Check for updates

Energy-efficient hierarchical routing in wireless sensor networks based on fog computing

Ademola Philip Abidoye*[iD] and Boniface Kabaso

*Correspondence:
abidoyea@cput.ac.za
Department of Information
Technology, Cape Peninsula
University of Technology,
Cape Town 7925, South
Africa

Abstract

Wireless sensor networks (WSNs) have been recognized as one of the most essential technologies of the twenty-first century. The applications of WSNs are rapidly increasing in almost every sector because they can be deployed in areas where cable and power supply are difficult to use. In the literature, different methods have been proposed to minimize the energy consumption of sensor nodes to prolong WSNs utilization. In this article, we propose an efficient routing protocol for data transmission in WSNs; it is called energy-efficient hierarchical routing protocol for wireless sensor networks based on fog computing. Fog computing is integrated into the proposed scheme due to its capability to optimize the limited power source of WSNs and its ability to scale up to the requirements of the Internet of things applications. In addition, we propose an improved ant colony optimization algorithm that can be used to construct an optimal path for efficient data transmission for sensor nodes. The performance of the proposed scheme is evaluated in comparison with P-SEP, EDCF, and RABACO schemes. The results of the simulations show that the proposed approach can minimize sensor nodes' energy consumption, data packet losses, and extends the network lifetime. We are aware that in WSNs, the certainty of the sensed data collected by a sensor node can vary due to many reasons such as environmental factors, drained energy, and hardware failures.

Keywords: Advanced nodes, Ant colony optimization, Cloud computing, Fog nodes, Fog computing, Heterogeneous network, Internet of things, Sensor nodes, Quality of service

1 Introduction

Next-generation wireless sensor networks (WSNs) are predicted to be deployed in the Internet of things (IoT). IoT has revolutionized the current world of technology. Electronic devices such as home appliances, medical equipment, various types of sensor nodes, cameras can connect and communicate with one another through the Internet. Thus, these devices have some limitations, and these include limited power source, storage, processing power, and bandwidth. Deployment of various types of sensor nodes in the areas of interest makes the networks heterogeneous. Heterogeneous sensor networks consist of sensor components with different abilities, such as hardware specifications,

identification, sensing type, communication capabilities, mode of operation, sensing rate, frequency [1]. Heterogeneity of the network provides greater flexibility in deployment with low overhead. The sensor nodes in the proposed scheme consist of low-end nodes and high-end nodes. The low-end nodes are normal sensor nodes that are constrained with limited computation and sensing abilities. They are responsible for gathering sensed data and transmission.

The high-end nodes consist of sensors with high processing power, long communication range, and high throughput. A network with heterogeneous sensor nodes can achieve load balancing and prolong the network lifetime utilization. WSNs are ubiquitously used everywhere in modern society, including in hydraulic power plants for environmental monitoring, in the production of vehicles in the automobile industry, for heat energy in greenhouses, and forest monitoring [2, 3]. In conventional WSNs, sensor nodes transmit their raw data to a base station for analysis and storage.

Recent improvements in IoT technologies have enabled WSNs to transmit raw data to the cloud for processing and storage. This process requires extremely high network bandwidth and drained the limited energy of sensor nodes [4]. However, instead of transmitting the huge amount of sensed data generated from the sensor nodes over the network and then processing them using a cloud computing platform, some processing can be performed close to the sensor networks using a technique called fog computing. Fog computing is composed of networking devices such as routers, proxy servers, set-top boxes, gateways [5]. These devices have higher processing capability and storage than sensor nodes. They can be placed between sensor nodes and cloud computing to receive sensed data from nodes. This technique will not only conserve sensor nodes' energy dissipation during transmission but also provide low latency, location awareness, and high bandwidth for WSNs. The term 'fog computing' was developed by Cisco to overcome limitations in cloud computing [6, 7]. Fog computing provides services closer to end-users. It consists of fog nodes (FNs) which provide resources at the edge of the network. They have a high processing power, power source, more storage which can be used as fog servers in the network. This new technology is implemented at the edge of the network. It offers several benefits to end-users; these include efficient network bandwidth usage, data security, fewer bottlenecks, the solving of high latency on the network, increased reliability of transmitted sensed data, and a higher speed of analysis [8]. Fog computing is integrated into our architectural model to address some of the constraints of WSNs and also to exploit all the benefits of fog computing mentioned above.

FNs have the capability to receive raw data from WSNs, aggregate, perform pre-processing, and store them temporarily instead of sensor nodes forwarding all the raw data to the cloud computing, thus reducing network bandwidth and latency [9].

The integration of fog computing with the WSNs and IoT creates a new type of service called fog as a service (FaaS) [10]. However, for fog computing to fully deploy its service, each fog node should meet the following requirements: (1) concurrent data collection from low-end nodes, (2) high processing power and performance to support real-time data processing and analysis, (3) high service reliability, and (4) low power consumption, to achieve long-time utilization. Fog nodes in fog computing architecture function as middleware, operating between end-users and cloud computing. In the context of WSNs, they provide resources to the underlying sensor

nodes. The integration of fog computing into WSNs can overcome numerous problems [11, 12].

Routing by ant colony optimization (ACO) is used to find the optimal path from source nodes to destination nodes in WSNs, using the following metrics: transmission distance, node's energy level, quality of service (QoS), pheromone, and heuristic factor [13, 14]. Ants in a routing algorithm are agents traveling from one node to another in the network. They deposit a chemical substance called pheromone on their path as they traverse the network. This method has been used to solve many optimization problems in WSNs [15–17].

This work is different from the work presented in [18] as follows.

We develop an algorithm that can uniformly distribute the advanced nodes in the network for uniform energy distribution among the nodes.

We ensure that only advanced nodes can be selected as cluster heads unlike in [18], and the authors stated that either normal nodes or advanced nodes can be used as cluster heads. This can create an energy hole in the networks and result in loss of data packets.

We develop an improved ACO algorithm to construct an efficient and optimal path among the nodes.

Continuous technological advancements and innovations in the twenty-first century have increased the development and applications of WSNs in many areas of human endeavor. However, sensor nodes that form WSNs have a limited power source. It is essential to save the lives of individual sensor nodes due to sensitive sensory data they transmit in most cases in the application areas. In order to overcome this challenge, fog computing is integrated into our scheme. Fog computing is a good technology that can provide a solution to some of the shortcomings of WSNs with its capability to meet the requirements of the IoT applications. Moreover, WSNs generate a large amount of sensed data which becomes significantly challenging to handle for the nodes. Thus, aggregating, filtering, and transmitting each sensed data would require devices with higher processing and memory capacity; fog computing can provide these functionalities.

Finally, this technology provides low latency, storage scalable, ubiquitous, layered, and federated network connectivity between WSNs and cloud computing.

The proposed method consists of a three-tier hierarchical network structure as shown in Fig. 1. We discuss each of these tiers in Sect. 3.

The contributions of this paper are as follows:

1. We propose an energy-efficient hierarchical routing in wireless sensor networks based on fog computing (EEHFC).
2. The proposed scheme consists of many nodes with different capabilities. We drive mathematical expressions for the specifications of each device in the network for seamless integration with one another.
3. We develop mathematical models for normal sensor nodes, advanced nodes, and fog nodes representing their physical components based on their service delivery for efficient data transmission in the networks.

Fig. 1 Architecture of the proposed scheme
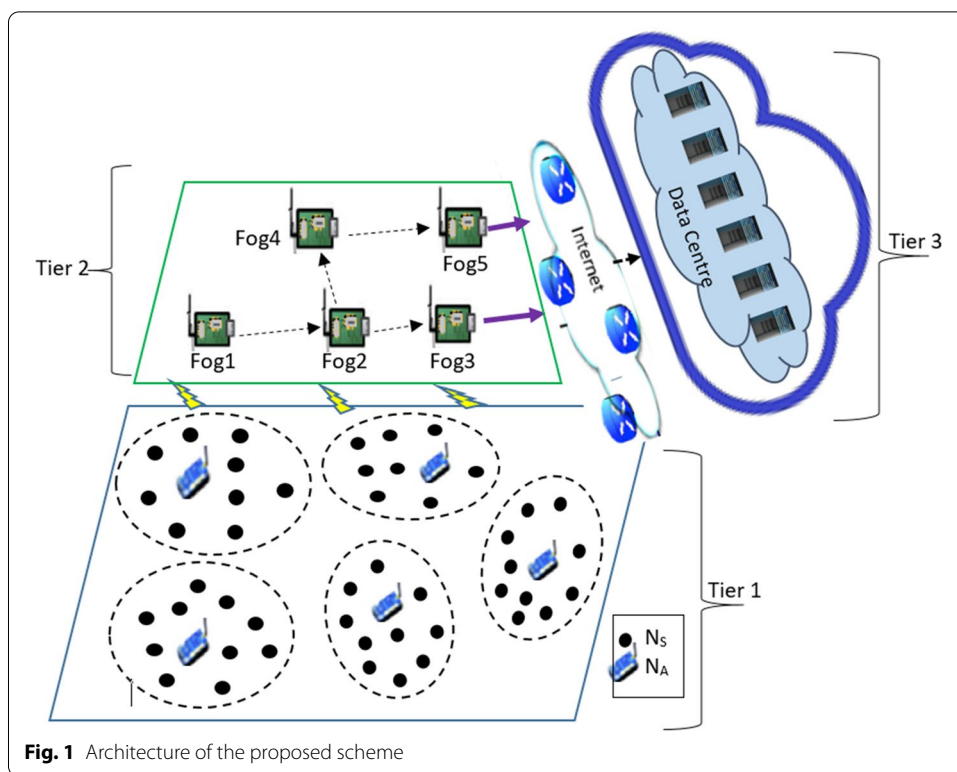
4. We develop mathematical models for the proposed scheme. An algorithm is designed for uniform distribution of advanced nodes to achieve load balancing within the network. In addition, an algorithm is designed for reliable end-to-end data packets transmitted between the fog nodes and cloud computing.

