



# Energy-efficient cluster head selection in wireless sensor networks-based internet of things (IoT) using fuzzy-based Harris hawks optimization

Sankar Sennan<sup>1</sup> · Somula Ramasubbareddy<sup>2</sup> · Rajesh Kumar Dhanaraj<sup>3</sup> · Anand Nayyar<sup>4</sup> · Balamurugan Balusamy<sup>5</sup>

Accepted: 1 June 2024 / Published online: 21 June 2024

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

## Abstract

The Internet of things (IoT) has become a cornerstone of the fourth industrial revolution. IoT sensor devices in the network are provisioned with limited resources, such as little processing speed, minimal computing capacity, and less power. Furthermore, IoT devices are battery-powered, which cannot provide battery sufficiently to some applications resulting in an energy scarcity problem. Clustering is an efficient method in IoT networks to save energy. Nodes can coordinate communication by selecting an optimal cluster head (CH) within the cluster and transmitting information to a central node or sink. The CH minimizes energy consumption associated with communication overhead and extends the overall lifespan of the network by facilitating coordination between clusters and the central server. Many existing optimization techniques have proposed CH selection to improve the network's lifespan but all the existing algorithms on CH selection are not practical due to the long convergence time. This research paper proposes a novel fuzzy-based Harris Hawks Optimization (FHHO) algorithm that chooses optimal CH considering Residual energy (RER) and distance between sink and node. The fitness function is evaluated using fuzzy logic over maximization and minimization network parameters. Extensive experimentations were conducted to test and validate the performance of proposed FHHO algorithm on MATLAB 2019a tool. And, the results stated that the proposed method FHHO has better results as compared to other CH selection techniques, namely, PSO-ECHS, FIGWO, and GWO-C, in network lifespan by 18–44% and throughput by 5–20%.

**Keywords** Internet of things · Computational intelligence · Clustering · Energy conservation · Fuzzy logic · Harris Hawks optimization

## 1 Introduction

Internet of Things (IoT) is a revolutionary innovation that has attracted increasing research interest in academia and industry in recent years [1–4]. IoT encompasses embedded system technologies that use wireless or wired connections as well as sensors, actuators, and physical objects that communicate with one another via internet [5–8]. According to various forecasts, more than 75 billion devices will be utilized in real time applications by 2025 [9–11]. Sensor nodes

✉ Rajesh Kumar Dhanaraj  
sangeraje@gmail.com

✉ Anand Nayyar  
anandnayyar@duytan.edu.vn

Sankar Sennan  
sankarcsharp@gmail.com

Somula Ramasubbareddy  
svramasubbareddy1219@gmail.com

Balamurugan Balusamy  
kadavulai@gmail.com

<sup>1</sup> Department of Computer Science and Engineering, Sona College of Technology, Salem, India

<sup>2</sup> Department of Information Technology, VNRVJIET, Hyderabad, India

<sup>3</sup> Symbiosis Institute of Computer Studies and Research (SICSR), Symbiosis International (Deemed University), Pune, India

<sup>4</sup> Graduate School, Faculty of Information Technology, Duy Tan University, Da Nang 550000, Viet Nam

<sup>5</sup> Shiv Nadar Institution of Eminence, Greater Noida, India

play a significant role in networks that can be operated without human interaction [12–15]. The sensor nodes are mainly used in IoT for industrial purposes to transfer collected data. IoT creates the ecosystem to achieve enhanced efficiency by communicating and sharing information with other devices. The IoT applications include the smart grid, smart lighting, smart door, self-driven vehicles, smart parking, smart wearable, smart farming, industrial automation, etc. [16–21].

Wireless Sensor Networks (WSNs) deploy nodes independently and spatially to monitor the environment or track real-time conditions. The network is created with various physical sensor nodes deployed randomly in the network area to collect scalar information. WSNs benefit from a low cost, compact size, and ease of deployment, making them one of the most effective technologies for real-time applications. Energy consumption of sensor nodes become critical issue in design of WSNs. The fundamental function of the WSN is to exchange collected data to the sink for analysis [22–25]. People lives can be improved with smart WSN technologies. WSNs is an enabling technology that can be applied in environmental monitoring, smart building, healthcare, precision agriculture and animal tracking, etc. Nevertheless, some WSNs contain a massive number of nodes, which are challenging to manage collectively. Therefore, an efficient algorithm is required to manage energy resources of nodes and improving network duration. Moreover, the WSN can be changed by external causes or a network system designer, subsequently affecting the network routing, localization, delay, Quality of Service (QoS), etc. as such, WSNs must be often redesigned due to its dynamic nature [26–29].

Routing plays a vital role in resource-constrained networks like WSN and IoT. Routing is the process of transferring data from one place to another. Due to redundant data transmission to sink by specific applications such as environmental and temperature monitoring [30–34], clustering can be used to overcome this problem. Cluster creation, CH election, and route formation are the three steps of the clustering protocol, whereby choosing the right CH is crucial to enhance a network's life. The cluster formation technique reduces energy consumption and increases network lifetime by managing the energy utilization of sensor nodes. The clustering process is achieved by dividing the deployment area into various sub-deployment areas. Each cluster consists of a group of sensor nodes and a CH node. CH collects and sends data from sensor nodes to sink nodes. The CH selection methods include a single routing metric, composite routing metric, as well as probability basis, mathematical models, and optimization algorithms [35].

Optimization algorithms are performed until the optimal solution is found. Some popular optimization algorithms are Evolutionary Algorithm (EA), Genetic Algorithm (GA), Fruit Fly Optimization (FFO), Butterfly Optimization Algorithm (BOA), Harmony Search Algorithm (HSA),

Firefly Algorithm (FA), etc. Particularly, the meta-heuristics approach is often used to obtain the optimal solution for CH selection [36] and, thus, increase the network's lifetime [37, 38].

This paper focuss on optimal CH selection to optimize network lifespan through a proposed novel fuzzy-based Harris Hawks Optimization (FHHO) algorithm. The fitness function is calculated to choose the right CH in the network by applying fuzzy logic over constraints RER and Distance. Thus, the FHHO algorithm improves the overall network performance by means of throughput, increasing the network's lifespan and effective energy consumption.

Objectives of the paper

The objectives of the paper are:

- To conduct an extensive analysis on related works towards CH selection that applied numerous optimization algorithms to improve network lifespan and throughput;
- To propose a novel FHHO algorithm to identify the optimal CH for increasing network longevity. And the fitness value of Harris Hawks was computed using RER and distance;
- To test and validate the proposed algorithm on diverse parameters like: calculating number of dead nodes, total residual energy, and throughput;
- And, to compare the proposed FHHO with other existing techniques such as GWO-C, PSO-ECHS and IFGWO.

**Organization of paper** The rest of the paper is organized as: Sect. 2 presents related work. Section 3 describes system preliminaries. Section 4 highlights proposed methodology. Section 5 enlightens results and discussion. Section 6 elaborates analysis. And, finally, Sect. 7 concludes the paper with future scope.

## 2 Related works

In recent years, many researchers had proposed various routing algorithms in IoT. Some IoT applications require cluster-based routing protocols to increase the lifespan of a network. The clustering protocol selects the appropriate CH from a set of nodes in the network using the CH selection method. The existing CH selection optimization algorithms in IoT are discussed in this section.

Agrawal et al. [39] suggested a new optimization algorithm called grey wolf optimizer-based clustering (GWO-C) for CH selection in WSNs. In addition, the intra-cluster distance, RER, and sink distance are used to compute the objective function. The simulations were conducted in several scenarios, and GWO-C was contrasted to FIGWO and PSO-ECHS. And, the results state that GWO-C is better in terms of throughput and network lifespan as compared to

other methods. However, it required more time to compute the objective function and for coverage during the CH rotation.

Mehta and Saxena [40] proposed a Multi-objective Cluster Head based Energy-aware Optimized Routing (MCH-EOR) algorithm in a WSN, mainly concentrating on traffic overload closest to the sink. MCH-EOR includes the following three processes for data transmission: clustering, selection of CH, and identification of the route. First, by using the division rule, the cluster was created. Second, the CH was selected according to probability. Third, the route identification was performed using the Sailfish Optimization Algorithm (SOA). Then, SOA efficacy was compared to GWO, GA, PSO, and ALO. The power consumption by the MCH-EOR algorithm and the number of live nodes were 21.9% and 24.4%, respectively, compared to GWO. The GWO approach extended the network lifespan effectively, but the CH selection was not optimal.

Karthick and Palanisamy [41] presented a novel Krill Herd (KH) algorithm for choosing the right CH in WSNs, focusing primarily on lifespan, network coverage, and data aggregation. The KH algorithm extended the network's lifetime as compared to the existing approaches, namely, LEACH and GA. Although the KH algorithm increased the lifetime of the network in WSN, the CH was usually a stationary node that gradually lost all its energy.

Poluru and Ramasamy [42] proposed a new clustering algorithm in IoT, which was divided into four stages: clustering, division and fusion, CH selection, and transmission of data. First, by using k-means algorithm the cluster was created. Second, the split and merge operation were performed in the cluster. Third, the Cyclic Rider Algorithm (C-ROA) determined CH in IoT networks. The data fusion tree was produced at the end of the procedure for data transmission. The effectiveness of the suggested algorithm was assessed in terms of the alive node, normalised energy, latency, and cost function measurements. The results showed that the C-ROA algorithm outperform the FF, WOA, ROA, and GWO algorithms by increasing alive nodes by 6–21% and reducing the delay by 6–8%. However, CH selection process needs to be enhanced by modifying the C-ROA approach.

