Collection in UAV-Aided Wireless Sensor Networks for Hilly Terrains IEEE Access 2021; 1:23168–23190.
28. Say S, Inata H, Liu J, Shimamoto S. Priority-based data gathering framework in UAV-assisted wireless sensor networks. IEEE Sensors J 2016; 16(14):5785–5794.
29. Sumathi MS, Anitha GS, Sridhar NH. Efficient data handling of wireless sensor network for real time landslide monitoring system using fuzzy technique. In Proc. International. Conference of Circuit, Power and Computing Technology. (ICCPCT), 2019; 235–251.
30. Sun Y, Xu D, Ng DWK, Dai L, Schober R. Optimal 3D-trajectory design and resource allocation for solar-powered UAV communication systems. IEEE Trans Commun 2017; 65 (3):1077-1091.
31. Tang W, Ma X, Wei J, Wang Z. Measurement and analysis of near- ground propagation models under different terrains for wireless sensor networks. Sensors 2019; 19(8):1901-1924.
32. You C and Zhang R. 3D trajectory optimization in Rician fading for UAV-enabled data harvesting. IEEE Trans Wirel Commun 2019; 18(6):3192–3207.
33. Zhan Y, Zeng U, Zhang R. Energy-efficient data collection in UAV enabled wireless sensor network. IEEE Wirel Commun J 2020; 7(3):328–331.