5. An improved ACO algorithm is developed to adjust routes to minimize the total number of broadcast transmission in the network.

6. We evaluated EEHFC using real data sets from the wireless environment. Results show that EEHFC performs better than the three related schemes.

The locations and number of sensors deployed in an interesting area determine the topology of the network, which will further influence many of its fundamental properties, such as its lifetime, connectivity, cost, and coverage. Thus, the performance of a sensor network depends to a large extent on its deployment and the ecological conditions. Indeed, the information provided by sensors may not always reliable, either due to environmental factors such as Gaussian white noise, hardware failures, security issues, or operational tolerance levels. Therefore, it is very important to take into account this uncertainty in the deployment process and the results that are obtained.

The remainder of this article is structured as follows: Sect. 2 discusses related work, while fog computing architecture is presented in Sect. 3. Section 4 focuses on performance evaluation and a discussion of the results before Sect. 5 concludes the article.

2 Related work

The unique characteristics and wide application of WSNs have motivated researchers to propose various approaches and algorithms to optimize the performance of sensor nodes in networks, concerning many metrics (e.g., latency, throughput, energy cost, scalability, average energy consumption, end-to-end delay). Most of the approaches that have been used in the literature are based on the existing protocol [19], and attempts have been made to improve on them. Performance evaluation, for some of these proposed protocols, is undertaken using simulations, while others are implemented through testbeds. Some of the outstanding approaches which have been proposed by different authors (as reported in the literature) to minimize energy consumption in WSNs are presented below. The strengths and weaknesses of each approach are highlighted. In the next sections, a review of those approaches is presented under three subsections: hierarchical energy-aware routing protocols, heterogeneous routing protocols, and ACO routing protocols.

In order to conserve the limited energy that is available to sensor nodes and extend the network the lifetime, they are usually organized into different clusters in layers to form a hierarchical clustering network [20]. In each cluster, at least a node is selected as a cluster head (CH), depending on the protocol used and the size of the network. The nodes are responsible for collecting sensed data from member nodes, aggregate, and forward them to a base station via multi-hop communication. LEACH was the first unique and outstanding protocol proposed for clustering sensor nodes in WSNs, based on the hierarchical routing technique [21]. Most other protocols proposed thereafter (for hierarchical routing) have used the LEACH protocol as a benchmark.

However, this protocol has several shortcomings: (1) the sensor nodes selected as CHs in LEACH are not evenly distributed within the network; thus, nodes that are far from the CH transmit through long distances and more energy is consumed during the transmission; and (2) the CH selection method is based on probability, which may lead to an increase in overhead when selecting new CHs and may result in an increase in energy consumption.

Wang et al. [22] proposed a novel energy-aware hierarchical cluster-based (NEAHC) routing protocol for WSNs. The goals of their work are to minimize the total energy consumption and to ensure the load balancing of energy consumption between sensor nodes. They developed an algorithm for the proposed scheme, which is divided into two phases: cluster setup phase and the steady-state phase.

They model the relay node as a nonlinear programming problem and use the property of convex function to determine the optimal solution. The proposed scheme was evaluated through simulations. They compared their approach with two other related protocols. The proposed scheme minimizes communication cost and direct data transfer by nodes close to base station prolongs the network lifetime compared with selected related protocols.

However, the network area and the number of nodes they used for their simulation are small. There is a high possibility for this approach not to perform better if the network size is large, for instance, if the number of nodes is 500 or more.

Xie et al. [23] proposed energy-efficient routing for mobile data collectors in WSNs with obstacles, by dividing the network region into grid cells with the same size. This approach provides a convenient construction of the spanning graph, using the line sweep approach.

The graph is composed of cells, which include the shortest search route for the mobile data collector (MDC). This method represents a heuristic tour-planning algorithm based on a complete graph. Performance evaluation of this approach shows that it is able to successfully dispatch MDC and extend the lifetime of the WSNs. However, only one MDC was used to evaluate this approach, when it should be evaluated using many collectors.

Ayoub et al. [24] proposed a cluster-based multi-hop advanced heterogeneity-aware energy efficient (MAHEE) protocol, which reduces the energy consumption of sensor nodes by choosing optimal CHs, thus allowing multi-hop inter-cluster communication using equal amounts of energy among all nodes. The algorithm is based on a clustering path planning algorithm for WSNs. In the proposed method, a node with higher energy is selected as a CH from among other nodes in the network, which achieves load balancing among the other nodes. MAHEE consists of two types of sensor nodes—normal and advance—which are equipped with different initial energy values. This helps to select a suitable CH based on the residual energy and distance from a base station. Simulation results show that the proposed approach performs better than existing state-of-the-art heterogeneous routing protocols and improves network stability and lifetime.

However, there is a high possibility that the CHs which are selected are not randomly distributed within the network. Thus, sensor nodes that are far from the CHs will transmit across a long distance and dissipate more energy.

Qureshi et al. [25] developed the Balanced Energy-Efficient Network-Integrated Super Heterogeneous (BEENISH) protocol. The protocol is an extension of (DEEC) in [26], in terms of choosing a CH based on the residual energy level of the nodes, with respect to the average energy of the network. BEENISH consists of four types of sensor nodes: normal, advance, super, and ultra-super nodes, each with a different energy level. The authors used the concept in [19] to divide the network into different clusters, in which ultra-super nodes have a higher probability of being selected as CHs than other types of nodes (because they are high-energy nodes). Simulation results show that this protocol performs better than previous clustering protocols in heterogeneous WSNs. However, the performance of the protocol is only based on two metrics—the number of alive nodes and data packets—which is not enough to determine how efficient and reliable the proposed protocol is, compared to other related heterogeneous protocols.

An efficient CH selection was proposed to extend the network lifetime, through an approach called Energy-Dependent Cluster Formation in heterogeneous WSNs (EDCF) [27]. The authors employ a clustering process to reduce energy consumption, to extend the network lifetime. The method of CH selection starts with the generation of a random number between zero and one. An individual sensor node that wants to become a CH generates that number and checks it against its threshold function. If the value of the number generated is less than the threshold value, the node is selected as the CH for that round. Otherwise, the same process is repeated by the next node, to become the CH. The results of the experiment show that the proposed protocol is better at prolonging the network life span than related heterogeneous protocols.

However, the proposed scheme is similar to the approach in [19], which means it will have the same weaknesses.

Fog-based energy-efficient routing protocol for WSNs (P-SEP) is proposed in [18]. The proposed scheme uses PEGASIS-based routing of fog nodes (FECR) and ACO-based

routing of fog nodes (FEAR) algorithms for the heterogeneous WSN. The selection of cluster heads (CHs) is based on probability function considering initial energy and current energy of sensor nodes in the network. The CHs transmit their data packets to the closest fog node which further processes and forwards its data to the cloud using FECR and FEAR algorithms. The results of the simulation show that the approach decreases energy consumption and increases network lifetime.

However, the authors claimed that either a normal node or an advanced node can be selected as a CH. If two or more normal nodes are consecutively selected as CHs, while advanced nodes remain as members of a cluster. An energy hole will be created and resulted in the loss of data packets being transmitted in the network.

Mohajerani and Gharavian [28] proposed a life-time-aware routing algorithm for WSNs (LTAWSN). A unique pheromone function was derived to integrate hops and energy consumption into the routing choice. The main aim of this approach is to optimize the energy consumption of the sensor nodes in WSNs. LTAWSN is better at minimizing energy consumption and extending the network lifetime than related routing algorithms for WSNs.

Sun et al. [13] proposed a unique routing algorithm called RABACO, based on the position information and search direction in ACO. This improves the network routing algorithm by considering the residual energy, heuristic function, sensor node communication distance, searching range, and transmission direction from the source node to the destination node. Simulation results show that the proposed algorithm minimizes the average energy consumption and prolongs the network lifetime.

3 Design of the study

The proposed scheme consists of three types of nodes, as shown in Fig. 1. The normal sensor nodes (N_S) and the advanced nodes (N_A) are distributed within the network, while the fog nodes (N_F) are placed at the edge of the network (middle layer). Advanced nodes are included in the sensor networks to eliminate overheads that are usually involved in the selection of CHs and unnecessary energy wastage during network operations. This differs from previous approaches reported in the literature, which are based on cluster formation and the selection of CHs [29–31]. In addition, the inclusion of fog computing ensures that not every sensed data are transmitted to the cloud for further processing and storage. Fog nodes can perform preprocessing on the received data instead of sending all the raw data to the cloud computing for processing and storage.