Ahmad [43] Introduced optimization algorithm termed Artificial Bee Colony (ABC) for right CH selection in WSN. Specifically, first, the cluster was created using the K-means method, followed by the CH chosen by applying the ABC algorithm. The fitness function played a vital part in the optimization algorithm in the CH selection process. In the objective function calculation, the developed ABC approach considers the network parameters, namely Distance and RER. The developed ABC algorithm extended network lifetime by 10–20% as compared to E-LEACH, LEACH, and PSO-C. However, the ABC approach can perform well in a small network only.

Alazab et al. [36] developed a fitness-weighted averaged modified Rider Optimization Algorithm (ROA) for the selection of CH in IoT, aiming to identify the appropriate CH from the network nodes, reduce delay, and enhance energy efficiency. The objective function considered various network parameters, namely delay, distance, and energy, to choose the right CH in the adapted network. The modified ROA obtains two sets of solutions: the first set maintains the average value of follower and bypass riders, and the second set includes the average over-taker and attacker rider's value. The proposed FA-ROA scheme extended network lifetime by 4–22% compared to ABC, PSO, GSA, GA, ROA, and MFO. However, As the number of nodes in the network increased, the FA-ROA increased the load and delay, taking more time to compute the fitness value using the parameters load, temperature, data traffic, and energy.

Pathak [44] proposed a WSN protocol called the Proficient Bee Colony Clustering Protocol (PBC-CP) in order to improve the network's lifespan. Considering that clustering is a suitable option to save energy, the WSN's clustering process relies heavily on the selection of CH. The objective function considers various network parameters, namely the number of nodes, node energy, and distance of a node from the sink for the CH Selection process. The proposed PBC-CP performance measured with PSO, PSO-HAS, and LEACH, which showed to increase the network lifetime by 5–20%. However, the CH selection was not optimal due to its probability selection.

Zhao et al. [50] proposed an algorithm termed Improved Greywolf optimization for selecting optimal CH in WSN. The novel algorithm fitness-based Improved Grey Wolf optimization (FIGWO) was proposed. The fitness value was evaluated based on distance and energy to choose the right CH in the cluster. Each node transmission range was recalculated when the CH node is selected. Thus, both energy consumption and average transmission distance was reduced. The simulation of FIGWO was conducted using the MATLAB tool. The performance of FIGWO in terms of stability period was improved by 31.5 and 57.8% as compared to SEP and LEACH protocols. However, the CH selection was not optimal for massive networks.

In the reviewed works, some optimization strategies for selecting the best CH in IoT networks have been presented. However, these studies presented some limitations, which are described as follows. Nodes closer to the sink deplete energy faster due to inappropriate CH selection. Many optimization strategies require more time to compute the fitness function. Some optimization techniques are only suited for small-scale networks, while some algorithms choose the CH based on probability. This study selects CH in IoT networks by using the FHHO approach. The objective function is calculated over RER and Distance parameters to improve the network's performance.

**Table 1** CH selection approaches

S. no	Authors	Method/algorithm for CH selection	Advantages	Limitations
1	Agrawal et al. [39]	Grey Wolf Optimizer-based clustering (GWO-C)	Increased the throughput and lifespan of the network	It is required more time to compute the objective function
2	Mehta and Saxena [40]	Multi-objective Cluster Head based Energy-aware Optimized Routing (MCH-EOR)	Increased the network lifetime	CH selection process is not beneficial
3	Karthick and Palanisamy [41]	Krill Herd (KH) algorithm	Increased the network lifetime	CH was usually a stationary node that gradually lost all its energy
4	Poluru and Ramasamy [42]	Cyclic Rider Algorithm (C-ROA)	Improved data transmission lifetime and latency	Improvements to C-ROA are needed to enhance the CH selection process
5	Ahmad [43]	Artificial Bee Colony (ABC)	Increased the network lifetime by 10–20%	It is not optimal for large networks
6	Alazab et al. [36]	fitness-weighted averaged modified Rider Optimization Algorithm (ROA)	Increased the network lifetime by 4–22%	It requires more time for objective function computation by using the parameters, load, temperature, data traffic, and energy
7	Pathak [44]	Proficient Bee Colony Clustering Protocol (PBC-CP)	Network lifetime is improved by 5–20%	CH selection was not optimal due to its probability selection
8	Zhao et al. [50]	Fitness based Improved grey-wolf optimization (FIGWO)	Increased the network stability 15–20%	it is not optimal for large networks

Table 1 summarizes the approaches, enhancements, and limitations of the various optimization algorithms proposed for CH selection.

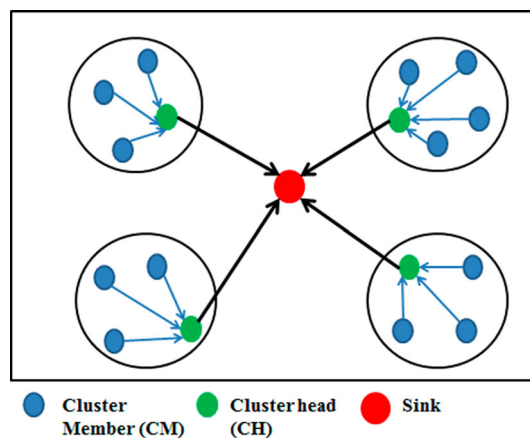
### 3 System preliminaries

#### 3.1 Network model

The network is composed of  $N$  nodes that are randomly dispersed throughout the network. The sink can be placed in three distinct locations: middle, corner, and outside. The proposed FHHO includes two operations: cluster creation and optimal CH selection [2]. The sink runs the CH selection algorithm, which has an extensive processing and memory capacity. The FHHO algorithm is used to choose the CH. The Euclidean distance creates clusters in the later phases. In Fig. 1, the network model is shown.

The following assumptions are made in the proposed FHHO algorithm:

- The energy of every node is the same.
- All nodes are fixed and are distributed at random.



**Fig. 1** Network model

- The distance between nodes in the network is calculated by using Euclidean distance.
- The sink can be placed in three distinct locations: middle, corner, and outside.

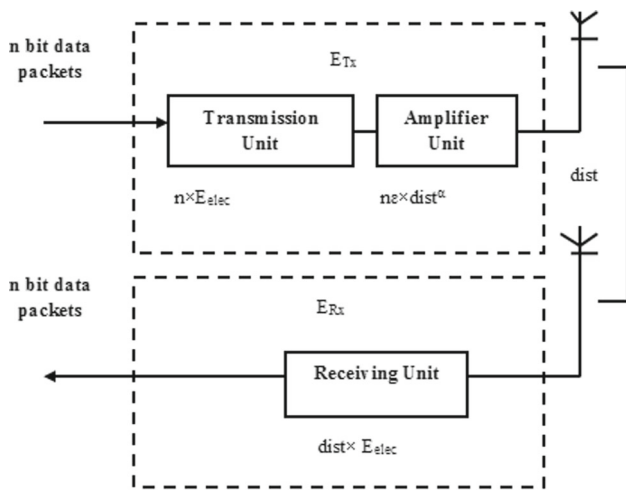


Fig. 2 Energy model

### 3.2 Energy model

Figure 2 depicts the FHHO energy model using the simplified channel model [45]. The FHHO algorithm can use either a free space or multi-fading channel model. The channel model is defined by distance between transmitter and the receiver. The energy used by node c as it delivers data packets to node d in relation to the distance  $dist(c,d)$  is determined using Eq. (1):

$$E_{TX}(n, dist) = n \times E_{elec} + n \times \varepsilon \times dist(c, d)^\alpha$$

$$= \begin{cases} n \times E_{elec} + n \times \varepsilon_{fr} dist(c, d)^2 & \text{wheredist}(c, d) < dist_0 \\ n \times E_{elec} + n \times \varepsilon_{mp} dist(c, d)^4 & \text{wheredist}(c, d) \geq dist_0 \end{cases} \quad (1)$$

The following  $dist_0$  threshold value is evaluated using Eq. (2):

$$dist_0 = \sqrt{\frac{\varepsilon_{fr}}{\varepsilon_{mp}}} \quad (2)$$

The amount of energy necessary to receive n bits from node x to node y is calculated with Eq. (3):

$$E_{RX}(dist) = dist \times E_{elec} \quad (3)$$

## 4 Proposed methodology: novel fuzzy-based Harris hawks optimizations (FHHO) algorithm for cluster head selection

The proposed novel fuzzy-based Harris Hawks Optimization (FHHO) algorithm selects the best CH in IoT. The CH

Table 2 Nomenclature of FHHO algorithm

Notation	Definition
RER	Residual Energy
HH	Harris Hawk
$S_{rand}(t)$	Randomly selected HH from population
$S(t)$	Current position of HH
$S_{rabbit}(t)$	Rabbit's position in the iteration t
LB	Lower bound
UB	Upper bound
$S_m(t)$	Average position of the HH's population in the current iteration
Ener	Escaping energy
$Ener_0$	Initial state of HH's energy
T	Maximum number of iterations
J	Random value
$\Delta S(t)$	The distance between position of the rabbit and present HH's position in t

selection process is mainly based on a fitness function. The fitness function is calculated by applying fuzzy logic over the network parameters such as RER and distance. Then, the sink is executed by the CH selection algorithm, which has higher computing environment than sensor nodes. A cluster is formed after selecting the CH based on the Euclidean distance. Table 2 presents the nomenclature of the FHHO algorithm.

### 4.1 CH selection process

As mentioned, the primarily goal of the proposed FHHO algorithm, population-based nature-inspired optimization algorithm [46], is to increase the network's lifespan. Fuzzy logic is used in the FHHO to determine the fitness function using RER and distance. The algorithm also includes Harris' Hawks (HH) cooperation and chasing style (HH), where HH cooperatively attacks the prey from different directions in the intelligent attack strategy. Generally, HH exhibits chasing behavior based on the situations, which is divided into three phases: (1) exploration, (2) shift from exploration to exploitation, and (3) exploitation.