Given the short transmission range of sensor nodes, we divided the network into clusters of equal numbers of nodes, on average. Normal nodes that are geo-spatially close to an advanced node are grouped together to form a cluster. The relationship between the normal and advanced nodes is many-to-one, that is, many normal nodes belong to an advanced node. The number of clusters depends on the number of nodes in a network, and the application area, sensor nodes are application dependent. To ensure load balancing between the N_S and N_A, we developed an algorithm (Algorithm 1) for uniform distribution of N_A within the network, to avoid placing N_A on the same side of the network.

The N_A has an injective mapping (one-to-one) with the N_F, which is placed at the edge of the network. The clusters comprising of the N_S and N_A within that location are served by the edge nodes N_F. A normal node may leave or join any N_A based on its geographical distance and be connected to, and disconnected from, the corresponding N_A.

Cloud computing consists of data centers that host servers. Cloud computing is responsible for the extensive analysis of all sensed data received from N_F and storage. This is unlike the case with conventional cloud computing architecture, in which all sensed data are transmitted directly from sensor nodes to the cloud for the analysis and storage of a large volume of data. Thus, with the inclusion of fog computing into our scheme, moderate and relevant data are transmitted to the cloud for processing, analysis, and storage, leading to improve utilization and efficient use of cloud platforms.

3.1 Assumptions for the proposed scheme

The integration of fog nodes into WSNs is still new and yet to be implemented on a large scale. Therefore, we assumed the following for this network.

- The lowest layer of the network consists of normal nodes and advanced nodes which are different in terms of energy and capacity;
- Normal sensor nodes are only responsible for sensing environmental data and data transmission;
- Advanced nodes are responsible for receiving sensed data from member nodes;
- Fog nodes aggregate sensed data into a single packet;
- Fog nodes can process and store data in addition to routing data to cloud computing;
- Advanced nodes can transmit directly to the fog nodes; and
- Fog nodes' batteries can be recharged and easily replaced if they have run out.

3.2 Energy model for the proposed scheme

In this subsection, we developed an energy model for the proposed scheme; it consists of normal nodes and advanced nodes. In this model, the energy of advanced nodes is more than that of normal nodes in the network. A denotes the total number of sensor nodes in the network, the number of normal nodes is denoted by P, and the fraction of advanced nodes is denoted by K. If the initial energy of a normal node is denoted by E_f and the energy of advanced nodes is h more than the energy of normal nodes, then the advanced node energy will be $(1 + h) * E_f$.

The energy of normal nodes is $P * E_f$ and that of advanced nodes is $K(1 + h) * E_f$. Therefore, the total energy of the nodes in the network is expressed as follows

$$\begin{aligned} E_{\text{total}} &= P * E_f + K(1 + h) * E_f \\ &= AE_f + hE_f. \end{aligned} \tag{1}$$

This model shows that advanced nodes have hE_f energy more than normal networks with the energy AE_f.

We adopted the radio energy model in [19] for the communication distance between the sensor nodes.

3.3 Energy consumption of a sensor node

WSNs are usually deployed to collect environmental data in the area of interest. Every sensor node has a sensing rate and at regular time intervals transmit their sensed data to the nearest fog node. The source transmits the packets to the fog node by utilizing

services of intermediate relay nodes if the received node (fog node) is not within its transmission range; thus reduces the probability of packet loss and energy consumption [32]. We assume that the maximum transmission range for every node is denoted by distance R. If the distance $d_{i,j}$ between two nodes i and j is less than R, then energy dissipated by a sensor node $E_{i,j}^{\mathrm{T}}$ to transmit 1-bit of a data packet from i to j can be expressed as:

$$E_{i,j}^{\mathrm{T}} = E_{Tx} + \xi_{\mathrm{amp}} * r_{i,j}^{\beta} \tag{2}$$

where E_{Tx} is the energy per data packet consumed by the transmitter electronics, ξ_{amp} is the transmit amplifier dissipation per bit per square meter, and parameter $r_{i,j}^{\beta}$ connotes the energy loss due to channel transmission. It is assumed that the receiving energy E_{Rx} is constant.

If a sensing rate at node i is denoted by O_i, the energy consumed, E_{consumed}, of a sensor node to transmit a data packet at time t is expressed as:

$$E_{\mathrm{consumed}} = E_{i,j}^{\mathrm{T}} \sum_{i=1}^{p} (N_i) O_i * t \tag{3}$$

Based on this model, sensor nodes that are one hop from the fog node(s) will receive higher data traffic than nodes that are multi-hop from the fog nodes. They will consume more energy than sensor nodes that are far from the fog nodes.

Fog nodes are not power constrained to receive and pre-process sensed data received from the entire sensor network. Unlike conventional cloud architecture, in fog computing, not all data packets are forwarded to data centers in cloud computing for processing. Instead, all real-time analysis and latency-sensitive data packets are processed at the fog layer. The fog nodes in this layer have limited semi-permanent storage that allows them to store the received data temporarily for analysis and then send the source devices the needful feedbacks.

3.4 Mathematical model

The mathematical model of our scheme is described in this section. The operations and all entities involved are defined. It is assumed that the number of nodes distributed within the network in Tier 1 will not change during network operation. In addition, the clusters covered all the nodes in their domains. The physical components of the proposed scheme are defined as follows:

3.4.1 Physical components of Tier 1

Normal nodes (N_S): A normal node denoted by N is expressed as tuples, as follows.

$$N_S = \langle N_{\mathrm{id}}, N_{\mathrm{pc}}, N_{te}, N_{\mathrm{ln}}, N_{\mathrm{cr}} \rangle. \tag{4}$$

Each of Eq. (4) parameters is defined as follows.

Definition 1   (*Identification of N_S*) N_{id} is the unique identification (ID) assigned to each normal node in the network.

Definition 2 (*Present condition of N_S*) Present condition of N is denoted by N_{pc}. It shows the current state of the node, whether it is in an active or an inactive state. 1 is used to denote an active state, and 0 an inactive state. It is expressed as a Boolean as $N_{pc} = \{0, 1\}$.

Definition 3 (*Event type of N_S*) The type of environmental data (event) N_s sense is denoted by N_{te}. It is expressed as $N_{te} = \{N_1^{te}, N_2^{te}, N_3^{te}, \ldots, N_k^{te}\}$, where N_{te} denotes a set of environmental data being monitored by the N_s and k is the total number of events.

Definition 4 (*Geo-spatial location of N_S*) The geo-spatial location (N_{ln}) of N is expressed as a three-dimensional space as follows

$$N_{ln} = \langle x, y, z(t) \rangle \tag{5}$$

where x, y and z represent the horizontal, vertical, and height of a node, respectively. Time t denotes at which the N sends its data to a neighboring node/advanced node.

Definition 5 (*Component of N_S*) The component of N_S is denoted by N_{ct} which includes hardware specifications, the mode of operation, the sampling rate, and frequency. It is expressed as a six tuple as follows.

$$N_{cr} = \langle N_{ps}, N_{mc}, N_{rm}, N_{st}, N_{ch}, N_{dr}, N_{op} \rangle \tag{6}$$

where N_{ps} denotes power supply. In most cases, sensor nodes are powered by small batteries. It describes, in detail, the type of batteries a node uses for power supply (e.g., alkaline, lithium, or zinc-air batteries) and the size (AA or AAA). The tuple N_{mc} denotes the microcontroller; it contains one or more processor cores along with memory and programmable input/ output peripherals. The details of the processor core include processor speed, cache memory, and bus specifications. The tuple N_{st} represents the different types of sensors that are used as subunits of a sensor node. The tuple N_{ch} denotes the sensor node's hardware communication type used for wireless communication—this includes ZigBee, radio frequency identification (RFID), and bluetooth. The tuple N_{dr} denotes the data rate of a sensor node. The tuple N_{op} denotes the operating system (OS) a sensor node uses, which could be TinyOS, Mantis, or Contiki OS [33]. The application and application instances of the sensor nodes are defined as a tuple below.

Definition 6 (*Application of N_S*) The application of a sensor node (A_{pp}) is expressed as a four tuple, as follows.

$$A_{pp} = \langle A_{id}, A_{tp}, A_{cp}, A(t) \rangle \tag{7}$$

where the tuple A_{id} denotes the unique identification (ID) of an application. The tuple A_{tp} denotes the type for which the application is applied (e.g., smart city, healthcare, automobile, home security, or industrial applications). The tuple A_{cp} denotes the minimum component the system needs to successfully run the application, including the OS, processor, RAM, and secondary storage. The tuple $A(t)$ denotes the time stamp (second) at which the node senses the environmental data.

Algorithm 1: Algorithm for uniform distribution
of Advanced nodes

Begin
1: Given a set of $\overline{N}_A = \{N_1, N_2, \ldots \ldots \ldots \ldots, N_K\}$
   Advanced nodes   distributed into an $M \times M$
   meter square area;
2: initialize individual Advanced nodes;
3: Input K;   % K is the desired number of clusters
4: **for** $k = 1 : K$
5:    $x(k) = \text{rand}(1,1) * A$
6:    $y(k) = \text{rand}(1,1) * A$
7:   **end**
8: voronoi (x, y)
9: $N_K = \bigcup_{k=1}^{K} C_k, \ K < C; \quad // \ C_k$ is a cluster

10: $C_k \neq \emptyset; \quad k \in \{1, K\};$
11: $l = 1;$
12: **while** $K - l > 1$ **do**
13: **select**
14: $C_k, C_{k+1} \in N_{K-l+1};$
15: $N_{K-l} = (N_{K-l+1}\{(N_{K+l})\}) \backslash \{C_k, C_{k+1}\};$
16: $l = l + 1;$
17: **end while**
18: **for** $i = 1$ to P $// P$ denotes number of nodes
        per cluster
19: $N_i^k \rightarrow N_A^k$
20: **if** number of nodes in C_k equals to p **then**
21: Go to 4
22: **end for**
End

Definition 7 (*Application instance*) The application instance, I_A, is expressed as a five tuple, as follows.

$$I_A = \langle I_{id}, N_{id}, A_{id}, I_{rq}, I_{pc} \rangle \tag{8}$$

where I_{id} denotes the application instance ID which is generated and assigned by the system. The tuples N_{id} and A_{id} were defined earlier. The tuple I_{rq} denotes the resource requirement of the I_A, which may be in terms of data processing and analysis ability for healthcare applications, network bandwidth for streaming applications or processing, and storage power for automobile applications. It is possible for many instances of an application to run concurrently on a sensor node, with the unique IDs distinguishing the instances from one another.

Tier 1 is divided into finite clusters C_K, and each cluster contains, on average, the same number of randomly distributed nodes. The sensor network A is expressed as follows.

$$A = \sum_{k=1}^{K} C_k \left(\sum_{i=1}^{p} N_i \right) \tag{9}$$

where p denotes the average number of nodes in a cluster C, while K denotes the number of clusters which is equivalent to the number of advanced nodes in the network. The number of p changes due to the death and introduction of new nodes in the cluster. The area covered by a cluster is mathematically expressed as a five tuple, as follows.

$$C = \left\langle C_k, N_A^k, G, p, M_{\mathrm{id}} \right\rangle \tag{10}$$

where C_k is a cluster k, such that $k = 1, 2, 3, ..., K$, N_A^k denotes an advanced node k, the tuple G denotes the area covered by a cluster, the tuple p was defined earlier, and mapping of a sensor node to its corresponding advanced node is denoted by M_{id}.

Property 1   *The relationship from the set of $\overline{N}_s$ to the $\overline{N}_A$ in the network is many-to-one mapping* (*i.e., many nodes transmit to the corresponding advanced node*).

$$f : \overline{N}_s \rightarrow \overline{N}_A \tag{11}$$

Definition 8   (*Advanced node device*) The advanced node is defined in terms of its types as a three tuple, as follows.

$$D = D_{\mathrm{id}}, D_{\mathrm{cp}}, D_{\mathrm{tp}} \tag{12}$$

where D_{id} denotes the ID of the advanced node, D_{cp} denotes the component of the device node (i.e., hardware and related specifications), and D_{tp} denotes the type of node.

Property 2   *The relationship between the advanced nodes and fog nodes is based on injective mapping and is represented as follows.*

$$f : \overline{N}_A \rightarrow \overline{N}_F \tag{13}$$

Definition 9   (*Fog instance*) The fog nodes' (middle layer) services, denoted by F, are represented by a five tuple and expressed as follows.

$$F = \langle F_{\mathrm{id}}, F_U, F_{\mathrm{RI}}, F_{\mathrm{PI}}, F_{SA} \rangle \tag{14}$$

where the tuple F_{id} denotes the ID of a fog node, the tuple F_U denotes purpose, F_{RI} represents Required Interface, F_{PI} denotes Provided Interface, and the last tuple denotes the Set of Attributes. Now that all the physical components of the architecture of the WSNs fog-based model have been defined, we want to show the relationship between the three tiers.

First, we want to show the relationship between the normal nodes (N_S) and advanced nodes (N_A) using the method of contraction.

Proposition 1   (*Subjective mapping*) *Given the set of normal nodes $\overline{N}_S$ and advance node $\overline{N}_A$, the number of $\overline{N}_S$ mapped to C_K, where C_K denotes the total number of clusters in the network.*

Proof   We prove this by using the method of contraction. Suppose there were such a surjection as $f : \overline{N}_s \rightarrow \overline{N}_A$.

We assume

$$N_S \neq N_i + N_k \quad \forall\, i \in \overline{N}_s : i = 1, 2, \ldots, p; \quad k \in \overline{N}_A : k = 1, 2, \ldots, K \tag{15}$$

There exists at least a pair (N_i, N_k), such that:

$$N_i, N_k \in C_k, \quad N_i \cap N_k \neq \Phi \tag{16}$$

There exists every normal node belonging to an advanced node in each cluster C_k, a contradiction to Proposition 1, which disproves our assumption. This concludes the proof.□

Proposition 2 (*Injective mapping*) *Given the set of advanced nodes $\overline{N}_A$ and the set of fog nodes $\overline{N}_F$, the mapping of $\overline{N}_A$ into $\overline{N}_F$ is null.*

Proof This will also be proved by means of contradiction. Suppose there were such a function as $f : \overline{N}_A \to \overline{N}_F$.

We assume that there exist $N_k, k \in \overline{N}_A$; $N_f, f \in \overline{N}_F$ where F denotes the number of fog nodes, such that $N_k \cap N_f \neq \Phi$.

There exist N_k, N_f such that

$$N_k, N_f \in D\big(f\big(\overline{N}_A\big) = f\big(\overline{N}_F\big)\big)$$
$$\Rightarrow N_k \neq N_f$$

This disproves the injectivity of Property 2. Thus, the assumption is not valid. This concludes the proof.□

3.5 Data transmission between fog nodes and data center

Our network model contains N_F deployed at the edge of the network, at the middle layer (Tier 2) of our model. N_F at the edge of the network communicates directly with N_A in Tier 1. Each fog node at the edge of the network receives data from N_A (aggregate data in each round) and transmits it to the next fog node in the chain.

Algorithm 2 is developed to transmit aggregated data between N_F and cloud computing. An algorithm [21] is adopted to form a chain of fog nodes, and it enables the nodes to communicate with their neighboring nodes in the chain. The fog nodes in the chain receive data from the advanced nodes and perform pre-processing on the received data, and these data move between N_F, with each fog node sending its data to the leader fog node. The leader fog node receives and aggregates all data received from N_F and finally transmits these to the cloud computing for further processing. The algorithm shows the steps of the data transmission—from fog nodes to cloud computing. Fog node chain and its communications, as shown in Fig. 1, are constructed by choosing the furthest fog node from the cloud computing using Eqs. (15) and (16), to ensure that fog nodes further from the cloud computing have close neighbors. The neighboring distances increase progressively since fog nodes already on the chain cannot be revisited. Equation (17) is used to connect nodes to their closest neighbors. A fog node with the shortest distance to the cloud computing in Tier 2 is selected as the leader, and its main

duty is to transmit aggregated data to cloud computing. Constructing a chain among N_F and selecting a node as the leader—to receive and forward aggregated data instead of each fog node transmitting its own data directly to the cloud computing, could minimize energy dissipation during data transmission in the network. Thus, in order to choose a FN as a leader, the distance of FNs to the cloud and their energy need to be considered.

$$d_{D\_center} = \sqrt{(x \cdot N_F - x \cdot D_{center})^2 + (y \cdot N_F - y \cdot D_{center})^2} \tag{17}$$

$$FNode_{Edge} = \max_{F(i) \in \overline{N}_F} (d_{D\_center}F(i)), \quad \forall F(i) \in \overline{N}_F \tag{18}$$

$$d_{N_F}(Fi) = \sqrt{(x \cdot F(i) - x \cdot F(j))^2 + (y \cdot F(i) - y \cdot F(j))^2}, \quad \forall F(j) \in \{\overline{N}_F - F(i)\}. \tag{19}$$

Algorithm 2 Data Transmission from Fog Node to Cloud

	Begin
	Input: fog node (N_F)
	Output: Optimal path of N_F to cloud computing
1:	Compute distance of N_F to cloud computing using equation (16)
2:	**for** $i = 1$ to K **do**
3:	$F(i)$ is the furthest fog node from cloud computing using equation (17)
4:	Add $F(i)$ to chain
5:	**If** the neighbor of $F(i)$ has not been visited **then**
6:	Add to chain list **else**
7:	move to next neighbouring node
8:	**endif**
9:	Compute the distance $d_{N_F}(Fi)$ between the fog nodes using equation (18)
10:	Select minimum distance Min($d_{N_F}(Fi)$) in the chain list
11:	Transmit aggregated sensed data from leader fog node to cloud computing
12:	**end for**
13:	**End**

3.6 Improved ACO algorithms

There are many routing paths between the fog nodes and the data center. It is, however, necessary to transmit the aggregated data from the leader fog node to the data center along the shortest, most efficient, and reliable path. Algorithm 3 is developed to construct the optimal path for data transmission between N_F and the data center. The aim is to determine the shortest possible path. An ACO routing technique is used to find the shortest distance between the leader fog node and the data center. ACO is a branch of optimization modeled algorithms based on the behavior of ants

in a colony and is a subclass of computational intelligence (IC) paradigms that aid in determining optimal solutions to optimization problems [15]. The ACO model was first proposed by Dorigo [16, 34]. Since then, the model has been widely studied and improved. The idea comes from observing ants' foraging behavior—how they find the shortest path between the food source and their nest. When searching for food, ants first explore the area surrounding their nest randomly. While moving, they deposit a chemical substance called pheromone on the paths as they search for food, forming pheromone trails. Thus, when other ants search for food, they can smell the pheromones that have been deposited on the paths, and tend to choose a path marked by strong concentrations of pheromone.