#### 4.1.1 Exploration phase

Before an attack, the HH must wait, observe, and monitor to find the prey in the site. Comparatively, in a network, every node represents a candidate solution, where the best candidate solution acting as the prey in each iteration, t. For instance, the HH perches in various locations and use one of two strategies to locate a rabbit. In the first strategy, the position of the HH perch relies on the prey and other HH family



members. For this strategy, the random parameter  $q$  value lies between 0 and 0.5 ( $q < 0.5$ ). In the second strategy, the HH perches on random tall trees, where the random parameter  $q$  is above or equal to 0.5 ( $q \geq 0.5$ ). These strategies are computed using Eq. (4):

$$S(t+1) = \begin{cases} S_{rand}(t) - rand_1 \\ \times |S_{rand}(t) - 2 \times rand_2 \times S(t)| \text{ where } q \geq 0.5 \\ (S_{rabbit}(t) - S_m(t)) \\ -rand_3 \times (LB + rand_4 \times (UB - LB)) \text{ where } q < 0.5 \end{cases} \quad (4)$$

where  $S_{rabbit}(t)$  indicated position of rabbit's in the iteration  $t$ ;  $S(t+1)$  specifies the HH vector position in the next iteration  $t$ ;  $S(t)$  represents HH's current position;  $rand_2, rand_3, rand_4$  and  $q$  are random numbers; the upper and lower specify UB and LB of the respective HH's;  $S_{rand}(t)$  denotes the HH determined at random from the population; and  $S_m(t)$  indicates the average position of the HH's population in the current iteration.

The suggested FHHO algorithm produces the LB and UB at random within the home range locations of the group. The first strategy is generated using a random location and various other HH families. The rabbit's position in the current iteration, HH average position, LB, and UB generate the second strategy. The average position of the HH is calculated with Eq. (5):

$$S_m(t) = \frac{1}{N} \times \sum_{i=1}^N S_i(t) \quad (5)$$

where  $S_i(t)$  is each HH location and  $N$  indicates the total network nodes.

#### 4.1.2 Exploration translation from exploration to exploitation

The proposed FHHO algorithm may switch from the exploration phase to the exploitation phase based on the escape energy of Prey's. In general, the energy of the prey declines when attempting to escape. The prey's energy is computed using Eq. (6):

$$Ener = 2 \times Ener_0 \times \left(1 - \frac{t}{T}\right) \quad (6)$$

where  $Ener$ ,  $t$ ,  $Ener_0$ , and  $T$  indicate the prey's escape energy, current iteration, initial state of its energy, and maximum iterations, respectively.

At each iteration, the  $Ener_0$  value is adjusted at random from  $-1$  to  $1$ . Generally, the rabbit becomes physically

weaker when  $Ener_0$  falls from  $0$  to  $-1$  and becomes stronger when  $Ener_0$  increases from  $0$  to  $1$ . Generally, the escape energy ( $Ener$ ) value dynamically decreases during the iteration. If the escape energy  $|Ener| \geq 1$  is greater than one, the HH seeks a new part of the site for the prey. Therefore, it continues to perform the exploration phase. If the escape energy  $|Ener| < 1$ , the FHHO algorithm shifts from the exploitation to exploration phase.

#### 4.1.3 Exploitation phase

During this phase, the HH makes a surprise attack on a specific prey, which tries to escape. Here, four different chasing strategies soft besiege, hard besiege, Soft Besiege with Progressive Rapid Dive and Hard Besiege with Progressive Rapid Dive are plausible in the proposed FHHO algorithm. The prey is continually trying to flee the attack scenario, in which the variable  $r$  is a number between 0 and 1 that is chosen at random. The prey has a greater probability of escaping if the  $r$  value is between 0 and 0.5 ( $r < 0.5$ ) and a worse chance if the  $r$ -value is greater than or equal to 0.5 ( $r \geq 0.5$ ). The HH kills the prey if it tries to escape. In real situations, the HH attempts to get closer to the prey, and all HH will surprise attack the prey together. The prey will continue to lose escape energy after a few minutes. Finally, the HH will catch the weak prey by initiating besiege process. Depending on the prey's escape energy, either a hard or soft besiege will be performed. Specifically, a soft besiege occurs if the escape energy value is  $|Ener| \geq 0.5$ , while a hard besiege is performed if  $|Ener| < 0.5$ .

**4.1.3.1 Soft besiege** If  $r \geq 0.5$  and  $|Ener| \geq 0.5$ , the prey has more escape energy and will attempt to flee from the HH. The HH will then encircle the prey softly and enforce a surprise attack when the prey exhausts its energy. The soft besiege behavior is modeled and is provided in Eq. (7), (9):

$$S(t+1) = \Delta S(t) - Ener \times |J \times S_{rabbit}(t)| - S(t) \quad (7)$$

$$\Delta S(t) = S_{rabbit}(t) - S(t) \quad (8)$$

As observed in Eq. (9), the jump  $J$  value changes at random with each iteration to simulate the nature of the prey directions.

$$J = 2(1 - rand_5) \quad (9)$$

where  $\Delta S(t)$  measures the distance from the current iteration  $t$  of the HH location to the vector position of the rabbit or prey's.  $J$  indicates the jump on the escaping behavior; and  $rand_5$  is a random value from 0 and 1.

**4.1.3.2 Hard besiege** If  $r \geq 0.5$  and  $|Ener| < 0.5$ , the prey is tired and has little escape energy. After that, the HH circles the chosen victim in a hard behavior before performing the surprise attack. The current location of the HH in this circumstance is computed with Eq. (10):

$$S(t+1) = S_{rabbit}(t) - Ener \times |\Delta S(t)| \quad (10)$$

**4.1.3.3 Soft besiege with progressive rapid dive** If  $r < 0.5$  and  $|Ener| \geq 0.5$ , the prey has enough escape energy, but the HH will perform the soft besiege before the surprise attack. The prey's escape pattern follows the Lévy flight (LF) idea, whereby LF imitates the prey's actual zigzag misleading motion during the escape phase. The HH forms a team and makes a series of dives around the prey, constantly adjusting their position and direction in relation to the prey. The FHHO algorithm selects the best dive with respect to the prey. If the HH wishes to catch the prey, a soft besiege is performed, then the next location of the HH is computed with Eq. (11):

$$Y = S_{rabbit}(t) - Ener \times |J \times S_{rabbit}(t) - S(t)| \quad (11)$$

Later, the HH compares its previous dive with respect to the prey. If it was not correct, the HH performs erratic and quick dives to catch the prey. The HH dives according to the LF patterns, which is calculated in Eq. (12):

$$Z = Y + D_s \times LF(D) \quad (12)$$

where  $D_s$  is a size of dimension  $1 \times D$ , which is a random vector; LF indicates the Lévy flight function [18]; and D is the dimension of the network space.

The fitness function is calculated in below section. In the gentle besiege, the position of the HH is updated using Eq. (13):

$$S(t+1) = \begin{cases} Y & \text{if } Fitness(Y) < Fitness(S(t)) \\ Z & \text{if } Fitness(Z) < Fitness(S(t)) \end{cases} \quad (13)$$

**4.1.3.4 Hard besiege with progressive rapid dive** If  $r < 0.5$  and  $|Ener| < 0.5$ , the prey is weak and unable to escape from one place to another. As a consequence, the HH established a hard besiege before surprise attacking the prey. This strategy is similar to the soft besiege with respect to the prey, but the HH tries to come as close as possible to the prey on average. The fitness function is computed in the below subsection. The hard besiege with rapid dive is calculated using Eq. (15):

$$S(t+1) = \begin{cases} Y & \text{if } Fitness(Y) < Fitness(S(t)) \\ Z & \text{if } Fitness(Z) < Fitness(S(t)) \end{cases} \quad (14)$$

where Y and Z are computed using Eqs. (15) and (16), respectively:

$$Y = S_{rabbit}(t) - Ener \times |J \times S_{rabbit}(t) - S_m(t)| \quad (15)$$

$$Z = Y + D_s \times LF(D) \quad (16)$$

where  $S_m(t)$  is calculated from Eq. (5).

#### 4.1.4 Fitness function computation

A fitness function gives the optimal solution by evaluating the given input. In the FHHO algorithm, the fitness function is determined using fuzzy logic over network parameters, namely RER and distance, which are both maximization and minimization network parameters. For effective CH selection, we can apply fuzzy logic to compute the fitness function.

**4.1.4.1 Residual energy (RER)** RER represents the quantity of energy that network nodes contain and is determined by Eq. (17):

$$RER(N_i) = \frac{E_i - E_d}{E_i} \quad (17)$$

where  $N_i$  indicates the  $i$ th node;  $E_i$  is the node's initial energy; and  $E_d$  represents the node's drained energy.

**4.1.4.2 Distance** The distance between the sink and the  $i$ th node in the network is evaluated using Euclidean distance. Its calculation is given in Eq. (18):

$$Distance(sink, N_i) = \sqrt{\sum_{i=1}^n (Sink - N_i)^2} \quad (18)$$

where  $N_i$  indicates the  $i$ th node.

The Fuzzy Inference System (FIS) has two essential processes: fuzzification and defuzzification. In the FHHO algorithm, the fuzzification process gets the crisp input, i.e., RER and distance. The fuzzification process includes linguistic variables, which hold the value in the form of words or sentences, and a member function that is used to evaluate the linguistic variables [47]. The pictorial representation of FIS is given in Fig. 3.