However, the conventional ACO schemes do not differentiate between the various types of data packets transmitting over a network. All data packets are transmitted in the same way to a destination, as these schemes are majorly designed to minimize congestion and improve network performance. On the other hand, WSNs in IoT environment have to support multimedia data transmission, requiring a different level of quality of service (QoS). This work divides QoS into three services (Q_s).

Q_{s_1} Q_{s_1} is guaranteed service, and it provides safe end-to-end delay guarantees. This service warranties that data packets will reach the receiver node at the right time zero data packets loss.

Q_{s_2} Q_{s_2} is a controlled load-balance service. The service is used where likelihood of time delays will happen. In addition, when there is congestion in the network, the network can provide services just as if there was no congestion in the network.

Q_{s_3} Q_{s_3} is the best-effort service. It refers to an Internet delivery service where the network does not provide any guarantees on the time delay limit the data packets will be delivered in the network.

We made modifications to the proposed traditional ACO algorithms in order to improve its performance and to accommodate the heterogeneity of our network. These modifications are explained as follows.

It is assumed that the success transmission rates of all the links between the advanced nodes and fog nodes are all 100%.

An advanced node has low communication performance if the data packets transmission success rate is less than 10%. Conversely, it has high communication performance if the data packets transmission rate is more than 90%.

Given a network topological graph $G = (K, E)$, where K denotes the number of advanced nodes in the network, and E represents the set of the links connecting two (advanced and fog) nodes in the network.

If the forward link of the node k is denoted by T_f and T_r denotes the reverse link of the node j. The link that directly joins nodes i and j is represented in Eq. (18)

$$l = \left(\overline{N}_A(k), \overline{N}_F(j) \right) \tag{20}$$

Each link $l \in E$ represents the success probability of data transmission on the link l.

For a path T_{path} and a link $l \in E$, the distance between an advanced node and a fog node is the sum of the single hop links and expressed as follows.

$$T_{\text{path}} = \sum_{u=1}^{w} l_u \tag{21}$$

Equation (21) shows that the smaller the T_{path}, the nearer the advanced node is to the fog node. The node should be used as a data packets forwarding node. We use $R(k,j)$ to denote the relationship between a sender node (k) and receiver node (j), as expressed in Eq. (22).

$$R(k,j) = \begin{cases} 1 & \text{if} \quad l_u \in T_{\text{f}}(k) \\ -1 & \text{if} \quad l_u \in T_{\text{r}}(k) \\ 0, & \text{others} \end{cases} \tag{22}$$

where $T_{\text{f}}(k)$ and $T_{\text{r}}(k)$ denote forward link and reverse link, respectively, of the node k.

3.7 Local pheromone update

The choice of selecting an optimal path is made based on the transfer probability decision rule in Eq. (21).

$$c = \begin{cases} \arg\max\limits_{j=1,2,\ldots,m} \left\{ \left[\lambda_{kj}^{Q_{s\_a}}\right]^{\alpha} * \left[\eta_{kj}^{Q_{s\_a}}\right]^{\beta} * R(k,j) \right\}, & q \leq q_o \\ C, & q > q_o \end{cases}$$

$$C = \begin{cases} \dfrac{\left[\lambda_{kj}^{Q_{s\_a}}\right]^{\alpha} * \left[\eta_{kj}^{Q_{s\_a}}\right]^{\beta} * R(k,j)}{\sum_{j \in T_{path,}} \left[\lambda_{kj}^{Q_{s\_a}}\right]^{\alpha} * \left[\eta_{kj}^{Q_{s\_a}}\right]^{\beta}}, & j \in \text{allowed}_u \\ 0, & j \notin \text{allowed}_u \end{cases} \tag{23}$$

where $\lambda_{kj}^{Q_{s\_a}}$ denotes the pheromone service, $\eta_{kj}^{Q_{s\_a}}$ denotes the local heuristic value of the path between the sender node and the receiver node, a denotes the QoS type, α and β are control parameters used to regulate the relative weight of the pheromone trail and the heuristic value, respectively. The q is a random variable ranging between 0 and 1 (i.e., [0, 1]), and q_o ($0 \leq q_o \leq 1$) is a given parameter. $j \in \text{allowed}_u$ for all $u = 1, 2, 3, \ldots, w$ are paths that can be selected by node k in the next step.

3.8 Global pheromone update

Once the communication paths between the sender nodes and receiver nodes have been completed by the search ants, Eq. (24) is used to choose an optima path

$$Z_O^{Q_{s\_a}} = \max \left\{ \left(Z_{\text{path}}^{Q_{s\_a}}\right)_u \right\} \tag{24}$$

where $\left(Z_{\text{path}}^{Q_{s\_a}}\right)_u$ is the path utility value. For a path between nodes k, j, the pheromones are updated as expressed in equation

$$\lambda_{kj}^{Q_{s\_a}} = (1-\rho)\lambda_{kj}^{Q_{s\_a}} + \rho\Delta\lambda_{kj}^{Q_{s\_a}} \tag{25}$$

where ρ is the pheromone evaporation factor which serves to diminish the intensity of existing trail over time, $\Delta\lambda_{kj}^{Q_{s\_a}}$ is the pheromone increment on the path between the advanced node and fog node and it is expressed as follows.

$$\Delta\lambda_{kj}^{Q_{s\_a}} = \begin{cases} \dfrac{\varpi * Z_{\text{path}}^{Q_{s\_a}}}{T_{\text{path}}}, & j \in \text{allowed}_u \\ 0, & j \notin \text{allowed}_u \end{cases} \tag{26}$$

where ϖ is an adjustment coefficient, and T_{path} denotes path length. Equation (26) is a pheromone update rule for the forward ants used to create the paths between the F_L and the $D_{\_center}$. φ^k denotes the contraindication list of the k.

	Algorithm 3 Fog nodes routing based on ACO
	Begin
	Input: fog nodes (N_F)
	Output: Construct optima path to fog nodes network
1:	Initialize $\lambda_{kj}^{Q_{s\_a}}$ and $a \leftarrow 1$
2:	**While** termination conditions is not met **do**
3:	$a \leftarrow a + 1$
4:	**for** $k = 1$ to K
5:	k is positioned at F_L
6:	$F(k) \leftarrow F_L$
8:	$R^k \leftarrow \emptyset$; $\varphi^k \leftarrow \emptyset$
9:	**While** $F(k) \neq F_L$ **do**
10:	**if** $C(F(k)) - \varphi^k \neq \emptyset$ **then**
11:	Choose $(F(j)$ from $C(F(k)) - \varphi^k$ to move based on the probabilistic transition rule in equation (19)
12:	$R^k \leftarrow R^k \cup \{F(i)\};\ \varphi^k \leftarrow \varphi^k \cup \{F(i)\};$ $k \leftarrow$ j
13:	**else**
14:	Back to the previous-hop of $F(k)$
15:	**end if**
16:	**end while**
17:	Compute the paths between the nodes using equation (23)
18:	Determine the value of $\Delta\lambda_{kj}^{Q_{s\_a}}$ using equation (26)
19:	**end for**
20:	Update the number of pheromone values using equation (25)
21:	Compare the values of solutions obtained
22:	**end while**
23:	Return the optimal solution
24:	Choose the best solution as the output
25:	**End**

4 Experimental results and discussion

The performance of our proposed scheme is presented in this section. We simulated our scheme (EEHFC) and compared the results with the P-SEP [18], EDCF [27], and RABACO [13] schemes. Network Simulator 2 (NS2) was used to evaluate the proposed algorithms. The network parameters used in our simulation are presented in Table 1, and Table 2 contains a list of abbreviations. The network contained three types of nodes: normal sensor nodes, advanced nodes, and fog nodes; values $h = 0.05$ and $h = 0.1$ show the fraction of advanced sensor nodes in the network. Thus, $h = 0.05$ means the proportion of advanced nodes to normal sensor nodes is 5% of the total normal sensor nodes, for a network with 100 sensor nodes. The performance of EEHFC, P-SEP, EDCF, and RABACO in the network area 100×100 m^2 was evaluated. All simulations ran 80 times, to generate different topologies to obtain accurate results. We determined the average values of the results and then used them to plot the figures. The simulation time (round) for each experiment was 5000 s. A round was defined as an equal period for which every node transmitted its sensed data to a data center, located outside the network, and at a relatively far distance from the WSNs. The number of fog nodes was considered to be five (5), and each had higher energy than the advanced nodes in the network.

4.1 Network lifetime

We studied the number of alive nodes of EEHFC, P-SEP, EDCF, and RABACO when the $a = 3$, $h = 0.1$. They dissipated their energy gradually as the simulation time increased (see Fig. 2). The first node died (FND) in EEHFC, P-SEP, EDCF, and RABACO at 1561, 1519, 1421, and 1387 rounds, respectively. In addition, half of the nodes died (HND) in EEHFC, P-SEP, EDCF, and RABACO at 2468, 2006, 1579, and 1536 rounds, respectively, while the last node died (LND) in EEHFC, P-SEP, EDCF, and RABACO at 4039, 3017, 3472, on 1864 rounds, respectively. In all the scenarios, the proposed algorithm had the highest number of alive nodes and prolonged the network lifetime beyond that of the selected algorithms.

4.2 Number of packets

We investigated the number of data packets transmitted to the final destination in each algorithm. EEHFC delivered more sensed data to the data center than all the other algorithms in terms of reliability, stability period, availability, and network lifetime as depicted in Fig. 3. It had the highest data delivery ratio. This result proves that our scheme not only performed better in a small network but also in a large network.

4.3 Total energy consumption of sensor nodes

Sensor nodes dissipate energy gradually as the network size increases in all the schemes and the energy curve is shown in Fig. 4. The RABACO algorithm dissipated the most energy during data transmission—the reason being that the algorithm kept to multiple paths and did not consider the residual energy of the neighboring node, which created an energy hole in the network. There was a high probability that the remaining nodes would transmit across a long distance and hence consume more energy. In addition,

Table 1 Simulation parameters for the network

Parameter	Value
Number of nodes	100
Network size	100×100 m^2
Initial energy of normal node	0.5 J
Initial energy of advanced node	1.5 J
Number of ants	5
α	2
β	2
Transmit data rate	250 kbps
ρ	0.2
r (round)	5000
$\lambda_{ij}(0)$	0.01
Packet size	512 bytes
ε_{amp}	0.0012 pJ/bit/m^4
E_{elec}	40 nJ/bit
E_{DA}	5 nJ/bit
MAC layer protocol	IEEE 802.11

Table 2 List of abbreviations

Abbreviations	Meanings
WSNs	Wireless sensor networks
CHs	Cluster heads
MDC	Mobile data collector
N_S	Sensor node
N_F	Fog nodes
N_A	Advanced nodes
A	Total number of sensor nodes
P	Normal nodes in the network
K	Fraction of advanced nodes
E_f	Initial energy of a normal node
h	Energy multiple factor
N_{pc}	Present condition of N
N_{te}	Type of environmental data sensed by N
N_{te}	Set of environmental data being monitored
k	Total number of events
N_{ln}	Location of N_S
N_{ct}	Component of N_S
N_{st}	Types of sensor nodes
N_{ch}	Sensor node's hardware communication type
A_{pp}	Application of sensor nodes
I_A	Application instance
A	Total nodes in the network
C	Area covered by a cluster
D	Type of advanced node
F	Fog node
$\overline{N}_S$	Set of normal nodes
$\overline{N}_A$	Set of advanced nodes
$\overline{N}_F$	Set of fog nodes

the energy dissipated by P-SEP was less than that of RABACO, because it transmitted along an alternate path whenever there was congestion along the primary path. Thus, the energy consumption in EEHFC is lowest because the scheme selects nodes on paths with few hops to transmit the data packets. Finally, overheads that are usually involved in the selection of CHs are eliminated in the proposed scheme. It is able to optimize the routing process to maintain load balance in the network.

4.4 Average energy dissipation in the network

Average energy dissipation of advanced nodes varying the number from two to six follows the curve shown in Fig. 5. The average energy dissipation was highest for two advanced nodes—the reason being that each normal node transmits to advanced nodes across a long distance, thereby using more energy to transmit. However, when the number increases to six, the average energy dissipation was lowest, since the nodes transmitted across a short distance and conserved more energy. We observe that after 3000 rounds, the nodes dissipate more energy. This is due to the congestion and retransmission of data packets. Finally, after the number of rounds, energy holes created in the network will increase the nodes' energy dissipation.

4.5 Network lifetimes with a varying number of advanced nodes

Table 3 shows the values of the network lifetimes obtained for each of the four algorithms. The algorithms have a different number of alive nodes after a certain number of rounds ($\alpha = 2$, $h = 0.05$). Before FND, the EEHFC increased by 34.8%, 53.4%, and 82.9% in P-SEP, EDCF, and RABACO, respectively. In addition, it increased by 21.0%, 33.5%, and 40.4% in P-SEP, EDCF, and RABACO, respectively, before HND. Finally, it increased by 10.9%, 20.3%, and 33.8%, respectively, in P-SEP, EDCF, and RABACO, before LND.

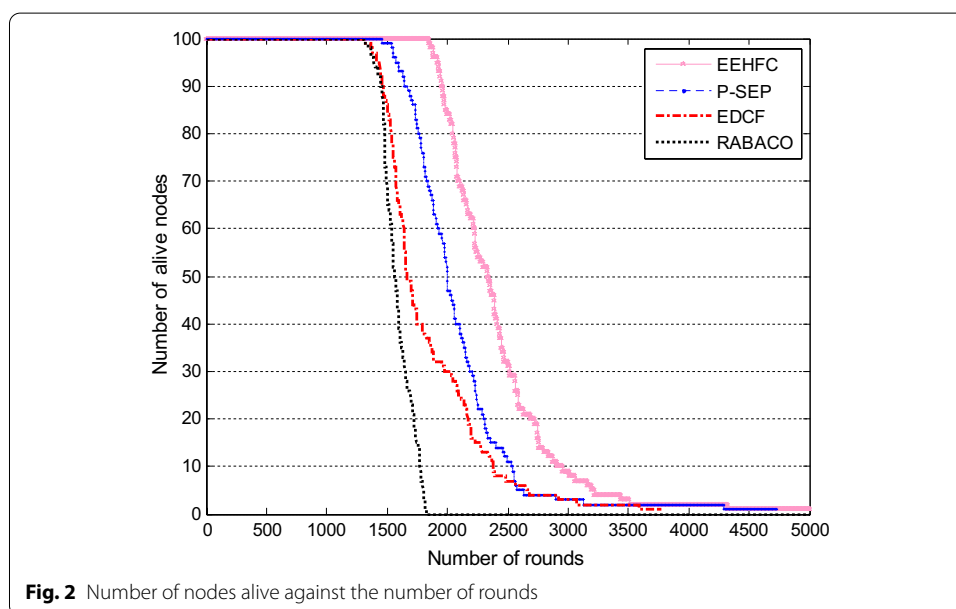### 4.6 Performance of QoS levels for different schemes

The performance of selected schemes and our scheme are compared based on the QoS levels. The results are presented in Fig. 6. The results show that regardless of the energy consumption, the number of data packets transmitted, data packets reliability, the average of QoS values of the nodes of the EEHFC scheme perform better than other selected schemes. Hence, it is an improvement over related schemes.

4.7 Data packets transmission loss rate

We perform experiments on data packets transmission loss rate for the selected schemes and the proposed scheme under different QoS levels for sensor nodes. We observe that when the distance between a sender node and the destination node is short, the data packets transmission loss rates are almost the same for all the schemes. Conversely, as the transmission distance between a sender node and the destination node increases, data packets transmission loss rate likewise increases. The reason is that as the distance between the nodes increases, the number of hops that are required for reliable data transmission equally increases. As shown in the figure, the data packets loss rate increases in all the schemes. Percentages for data packets loss rate for the different

Table 3 Performance comparison of algorithms under network size 100 with $r = 5000$ and $a = 2$

Protocol	FND	HND	LND
EEHFC ($a = 2, h = 0.1$)	654	1753	2213
EEHFC ($a = 2, h = 0.05$)	496	1697	2029
P-SEP ($a = 2, h = 0.1$)	427	1581	1953
P-SEP ($a = 2, h = 0.05$)	368	1402	1829
EDCF ($a = 2, h = 0.1$)	301	1387	1752
EDCF ($a = 2, h = 0.05$)	289	1271	1686
RABACO ($a = 2, h = 0.1$)	233	1223	1578
RABACO ($a = 2, h = 0.05$)	188	1209	1516

Fig. 2 Number of nodes alive against the number of rounds

schemes are shown in Table 4. EEHFC scheme performs better than other selected schemes. It can select the optimal path for its data transmission, minimizing the packet loss rate.