The trapezoidal and triangle membership functions are used in the network parameters for computing the fitness function. The real-valued membership function of the position vector S is represented as a triangle curve. There are three parameters, namely  $i_1$ ,  $j_1$  and  $k_1$ . The parameters  $i_1$  and  $k_1$  are the base of the triangle, and  $j_1$  is the peak value of the triangle. The generic representation of the triangle membership

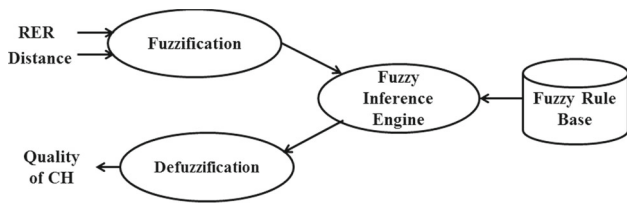


Fig. 3 Fuzzy inference system

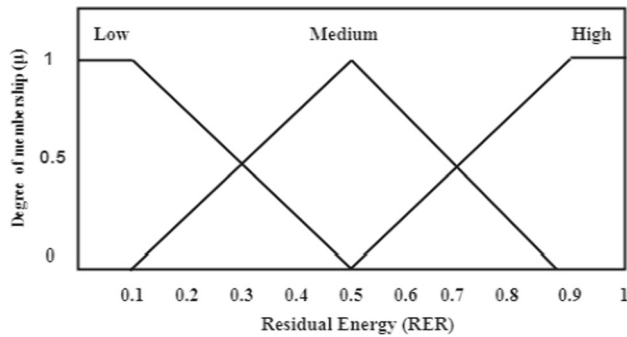


Fig. 4 Membership function of RER

function is provided in Eq. (19):

$$\mu_{R1}(S) = \begin{cases} 0 & S \leq i_1 \\ \frac{S-i_1}{j_1-i_1} & i_1 \leq S \leq j_1 \\ \frac{k_1-S}{k_1-j_1} & j_1 \leq S \leq k_1 \\ 0 & k_1 \leq S \end{cases} \quad (19)$$

If the random vector  $S$  has a trapezoidal distribution, its membership function has a true (i.e., measurable) value. The parameters  $i_2$  and  $j_2$  define the curve’s lower limit, while  $k_2$  and  $l_2$  define its upper limit. The trapezoidal membership is generically represented in Eq. (21):

$$\mu_{R2}(S) = \begin{cases} 0 & S \leq i_2 \\ \frac{S-i_2}{j_1-i_2} & j_2 \leq S \leq k_2 \\ 1 & j_2 \leq S \leq k_2 \\ \frac{k_2-S}{k_1-j_2} & k_2 \leq S \leq l_2 \\ 0 & l_2 \leq S \end{cases} \quad (20)$$

Figure 4 depicts the membership function of RER, which ranges from 0.1 to 1. Linguistic variables are Low, Medium, and High. Medium indicates the triangle membership function, while Low and High refer to trapezoidal membership.

Figure 5 illustrates the distance membership function, which ranges 0.1–1. The linguistic variables also include High, Low, and Medium that indicate the trapezoidal membership function.

Figure 6 depicts the quality of the CH membership function, which ranges 0–1. The function gives an output variable with five linguistic variables: Very good, Excellent, Good,

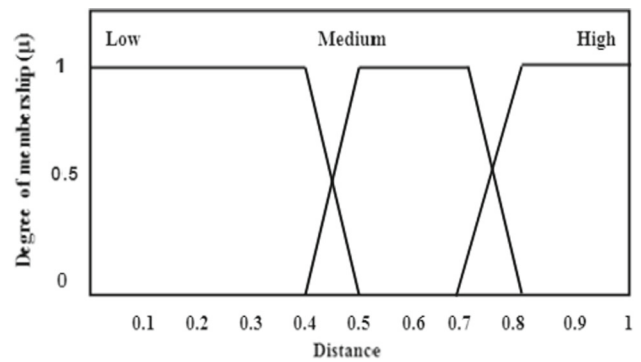


Fig. 5 Membership function of distance

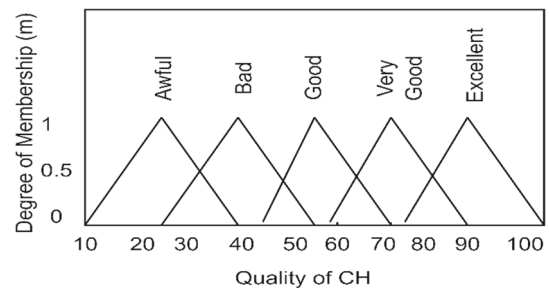


Fig. 6 Quality of CH membership function

Table 3 Fuzzy rules

RER	Distance	Quality of the CH
Low	Low	Good
High	High	Good
Low	High	Awful
Medium	High	Bad
Medium	Medium	Good
Medium	Low	Very good
High	Medium	Very good
High	Low	Excellent
Low	Medium	Bad

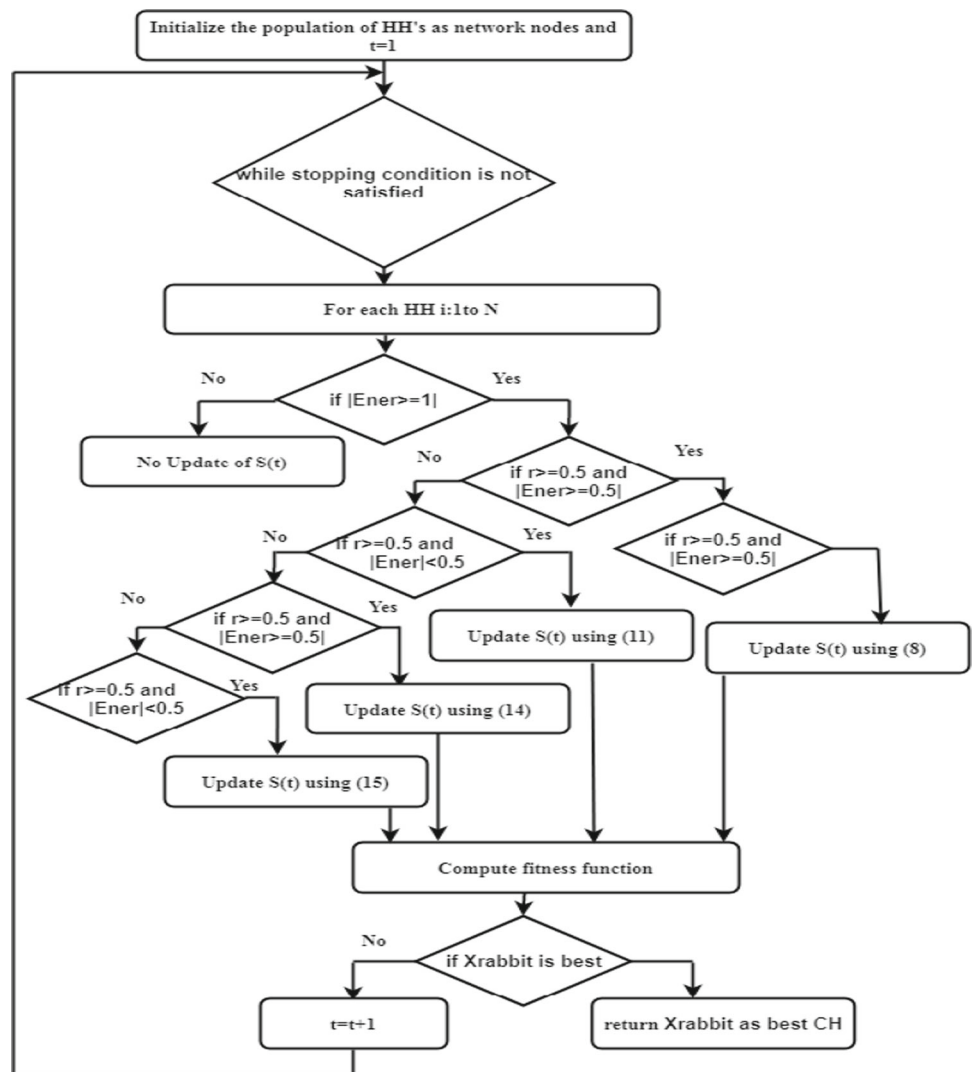
Bad, and Awful. The membership function is being utilized as a triangle.

The FHHO algorithm generates the nine rules, which are created based on the linguistic variables and membership functions. Table 3 shows the number of fuzzy rules, where the quality of CH values range between 0 and 100. The fuzzy number rules are entirely dependent on the application requirements. In the FHHO algorithm, the If–Then rule is used to check the condition, and the Mamdani model is applied to evaluate the rules [48].

Defuzzification is an important operation in FIS since it converts the representation into numerical values about the CH. For defuzzification in FHHO, the center of area approach



**Fig. 7** Workflow of FHHO algorithm



is utilized, which can be seen in Eq. (21).

$$COA(S) = \frac{\int \mu_R(S) S dS}{\int \mu_R(S) dS} \tag{21}$$

where  $\mu_R(S)$  denotes the output membership function; COA (S) is the defuzzified output variable; S is the output fuzzy variable; and R is a fuzzy region.

Figure 7 depicts the workflow of CH selection process using FHHO strategy.

Algorithm 1 provides the proposed FHHO algorithm for CH selection.

---

**Input:** Population N as total number of nodes, maximum number of iterations T  
**Output:** optimal CH in the network  
 1: network population  $S_i$  ( $i=1, 2, \dots, N$ ) and iteration  $t=1$   
 2: **While** (Stopping condition is not satisfied) **do**  
 3: **for** (each  $HH(S_i)$ ) **do**  
 4: Update  $Ener$ ,  $Ener_0$  and  $J$  using (7) and (10)  
 5: **if** ( $|Ener| \geq 1$ ) **then**  
 6: Update node location vector  $S(t)$  using (4)  
 7: **if** ( $|Ener| < 1$ ) **then**  
 8: **if** ( $r \geq 0.5$  and  $|Ener| \geq 0.5$ ) **then**  
 9: update node location vector  $S(t)$  using (8)  
 10: **else if** ( $r \geq 0.5$  and  $|Ener| < 0.5$ ) **then**  
 11: update node location vector  $S(t)$  using (11)  
 12: **else if** ( $r < 0.5$  and  $|Ener| \geq 0.5$ ) **then**  
 13: update node location vector  $S(t)$  using (14)  
 14: **else if** ( $r < 0.5$  and  $|Ener| < 0.5$ ) **then**  
 15: update node location vector  $S(t)$  using (15)  
 16: Calculate the fitness function of HH's  
 17: Set  $X_{rabbit}$  as best location  
 18: **return**  $X_{rabbit}$  as best CH

---

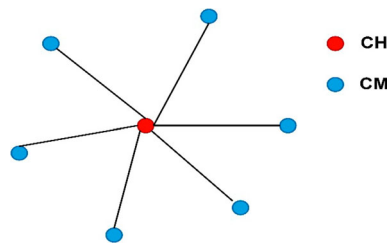


Fig. 8 Cluster formation

### 4.1.5 Cluster formation

The cluster is formed by grouping all nearby nodes to the CH. The distance between CH and its neighbor is computed using Euclidean distance [48], as given in Eq. (22):

$$Distance(CH, N_i) = \sqrt{\sum_{i=1}^n (CH - N_i)^2} \quad (22)$$

where  $N_i$  indicates the  $i$ th node. In Fig. 8, the cluster formation with CH is shown.

## 5 Results and discussions

The performance of the proposed FHHO approach was verified with similar algorithms, namely PSO-ECHS [49], FIGWO [50], and GWO-C [39]. The simulation was accomplished in MATLAB 2019a, which included a total of 100 nodes randomly distributed throughout a network area of  $100 \times 100$  m and 10 CHs. All nodes had the same beginning energy of 0.5 J. The simulation was conducted in three

Table 4 Simulation Specifications

Parameter	Value
Number of nodes	100
Network area	$100 \times 100$ m
Cluster heads	10
Tool	MATLAB2019a
Network area number of	$100 \times 100$ m
Initial energy	0.5 J
Sink	1
Sink locations	Center, corner and outside
Percentage of CH's	10%
Amplifier coefficient ( $q < q_0$ )	$10\text{pj/bit/m}^3$
Energy for data aggregation $E_{DA}$	$5\text{nj/bit/m}^4$
Amplifier coefficient ( $q \geq q_0$ )	$0.0013\text{pj/bit/m}^4$
Maximum network round	1500

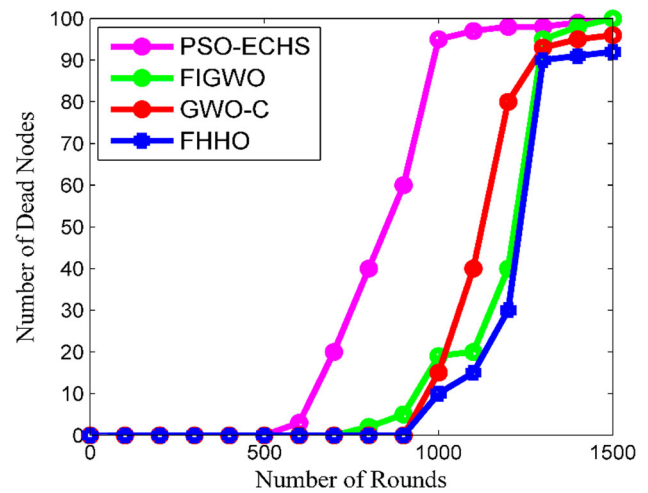


Fig. 9 Number of dead nodes with center location of sink node

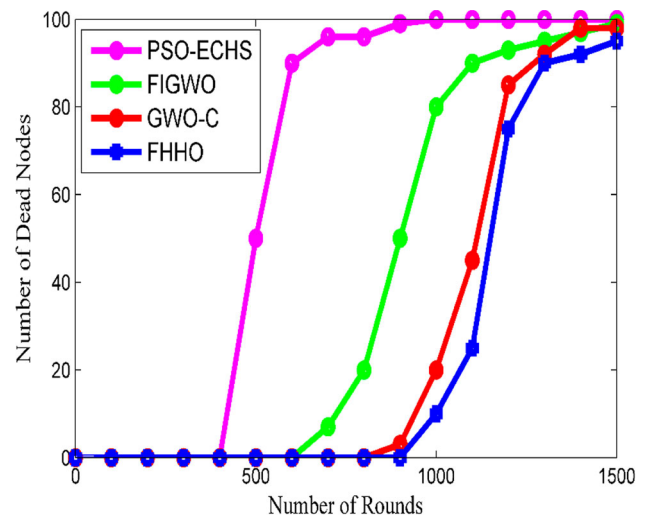


Fig. 10 Number of dead nodes with corner location of sink node

different scenarios based on the sink locations: in the center, corner, and outside. Each scenario was ran 10 times, with the average value being included in the performance evaluation metrics. The values of the simulation specifications are listed in Table 4.

### 5.1 Number of dead nodes

The network lifetime is an important performance metric that is calculated by the number of nodes that dies during a given round. The sink node in the scenario 1 was positioned at the center. Figure 9 depicts the connection between the number of dead nodes and the number of rounds. It is observed that the number of first dead nodes in PSO-ECHS, FIGWO, GWO-C, and FHHO was 523, 697, 756, and 920, respectively. Table 5 demonstrates the average results of scenario 1. The overall performance of the proposed FHHO is

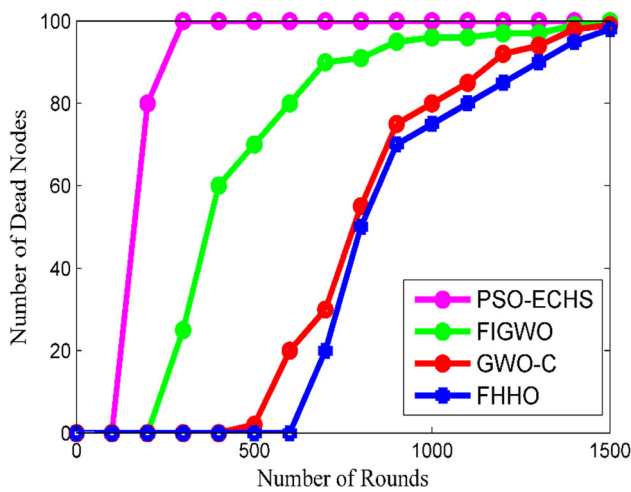


Fig. 11 Number of dead nodes with outside location of sink node

Table 5 Number of dead nodes vs number of rounds

Number of rounds	PSO-ECHS	FIGWO	GWO-C	FHHO
0	0	0	0	0
100	0	0	0	0
200	0	0	0	0
300	0	0	0	0
400	0	0	0	0
500	0	0	0	0
600	3	0	0	0
700	20	0	0	0
800	40	2	0	0
900	60	10	0	0
1000	95	19	15	10
1100	97	20	40	15
1200	98	40	80	30
1300	98	95	93	90
1400	99	98	95	91
1500	100	100	96	92

compared against similar algorithms regarding the number of dead nodes. The number of dead nodes increased while the number of iterations increased. It is also noted that FHHO enhanced the network lifespan by 44, 25, and 18% when compared to PSO-ECHS, FIGWO, and GWO-C. Fast convergence during the CH rotation and accurate fitness function calculation is mainly responsible for this result.

In scenario 2, the sink was located at the corner. Figure 10 depicts the number of nodes that died this location. It is observed that the 423, 620, 810 and 901 nodes died in PSO-ECHS, FIGWO, GWO-C, and FHHO, respectively. Table

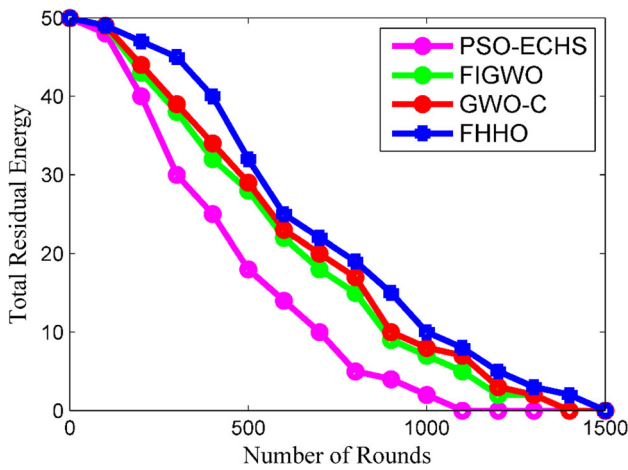
Table 6 Number of dead nodes vs number of rounds

Number of rounds	PSO-ECHS	FIGWO	GWO-C	FHHO
0	0	0	0	0
100	0	0	0	0
200	0	0	0	0
300	0	0	0	0
400	0	0	0	0
500	50	0	0	0
600	90	0	0	0
700	96	7	0	0
800	96	20	0	0
900	99	50	3	0
1000	100	80	20	10
1100	100	90	45	25
1200	100	93	85	75
1300	100	95	92	90
1400	100	97	98	92
1500	100	99	98	95

Table 7 Number of dead nodes vs number of rounds

Number of rounds	PSO-ECHS	FIGWO	GWO-C	FHHO
0	0	0	0	0
100	0	0	0	0
200	80	0	0	0
300	100	25	0	0
400	100	60	0	0
500	100	70	2	0
600	100	80	20	0
700	100	90	30	20
800	100	91	55	50
900	100	95	75	70
1000	100	96	80	75
1100	100	96	85	80
1200	100	97	92	85
1300	100	97	94	90
1400	100	99	98	95
1500	100	100	99	98

6 demonstrates the average results of scenario 2. The overall performance of the proposed FHHO is compared against similar algorithms regarding the number of dead nodes. The number of dead nodes increased while the number of iterations increased. It was also seen that FHHO enhanced the



**Fig. 12** Total residual energy with center location of sink node

network lifetime by 54, 22, and 11% when compared to PSO-ECHS, FIGWO, and GWO-C, respectively.