5 Conclusion

EEHFC is proposed as a fog-based algorithm for WSNs. This algorithm presents efficient methods for data transmission in the network. A three-tier architecture was developed for the proposed scheme, consisting of three types of nodes—normal sensor nodes, advanced nodes, and fog nodes. EEHFC presented a hierarchical routing architecture for data transmission from normal sensor nodes to the data center through fog nodes. In addition, an algorithm was designed for the uniform distribution of advanced nodes within the network, to minimize the transmission distance of each sensor node. The advanced nodes forward gathered data from the normal sensor nodes of their clusters to the fog nodes. Thereafter, the fog nodes performed pre-processing on the received data and stored them locally, which invariably reduced the processing of a large volume

Fig. 3 Number of packets transmitted to data center

Fig. 4 Total energy consumption of node
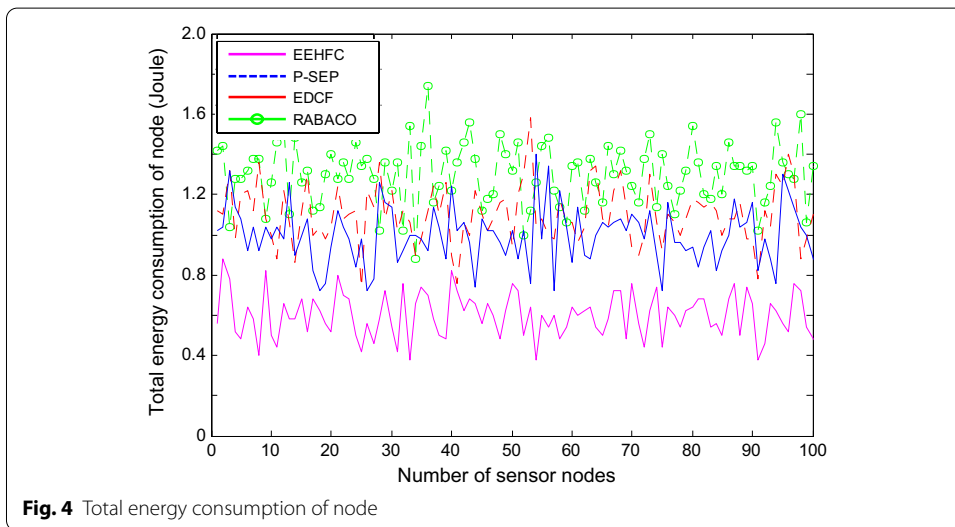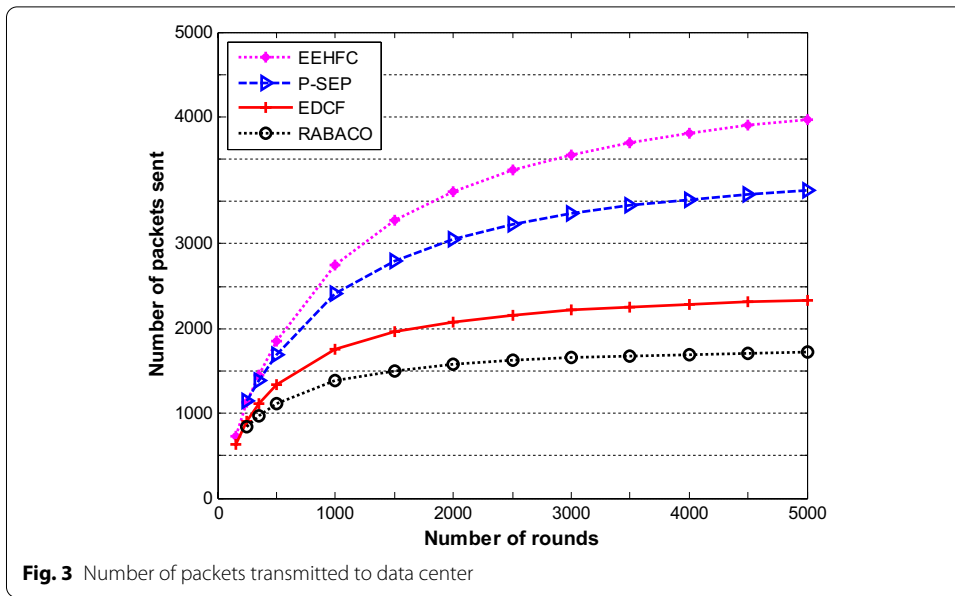
of data at the data center. Only sensed data that cannot be processed in fog nodes are transmitted to the data center. Moreover, the inclusion of various levels of energy in our scheme causes the network to be heterogeneous and helps maintain load balancing in the network. The proposed scheme is energy efficient and extends the network lifetime due to the higher energy of fog nodes. To increase data reliability at the data center, we developed algorithm 2 for data transmission from the fog nodes to the data center. Algorithm 3 was designed to construct the optimal and most efficient path between the fog nodes and the data center, using the ACO approach. The proposed EEHFC approach is energy-efficient, has high data reliability and availability, and prolongs the network lifespan—more so that related approaches, such as the P-SEP, EDCF, and RABACO schemes. However, since the F_s store only important data and the individual node has a balanced
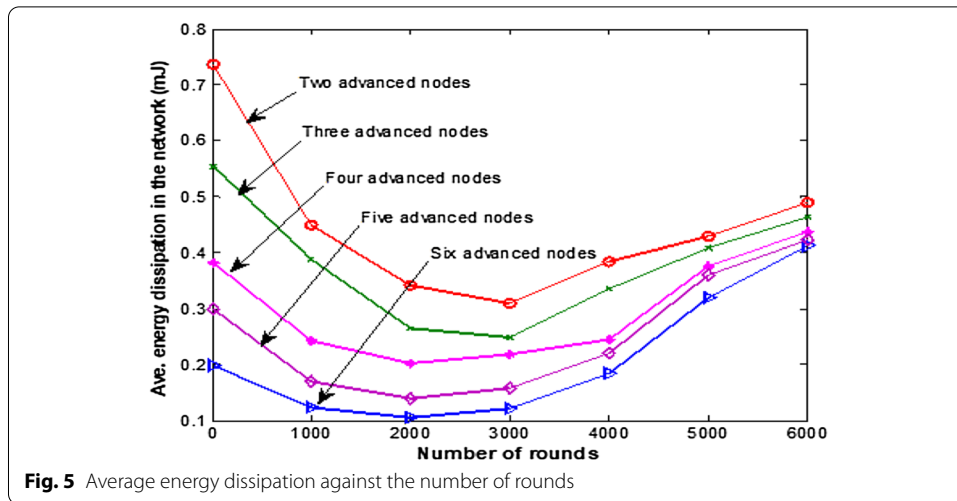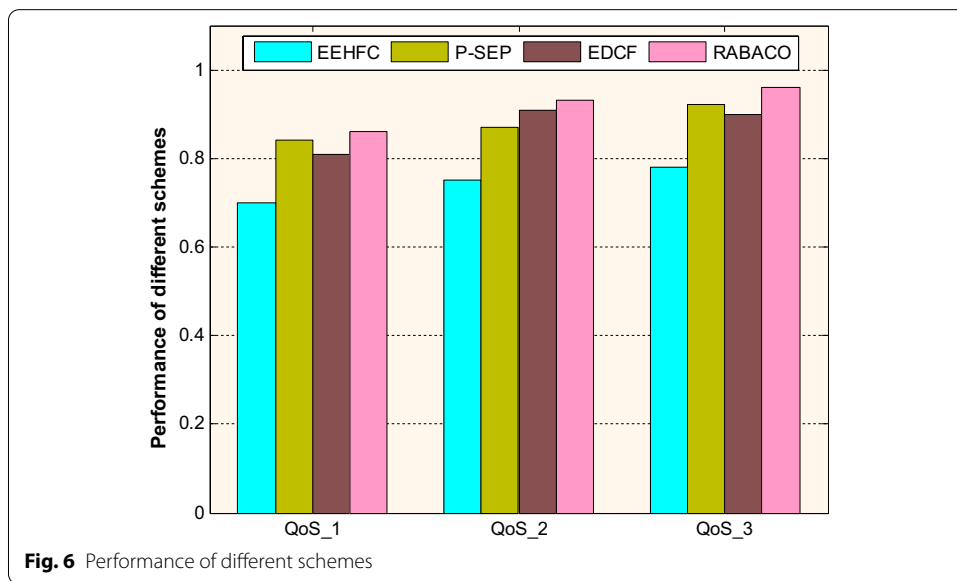
Fig. 5 Average energy dissipation against the number of rounds

Table 4 Rate of data packets loss of selected schemes

	Q_{s_1}	Q_{s_2}	Q_{s_3}
EEHFC (%)	21	23	22
P-SEP (%)	23	26	24
EDCT (%)	42	45	41
RABACO (%)	53	52	54

Fig. 6 Performance of different schemes

processing load, optimal resource allocation for them is one of the drawbacks of this approach.

In addition, data transmission in the network is open and is susceptible to different attacks. There is a need to provide adequate security solutions to these attacks. These will be looked into in our future research.