In scenario 3, the sink was located at outside the network. In Fig. 11, 423, 620, 810, and 901 nodes died in PSO-ECHS, FIGWO, GWO-C, and FHHO, respectively. Table 7 demonstrates the average results of scenario 3. The overall performance of the proposed FHHO is compared against similar algorithms regarding the number of dead nodes. The number of dead nodes increased while the number of iterations increased. In addition, FHHO enhanced the network lifetime by 71, 60, and 28% compared to PSO-ECHS, FIGWO, and GWO-C, respectively.

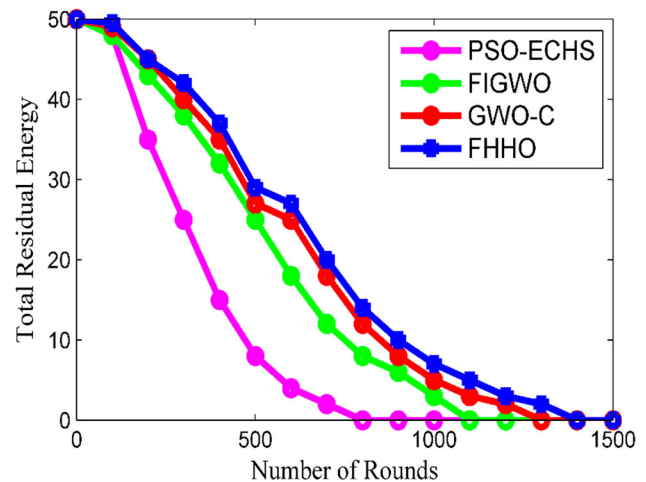
### 5.2 Total residual energy

As one of the prominent performance evaluation metrics, in IoT, total residual energy indicates the quantity of remaining energy in the nodes in a particular round. The sink node in scenario 1 was positioned at the center, the total residual energy in FHHO was high in all rounds compared to other PSO-ECHS, FIGWO, and GWO-C, as seen in Fig. 12. It is also noted that the total residual energy in PSO-ECHS, FIGWO, GWO-C, and FHHO was 2, 7, 8, and 10 J respectively, for 1000 network rounds. The proposed scheme demonstrated better performance in terms of residual energy compared against similar algorithms as shown in Table 8 (Scenario 1). As the number of iterations increased, the RER of the node decreased. The network parameters for the fitness computation, RER, and distance, are mainly responsible for the higher energy of FHHO. Therefore, it is evident that distance plays a significant impact on the sink.

Figure 13 depicts the total residual energy when sink is located at the corner (scenario 2). As seen, the total residual energy in FHHO was higher in all rounds when compared to PSO-ECHS, FIGWO, and GWO-C. The proposed scheme demonstrated better performance in terms of residual energy

**Table 8** Total residual energy (J) vs number of rounds

Number of rounds	PSO-ECHS	FIGWO	GWO-C	FHHO
0	50	50	50	50
100	48	49	49	49
200	40	43	44	47
300	30	38	39	45
400	25	32	34	40
500	18	28	29	32
600	14	22	23	25
700	10	18	20	22
800	5	15	17	19
900	4	9	10	15
1000	2	7	8	10
1100	0	5	7	8
1200	0	2	3	5
1300	0	2	2	3
1400	0	0	0	2
1500	0	0	0	0



**Fig. 13** Total residual energy with corner location of sink node

compared against similar algorithms as shown in Table 9 (scenario 2). As the number of iterations increased, the RER of the node decreased. The total residual energy in PSO-ECHS, FIGWO, GWO-C, and FHHP was determined to be 2, 3, 5, and 7 J, respectively, for 1000 network rounds.

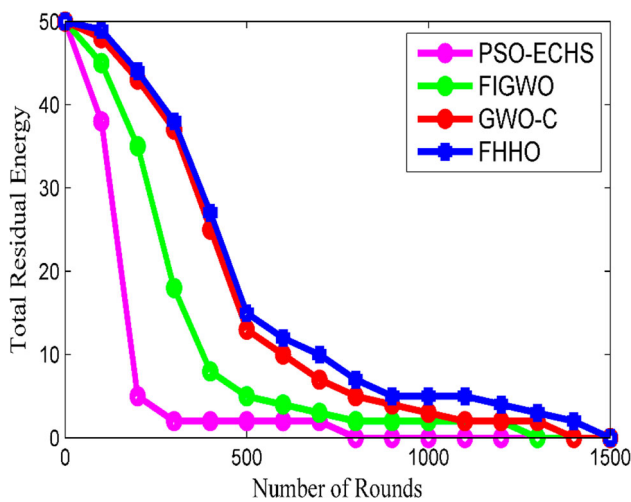
In the scenario 3, the sink node is located at outside the network. According to Fig. 14, the total residual energy in FHHO was higher in rounds than that in the other algorithms. The proposed scheme demonstrated better performance in terms of residual energy compared against similar algorithms as shown in Table 10 (scenario 3). As the number of iterations increased, the RER of the node decreased. the total residual

**Table 9** Total residual energy (J) vs number of rounds

Number of rounds	PSO-ECHS	FIGWO	GWO-C	FHHO
0	50	50	50	50
100	48	48	49	49.5
200	35	43	45	45
300	25	38	40	42
400	15	32	35	39
500	8	25	27	29
600	4	18	25	27
700	2	12	18	20
800	0	8	12	14
900	0	6	8	10
1000	0	3	5	7
1100	0	0	3	5
1200	0	0	2	3
1300	0	0	0	2
1400	0	0	0	0
1500	0	0	0	0

**Table 10** Total residual energy (J) vs number of rounds

Number of rounds	PSO-ECHS	FIGWO	GWO-C	FHHO
0	50	50	50	50
100	38	45	48	49
200	5	35	43	44
300	2	18	47	38
400	2	8	25	27
500	2	5	13	15
600	2	4	10	12
700	2	3	7	10
800	0	2	5	7
900	0	2	4	5
1000	0	2	3	5
1100	0	2	2	5
1200	0	2	2	4
1300	0	0	2	3
1400	0	0	0	2
1500	0	0	0	0

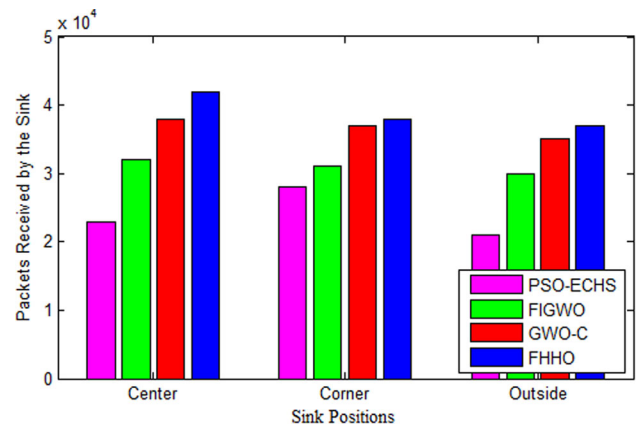


**Fig. 14** Total residual energy with outside location of sink node

energy in PSO-ECHS, FIGWO, GWO-C, and FHHO was 1, 2, 3, and 5 J, respectively, for 1000 network rounds.

### 5.3 Throughput

The throughput specifies how many data packets the sink has successfully received rely on RER and network lifetime. Figure 15 shows the packets received by the sink with respect to the sink’s position. In the PSO-ECHS, FIGWO, GWO-C, and FHHO algorithms, the sink received  $2.3 \times 10^4$ ,  $3.2 \times 10^4$ ,  $3.8 \times 10^4$  and  $4.2 \times 10^4$  packets when located at



**Fig. 15** Packets received by the Sink vs. Sink position

the center,  $2.8 \times 10^4$ ,  $3.1 \times 10^4$ ,  $3.7 \times 10^4$  and  $3.8 \times 10^4$  packets when at the corner, and  $2.1 \times 10^4$ ,  $3 \times 10^4$ ,  $3.5 \times 10^4$  and  $3.7 \times 10^4$  packets when located outside the network, respectively. It can be observed that when the sink received a higher number of packets when located in the center as shown in Table 11. In this location, all nodes in the cluster can easily communicate with the sink.

### 6 Analysis

The proposed FHHO approach is simulated and compared with existing systems showing better lifetime and throughput in the network. The simulation was run in three different



**Table 11** Number of packets received (J) vs Sink position

Sink position	PSO-ECHS	FIGWO	GWO-C	FHHO
Center	23,000	32,000	38,000	42,000
Corner	28,000	31,000	37,000	38,000
Outside	21,000	30,000	35,000	37,000

scenarios with the sink located in various positions across the network, including the center, corner, and outside. It is concluded that the sink placement plays a vital role in data collection as well as the entire network's performance. According to the scenarios discussed above, placing the sink in the center is more effective for improving network lifespan and throughput. Additionally, the proposed FHHO technique was executed at the sink and selected CH more effectively compared to the other optimization algorithms. The fitness functions are determined using fuzzy logic based on distance and RER network properties. Thus, the FHHO improved the network lifetime by 18–44% and throughput by 5–20%. However, the proposed FHHO algorithm was applied only to a static network environment and is suitable for only single and multi-objective problems.

## 7 Conclusion and future scope

Conserving energy in the context of IoT is a complex challenge due to the interconnection of IoT devices with resource-constrained devices. In addition, the battery source that powers IoT devices in most IoT applications cannot be charged; therefore, energy is a valuable resource. Thus far, clustering has proven to be the best method to save energy in IoT networks. However, The existing CH selection algorithms experience prolonged convergence due to the unstable balance between exploration and exploitation, two global search constraints. To solve these problems, this work proposes a novel FHHO algorithm for choosing optimal CH in network. The fitness functions are determined using fuzzy logic based on distance and RER network properties. Herein, the sink was positioned in three various locations, including the center, corner, and outside the network, to determine the optimal location for CH selection. The FHHO method enhanced network lifespan by 18–44% and throughput by 5–20% when the sink is positioned at the network's center. The performance of the FHHO algorithm is improved comprehensively in various scenarios using homogeneous sensor nodes. However, a few challenges are observed during the simulation, such as the FHHO approach working well in environments with homogeneous sensor nodes and its current architecture unable to manage scenarios with multiple node failures.

In the near future, it is planned to enhance the proposed FHHO algorithm by further increasing the network lifespan via implementation in real-time scenarios and also to combine FHHO with other algorithms to improve other aspects, such as throughput, network lifetime, latency, and packet delivery ratio. In addition, there are plans to implement the FHHO algorithm in a real-time environment.

**Author contributions** All the authors have equally contributed to the design and development of the manuscript.

**Funding** The authors received no specific funding for this study.

**Data availability statement** We declare that all the data associated with the manuscript is mentioned in the manuscript.

## Declarations

**Conflicts of interest** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

1. Senthil, G. A., Raaza, A., & Kumar, N. (2021). Internet of things multi hop energy efficient cluster-based routing using particle swarm optimization. *Wireless Networks*, 27, 5207–5215.
2. Manuel, A. J., Deverajan, G. G., Patan, R., & Gandomi, A. H. (2020). Optimization of routing-based clustering approaches in wireless sensor network: Review and open research issues. *Electronics*, 9(10), 1630.
3. Behera, T. M., Mohapatra, S. K., Samal, U. C., Khan, M. S., Daneshmand, M., & Gandomi, A. H. (2019). Residual energy-based cluster-head selection in WSNs for IoT application. *IEEE Internet of Things Journal*, 6(3), 5132–5139.
4. Sennan, S., Ramasubbareddy, S., Balasubramaniam, S., Nayyar, A., Abouhawwash, M., & Hikal, N. A. (2021). T2FL-PSO: Type-2 fuzzy logic-based particle swarm optimization algorithm used to maximize the lifetime of internet of things. *IEEE Access*, 9, 63966–63979.
5. Somula, R., Cho, Y., & Mohanta, B. K. (2024). SWARAM: osprey optimization algorithm-based energy-efficient cluster head selection for wireless sensor network-based internet of things. *Sensors*, 24(2), 521.
6. Chandrasekaran, S. K., & Rajasekaran, V. A. (2024). Energy-efficient cluster head using modified fuzzy logic with WOA and path selection using enhanced CSO in IoT-enabled smart agriculture systems. *The Journal of Supercomputing*, 80, 11149–11190.
7. Somula, R., Cho, Y., & Mohanta, B. K. (2023). EACH-COA: an energy-aware cluster head selection for the internet of things using the coati optimization algorithm. *Information*, 14(11), 601.
8. Gowda, S. S., & Ramalingappa, A. (2024). Energy optimized cluster head selection based on multi-objective sand cat swarm optimization in under water wireless sensor networks. *International Journal of Intelligent Engineering & Systems*, 17(1), 383.
9. Sankar, S., Ramasubbareddy, S., Dhanaraj, R. K., Balusamy, B., Gupta, P., Ibrahim, W., & Verma, R. (2023). Cluster head selection for the internet of things using a sandpiper optimization algorithm (SOA). *Journal of Sensors*, 2023(1), 3507600.
10. Janarthanan, A., & Srinivasan, V. (2024). Multi-objective cluster head-based energy aware routing using optimized auto-metric

- graph neural network for secured data aggregation in Wireless Sensor Network. *International Journal of Communication Systems*, 37(3), e5664.
11. Aramuthakannan, S., Kumar, R. R., Mariammal, G., & Geetha, M. (2024). Enhanced cluster head selection and routing in wireless sensor networks using fuzzy logic and adaptive cat swarm optimization. *International Journal of Intelligent Engineering & Systems*, 17(1), 721.
  12. Kirubasri, G., Sankar, S., Guru Prasad, M. S., Naga Chandrika, G., & Ramasubbareddy, S. (2023). LQETA-RP: link quality based energy and trust aware routing protocol for wireless multimedia sensor networks. *International Journal of System Assurance Engineering and Management*, 15(1), 564–576.
  13. Srivastava, A., & Mishra, P. K. (2024). Fuzzy based multi-criteria based cluster head selection for enhancing network lifetime and efficient energy consumption. *Concurrency and Computation: Practice and Experience*, 36(4), e7921.
  14. Sankar, S., Ramasubbareddy, S., Luhach, A. K., & alnumay, W. S., & Chatterjee, P. (2022). NCCLA: New caledonian crow learning algorithm based cluster head selection for Internet of Things in smart cities. *Journal of Ambient Intelligence and Humanized Computing*, 13(10), 4651–4661.
  15. Wu, D., Yang, Z., Li, T., & Liu, J. (2024). JOCP: A jointly optimized clustering protocol for industrial wireless sensor networks using double-layer selection evolutionary algorithm. *Concurrency and Computation: Practice and Experience*, 36(4), e7927.
  16. Sankar, S., Somula, R., Parvathala, B., Kolli, S., & Pulipati, S. (2022). SOA-EACR: Seagull optimization algorithm based energy aware cluster routing protocol for wireless sensor networks in the livestock industry. *Sustainable Computing: Informatics and Systems*, 33, 100645.
  17. Janakiraman, S. (2024). Energy efficient clustering protocol using hybrid bald eagle search optimization algorithm for improving network longevity in WSNs. *Multimedia Tools and Applications*, 1–23.
  18. Sennan, S., Ramasubbareddy, S., Balasubramaniam, S., Nayyar, A., Kerrache, C. A., & Bilal, M. (2021). MADCR: Mobility aware dynamic clustering-based routing protocol in Internet of Vehicles. *China Communications*, 18(7), 69–85.
  19. Sharma, S. K., & Chawla, M. (2024). PRESEP: Cluster based meta-heuristic algorithm for energy-efficient wireless sensor network application in internet of things. *Wireless Personal Communications*, 133(2), 1243–1263.
  20. Sennan, S., Ramasubbareddy, S., Nayyar, A., Nam, Y., & Abouhawwash, M. (2021). LOA-RPL: Novel energy-efficient routing protocol for the internet of things using lion optimization algorithm to maximize network lifetime. *Computers, Materials & Continua*, 69(1) 351–371.
  21. Afzal, H., Kanwal, S., Zulfiqar, M., Gill, H. B., & Mufti, M. R. (2023). Performance evaluation of various algorithms for cluster head selection in WSNs. *The Nucleus*, 60(1), 35–44.
  22. Sennan, S., Somula, R., Luhach, A. K., Deverajan, G. G., Alnumay, W., Jhanjhi, N. Z., Ghosh, U., & Sharma, P. (2021). Energy efficient optimal parent selection based routing protocol for Internet of Things using firefly optimization algorithm. *Transactions on Emerging Telecommunications Technologies*, 32(8), e4171.
  23. Srivastava, A., & Mishra, P. K. (2023). Load-balanced cluster head selection enhancing network lifetime in WSN using hybrid approach for IoT applications. *Journal of Sensors*, 2023(1), 4343404.
  24. Zheng, W. M., Xu, L. D., Pan, J. S., & Chai, Q. W. (2023). Cluster head selection strategy of WSN based on binary multi-objective adaptive fish migration optimization algorithm. *Applied Soft Computing*, 148, 110826.
  25. Sankar, S., Srinivasan, P., Ramasubbareddy, S., & Balamurugan, B. (2020). Energy-aware multipath routing protocol for internet of things using network coding techniques. *International Journal of Grid and Utility Computing*, 11(6), 838–846.
  26. Kumar, M., Mukherjee, P., Verma, K., Verma, S., & Rawat, D. B. (2021). Improved deep convolutional neural network based malicious node detection and energy-efficient data transmission in wireless sensor networks. *IEEE Transactions on Network Science and Engineering*, 9(5), 3272–3281.
  27. Ouyang, Y., Liu, A., Xiong, N., & Wang, T. (2020). An effective early message ahead join adaptive data aggregation scheme for sustainable IoT. *IEEE Transactions on Network Science and Engineering*, 8(1), 201–219.
  28. Sankar, S., Srinivasan, P., Luhach, A. K., Somula, R., & Chilamkurti, N. (2020). Energy-aware grid-based data aggregation scheme in routing protocol for agricultural internet of things. *Sustainable Computing: Informatics and Systems*, 28, 100422.
  29. Zhang, R., Zhang, S., Wang, T., & Xiong, N. (2021). A class of differential data processing-based data gathering schemes in internet of things. *IEEE Transactions on Network Science and Engineering*, 8(4), 3113–3128.
  30. Sennan, S., Ramasubbareddy, S., Luhach, A. K., Nayyar, A., & Qureshi, B. (2020). CT-RPL: Cluster tree based routing protocol to maximize the lifetime of internet of things. *Sensors*, 20(20), 5858.
  31. Wu, D., Sun, X., & Ansari, N. (2019). An FSO-based drone assisted mobile access network for emergency communications. *IEEE Transactions on Network Science and Engineering*, 7(3), 1597–1606.
  32. Usman, M., Jan, M. A., He, X., & Chen, J. (2018). A mobile multimedia data collection scheme for secured wireless multimedia sensor networks. *IEEE Transactions on Network Science and Engineering*, 7(1), 274–284.
  33. Ravi, G., & Kashwan, K. R. (2015). A new routing protocol for energy efficient mobile applications for ad hoc networks. *Computers & Electrical Engineering*, 48, 77–85.
  34. Shende, D. K., & Sonavane, S. S. (2020). CrowWhale-ETR: CrowWhale optimization algorithm for energy and trust aware multicast routing in WSN for IoT applications. *Wireless Networks*, 26, 4011–4029.
  35. Shyjith, M. B., Maheswaran, C. P., & Reshma, V. K. (2021). Optimized and dynamic selection of cluster head using energy efficient routing protocol in WSN. *Wireless Personal Communications*, 116(1), 577–599.
  36. Alazab, M., Lakshmana, K., Reddy, T., Pham, Q. V., & Maddikunta, P. K. R. (2021). Multi-objective cluster head selection using fitness averaged rider optimization algorithm for IoT networks in smart cities. *Sustainable Energy Technologies and Assessments*, 43, 100973.
  37. Sefati, S., Abdi, M., & Ghaffari, A. (2021). Cluster-based data transmission scheme in wireless sensor networks using black hole and ant colony algorithms. *International Journal of Communication Systems*, 34(9), e4768.
  38. Senthil, G. A., Raaza, A., & Kumar, N. (2022). Internet of things energy efficient cluster-based routing using hybrid particle swarm optimization for wireless sensor network. *Wireless Personal Communications*, 122(3), 2603–2619.
  39. Agrawal, D., Wasim Qureshi, M. H., Pincha, P., Srivastava, P., Agarwal, S., Tiwari, V., & Pandey, S. (2020). GWO-C: Grey wolf optimizer based clustering scheme for WSNs. *International Journal of Communication Systems*, 33(8), e4344.
  40. Mehta, D., & Saxena, S. (2020). MCH-EOR: Multi-objective cluster head based energy-aware optimized routing algorithm in wireless sensor networks. *Sustainable Computing: Informatics and Systems*, 28, 100406.