Fig. 7 Data traffic in a network with 100 sensor nodes and transmission range 75 m
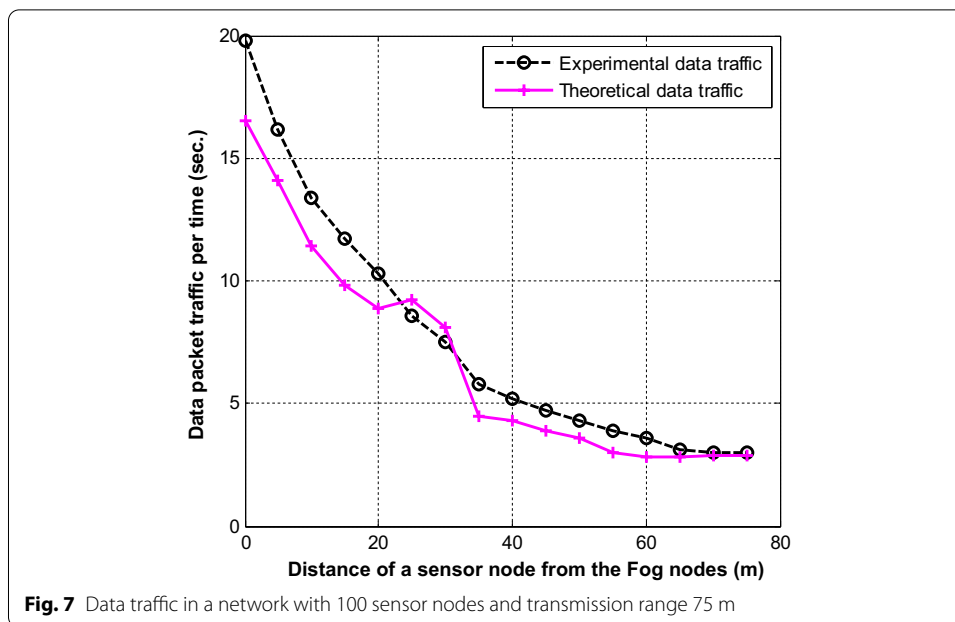
We are aware that sensed data generated by sensor nodes can change due to environmental factors and hardware failures which might cause sensor nodes to produce incomplete data or erred. The end-users need to be mindful of the uncertainty level of the collected data and various results obtained in the research.

5.1 Data traffic load distribution

We performed an experiment using the model developed in Sect. 3.3 to determine the energy consumption of sensor nodes based on the data traffic and distance of each node from fog nodes. In total, 100 sensor nodes are randomly distributed and set each node transmission range to 75 m. The improved ACO routing algorithm 3 is also used in this simulation, and the packet transmission rate is 1-bit per second. The results of the experiment are presented in Fig. 7. We can see that sensor nodes with short distance from the fog nodes receive higher traffic load than those that are far from them. The reason is that the closer nodes act as relay nodes for other nodes that are far; hence, there is high possibility for these nodes to use up the limited energy and die faster than those nodes that are far as shown in the figure. It can be seen that the experimental traffic load distribution agrees with the theoretical traffic load distribution when there is number of sensor nodes. This consolidates the practical importance of the energy model presented in Sect. 3.3.

References

1. F. Karray, M.W. Jmal, A. Garcia-Ortiz et al., A comprehensive survey on wireless sensor node hardware platforms. Comput. Netw. **144**, 89–110 (2018)
2. A. Ali, Y. Ming, S. Chakraborty et al., A comprehensive survey on real-time applications of WSN. Fut. Internet. **9**(4), 77 (2017)
3. S.J. Ramson, D.J. Moni, Applications of wireless sensor networks—a survey, in *Proceedings—2017 International Conference on Innovations in Electrical, Electronics, Instrumentation and Media Technology* (*ICEEIMT)* (2017), pp. 325–329
4. S.K. Gawali, M.K. Deshmukh, Energy autonomy in IoT technologies. Energy Proc. **156**, 222–226 (2019)
5. A. Yassine, S. Singh, M.S. Hossain et al., IoT big data analytics for smart homes with fog and cloud computing. Fut. Gener. Comput. Syst. **91**, 563–573 (2019)
6. CISCO *Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are* (2015), pp. 1–6
7. S.P. Singh, A. Nayyar, R. Kumar et al., Fog computing: from architecture to edge computing and big data processing. J. Supercomput. **75**(4), 2070–2105 (2019)
8. S. Yi, Z. Hao, Z. Qin, et al., Fog computing: platform and applications, in *Proceedings—3rd IEEE Workshop on Hot Topics in Web Systems and Technologies* (*HotWeb)* (Washington, DC, USA, 12–13 Nov. 2015), pp. 73–78
9. Z. Wen, R. Yang, P. Garraghan et al., Fog orchestration for internet of things services. IEEE Internet Comput. **21**(2), 16–24 (2017)
10. N. Chen, Y. Yang, T. Zhang et al., Fog as a service technology. IEEE Commun. Mag. **56**(11), 95–101 (2018)
11. Z. Mahmood, Fog computing: concepts, frameworks and technologies, in *Fog Computing: Concepts, Principles and Related Paradigms* (Springer, Berlin, 2018), pp. 1–21
12. A.V. Dastjerdi, R. Buyya, Fog computing: helping the internet of things realize its potential. Computer **49**(8), 112–116 (2016)
13. Y. Sun, W. Dong, Y. Chen, An improved routing algorithm based on ant colony optimization in wireless sensor networks. IEEE Commun. Lett. **21**(6), 1317–1320 (2017)
14. V. Sharma, A. Grover, A modified ant colony optimization algorithm (mACO) for energy efficient wireless sensor networks. Opt. Int. J. Light Electron Opt. **127**(4), 2169–2172 (2016)
15. A. Gonzalez-Pardo, J. Del Ser, D. Camacho, Comparative study of pheromone control heuristics in ACO algorithms for solving RCPSP problems. Appl. Soft Comput. **60**, 241–255 (2017)
16. M. Dorigo, T. Stützle, Ant colony optimization: overview and recent advances, in *Handbook of Metaheuristics* (Springer, Berlin, 2019), pp. 311–351
17. S. Abi, B. Benhala, H. Bouyghf, et al., A comparative study between ACO and DE techniques by numerical functions optimization, in *Proceedings—5th International Conference on Optimization and Applications* (*ICOA)* (2019), pp. 1–6
18. E.M. Borujeni, D. Rahbari, M. Nickray, Fog-based energy-efficient routing protocol for wireless sensor networks. J. Supercomput. **74**(12), 6831–6858 (2018)
19. W.B. Heinzelman, A. Chandrakasan, H. Balakrishnan, An application—specific protocol architecture for wireless microsensor networks. IEEE Trans. Wirel. Commun. **1**(4), 660–670 (2002)
20. Y. Zhou, N. Wang, W. Xiang, Clustering hierarchy protocol in wireless sensor networks using an improved PSO algorithm. IEEE Access. **5**, 2241–2253 (2016)
21. S. Lindsey, C. Raghavendra, PEGASIS: power efficient gathering in sensor information systems, in *Proceedings of the IEEE Aerospace Conference, Big Sky, MT. Montana*, vol. 3 (2002), pp. 1125–1130
22. W. Ke, O. Yangrui, J. Hong et al., Energy aware hierarchical cluster-based routing protocol for WSNs. J. China Univ. Posts Telecommun. **23**(4), 46–52 (2016)
23. G. Xie, K. Ota, M. Dong et al., Energy-efficient routing for mobile data collectors in wireless sensor networks with obstacles. Peer Peer Netw. Appl. **10**(3), 472–483 (2017)
24. N. Ayoub, M. Asad, M. Aslam, et al., MAHEE: multi-hop advance heterogeneity-aware energy efficient path planning algorithm for wireless sensor networks, in *Proceedings—2017 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing* (*PACRIM)* (2017), pp. 1–6
25. T. Qureshi, N. Javaid, A. Khan et al., BEENISH: balanced energy efficient network integrated super heterogeneous protocol for wireless sensor networks. Proc. Comput. Sci. **19**, 920–925 (2013)
26. L. Qing, Q. Zhu, M. Wang, Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks. Comput. Commun. **29**(12), 2230–2237 (2006)
27. R. Priyadarshi, P. Rawat, V. Nath, Energy dependent cluster formation in heterogeneous wireless sensor network. Microsyst. Technol. **25**(6), 2313–2321 (2019)
28. A. Mohajerani, D. Gharavian, An ant colony optimization based routing algorithm for extending network lifetime in wireless sensor networks. Wirel. Netw. **22**(8), 2637–2647 (2016)
29. A. Bagula, A.P. Abidoye, G.-A.L. Zodi, Service-aware clustering: an energy-efficient model for the internet-of-things. Sensors **16**(1), 1–9 (2015)
30. S. Fouchal, D. Mansouri, L. Mokdad et al., Recursive-clustering-based approach for denial of service (DoS) attacks in wireless sensors networks. Int. J. Commun. Syst. **28**(2), 309–324 (2015)
31. V. Sundararaj, S. Muthukumar, R. Kumar, An optimal cluster formation based energy efficient dynamic scheduling hybrid MAC protocol for heavy traffic load in wireless sensor networks. Comput. Secur. **77**, 277–288 (2018)

32. K.A. Darabkh, S.M. Odetallah, Z. Al-qudah et al., Energy-aware and density-based clustering and relaying protocol (EA-DB-CRP) for gathering data in wireless sensor networks. Appl. Soft Comput. **80**, 154–166 (2019)
33. T.V. Chien, H.N. Chan, T.N. Huu, A comparative study on operating system for wireless sensor networks, in *Proceedings—International Conference on Advanced Computer Science and Information Systems, Jakarta, Indonesia* (17–18 Dec. 2011), pp. 73–78
34. M. Dorigo, M. Birattari, Ant colony optimization, in *Encyclopedia of Machine Learning* (Springer, Berlin, 2010), pp. 36–39

Publisher's Note