41. Karthick, P. T., & Palanisamy, C. (2019). Optimized cluster head selection using krill herd algorithm for wireless sensor network. *Automatika: Časopis za Automatiku, Mjerenje, Elektroniku, Računarstvo i Komunikacije*, 60(3), 340–348.
42. Poluru, R. K., & Ramasamy, L. K. (2020). Optimal cluster head selection using modified rider assisted clustering for IoT. *IET Communications*, 14(13), 2189–2201.
43. Ahmad, T. (2020). Energy EC: An artificial bee colony optimization based energy efficient cluster leader selection for wireless sensor networks. *Journal of Information and Optimization Sciences*, 41(2), 587–597.
44. Pathak, A. (2020). A proficient bee colony-clustering protocol to prolong lifetime of wireless sensor networks. *Journal of Computer Networks and Communications*, 2020(1), 1236187.
45. Sennan, S., Balasubramaniyam, S., Luhach, A. K., Ramasubbareddy, S., Chilamkurti, N., & Nam, Y. (2019). Energy and delay aware data aggregation in routing protocol for Internet of Things. *Sensors*, 19(24), 5486.
46. Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future generation computer systems*, 97, 849–872.
47. Lata, S., Mehruz, S., Urooj, S., & Alrowais, F. (2020). Fuzzy clustering algorithm for enhancing reliability and network lifetime of wireless sensor networks. *IEEE Access*, 8, 66013–66024.
48. Panchal, A., & Singh, R. K. (2021). EHCR-FCM: Energy efficient hierarchical clustering and routing using Fuzzy C-Means for wireless sensor networks. *Telecommunication Systems*, 76(2), 251–263.
49. Rao, P. S., Jana, P. K., & Banka, H. (2017). A particle swarm optimization based energy efficient cluster head selection algorithm for wireless sensor networks. *Wireless networks*, 23(7), 2005–2020.
50. Zhao, X., Zhu, H., Aleksic, S., & Gao, Q. (2018). Energy-efficient routing protocol for wireless sensor networks based on improved grey wolf optimizer. *KSII Transactions on Internet and Information Systems (TIIS)*, 12(6), 2644–2657.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.



**Dr Sankar Sennan** received the M.E. degree from Anna University and the Ph.D. degree from VIT University, Vellore, India, in 2019. He is currently working as an Assistant Professor with the Sona College of Technology, Salem. He has published various articles in international journals and conferences. His research interests include the IoT, WSNs, and machine learning.



**Dr Somula Ramasubbareddy** received the Ph.D. degree in computer science and engineering from VIT University, Vellore, India, in 2022. He was a Postdoctoral Research with the Department of Information and Communication, Suncheon National University, South Korea, in 2024. His academic and research achievements have earned him a place among the top 2% of scientists globally, a recognition bestowed upon him by Elsevier and Stanford University.

He is currently an Assistant Professor with the Department of Information Technology, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad. He has more than 40 publications in reputed journals and conferences. His research interests include mobile cloud computing, the IoT, machine learning, and edge computing.



**Dr Rajesh Kumar Dhanaraj** is a distinguished Professor at Symbiosis International (Deemed University) in Pune, India. Before joining Symbiosis International University, he served as a Professor at the School of Computing Science & Engineering at Galgotias University in Greater Noida, India. His academic and research achievements have earned him a place among the top 2% of scientists globally, a recognition bestowed upon him by Elsevier and Stanford University.

He earned his B.E. degree in Computer Science and Engineering from Anna University Chennai, India, in 2007. Subsequently, he obtained his M.Tech degree from Anna University Coimbatore, India, in 2010. His relentless pursuit of knowledge culminated in a Ph.D. in Computer Science from Anna University, Chennai, India, in 2017. He has authored and edited over 60 books on various cutting-edge technologies and holds 23 patents. Furthermore, he has contributed over 150 articles and papers to esteemed refereed journals and international conferences, in addition to providing chapters for several influential books.



**Dr. Anand Nayyar** received Ph.D (Computer Science) from Desh Bhagat University in 2017 in the area of Wireless Sensor Networks, Swarm Intelligence and Network Simulation. He is currently working in School of Computer Science-Duy Tan University, Da Nang, Vietnam as **Professor, Scientist, Vice-Chairman (Research) and Director- IoT and Intelligent Systems Lab**. A Certified Professional with **125+ Professional certifications** from CISCO,

Microsoft, Amazon, EC-Council, Oracle, Google, Beincert, EXIN, GAQM, Cyberoam and many more Published more than **200+**

**Research Papers** in various High-Quality ISI-SCI/SCIE/SSCI Impact Factor- Q1, Q2, Q3, Q4 Journals cum Scopus/ESCI indexed Journals, 80+ Papers in International Conferences indexed with Springer, IEEE and ACM Digital Library, 50+ Book Chapters in various SCOPUS/WEB OF SCIENCE Indexed Books with Springer, CRC Press, Wiley, IET, Elsevier with Citations: (**Google Scholar**): **14500+**, **H-Index: 64** and **I-Index: 228**; (**Scopus**): **7500+**; **H-index: 47**. Member of more than 60+ Associations as Senior and Life Member like: IEEE (Senior Member) and ACM (Senior Member). He has authored/co-authored cum Edited **60+ Books of Computer Science**. Associated with more than 600+ International Conferences as Programme Committee/Chair/Advisory Board/Review Board member. He has 18 Australian Patents, 15 German Patents, 4 Japanese Patents, 40 Indian Design cum Utility Patents, 13 UK Patents, 1 USA Patent, 3 Indian Copyrights and 2 Canadian Copyrights to his credit in the area of Wireless Communications, Artificial Intelligence, Cloud Computing, IoT, Healthcare, Drones, Robotics and Image Processing. Awarded **48 Awards for Teaching and Research**—Young Scientist, Best Scientist, Best Senior Scientist, **Asia Top 50 Academicians and Researchers**, Young Researcher Award, Outstanding Researcher Award, Excellence in Teaching, Best Senior Scientist Award, DTU **Best Professor and Researcher Award- 2019, 2020-2021, 2022, 2022-2023 Distinguished Scientist Award by National University of Singapore, Obada Prize 2023, Lifetime Achievement Award 2023; Asian Admirable Achievers 2024; Distinguished Academic Leader 2024** and many more. **He is listed in Top 2% Scientists as per Stanford University (2020, 2021, 2022), Ad Index (Rank No:1 Duy Tan University, Rank No:1 Computer Science in Viet Nam)** and Listed on <https://research.com/> (**Top Scientist of Computer Science in Viet Nam- National Ranking: 2; D-Index: 42; World Ranking: 6968**).



**Prof Balamurugan Balusamy** is currently working as Associate Dean Student in Shiv Nadar Institution of Eminence, Delhi-NCR. Stanford University has recently released an update of the list that represents the top 2 percent of the most-cited scientists in various disciplines. He is part of the Top 2% Scientists Worldwide 2023 by Stanford University in the area of Data Science/AI/ML. He is also an Adjunct Professor, Department of Computer Science & Information Engineering,

Taylor University, Malaysia. Prior to this assignment he was Professor, School of Computing Sciences & Engineering and Director International Relations at Galgotias University, Greater Noida, India. His contributions focus on Engineering Education, Block chain and Data Sciences. His Academic degrees and twelve years of experience working as a Faculty in a global University like VIT University, Vellore has made him more receptive and prominent in his domain. He does have 200 plus high impact factor papers in Springer, Elsevier and IEEE. He has done more than 200 Edited and authored books and collaborated with eminent professors across the world from top QS ranked university. Prof. Balamurugan Balusamy has served up to the position of Associate Professor in his stint of 15 years of experience with VIT University, Vellore. His passion is teaching and adapts different design thinking principles while delivering his lectures. He has published 80+ books on various technologies and visited 15 plus countries for his technical course. He has several top-notch conferences in his resume and has published over 200 of quality journal, conference and book chapters combined. He serves in the Expert Advisory Panel Member for several start-ups and forums and does consultancy work for industry on Industrial IOT. He has given over 210 talks in various events and symposium. He has travelling more than 15 countries for his academic and research assignments and talks.