information

Article

EACH-COA: An Energy-Aware Cluster Head Selection for the Internet of Things Using the Coati Optimization Algorithm

Ramasubbareddy Somula [1], Yongyun Cho [1,*] and Bhabendu Kumar Mohanta [2,*]

[1] Department of Information and Communication Engineering, Sunchon National University, Suncheon Si 57922, Republic of Korea; subbu1219@scnu.ac.kr
[2] Department of Computer Science & Engineering, Koneru Lakshmaiah Education Foundation, Green Fileds, Vaddeswaram, Andhra Pradesh 522302, India
* Correspondence: yycho@scnu.ac.kr (Y.C.); bkumarmohanta@kluniversity.in (B.K.M.)

Abstract: In recent years, the Internet of Things (IoT) has transformed human life by improving quality of life and revolutionizing all business sectors. The sensor nodes in IoT are interconnected to ensure data transfer to the sink node over the network. Owing to limited battery power, the energy in the nodes is conserved with the help of the clustering technique in IoT. Cluster head (CH) selection is essential for extending network lifetime and throughput in clustering. In recent years, many existing optimization algorithms have been adapted to select the optimal CH to improve energy usage in network nodes. Hence, improper CH selection approaches require more extended convergence and drain sensor batteries quickly. To solve this problem, this paper proposed a coati optimization algorithm (EACH-COA) to improve network longevity and throughput by evaluating the fitness function over the residual energy (RER) and distance constraints. The proposed EACH-COA simulation was conducted in MATLAB 2019a. The potency of the EACH-COA approach was compared with those of the energy-efficient rabbit optimization algorithm (EECHS-ARO), improved sparrow optimization technique (EECHS-ISSADE), and hybrid sea lion algorithm (PDU-SLno). The proposed EACH-COA improved the network lifetime by 8–15% and throughput by 5–10%.

Keywords: Internet of Things (IoT); cluster head; optimization technique; network lifetime; coati optimization algorithm

Citation: Somula, R.; Cho, Y.; Mohanta, B.K. EACH-COA: An Energy-Aware Cluster Head Selection for the Internet of Things Using the Coati Optimization Algorithm. *Information* **2023**, *14*, 601. https://doi.org/10.3390/info14110601

Academic Editor: Shingo Yamaguchi

Received: 28 September 2023
Revised: 31 October 2023
Accepted: 3 November 2023
Published: 5 November 2023

1. Introduction

The Internet of Things (IoT) is an emerging technology that plays an essential role in the efficiency and productivity of our everyday lives, among various other aspects. This technology has attracted the attention of both academics and industry members. The term IoT was coined in 1990 by Ashton [1–3]. IoT refers to a group of physical objects, namely, vehicles, machines, buildings, and other devices, that are connected to the Internet to transfer data to other participants in the network. A wide range of applications adopt IoT features, including transportation, smart healthcare, agriculture, and smart homes [4–6].

Sensor nodes in wireless sensor networks (WSNs) are geographically distributed to communicate and transmit data to each other. These sensors detect humidity, temperature, smoke, and other physical and environmental factors. Sensor nodes are designed to perform sensing, processing, and analysis of received data [7–9]. Sensor nodes have been enhanced with low power consumption owing to recent technological advancements of WSNs in, e.g., industrial automation and agriculture. In WSN applications, sensor nodes are distributed and managed using efficient computational algorithms. The network structure of sensor nodes changes owing to external factors. Consequently, sensor nodes face challenges, such as routing processes, data loss, limited processing power, and localization. The overall performance of a network can be enhanced by redesigning the network topology in a WSN [10–12].

WSN technology has been adapted to various emerging applications. Sensor nodes are low-power devices that operate using batteries. The energy consumption of sensors depends on multiple factors, including the type of routing protocol, distance between transmission nodes, and transmission packet size [13,14]. This affects the overall network performance. To avoid an energy-efficiency problem, an optimal routing protocol is required to maintain energy equally among every node in a network. The optimal routing protocol is adapted depending on the application requirements and network characteristics to improve the network lifetime [15,16].

Cluster head (CH) selection is a practical approach for forming a group of nodes as clusters in a WSN. A CH node consumes more energy than other cluster members (CMs) in a network. Various challenges, such as dynamic environment conditions, network scalability, and energy constraints, necessitate CH selection in the network. The designated CH in the cluster aims to collect and transmit data to a central node or base station in the network [17,18]. The CH consumes more energy than any other nodes in a cluster. Therefore, it is essential to select an optimal CH for a cluster by employing optimization techniques to ensure network stability and energy consumption [19,20]. The CH is chosen based on various network parameters, such as the remaining energy and distance between nodes, to improve the overall network performance. Several methods have been proposed for the CH selection [21–23].

An optimization algorithm is a mathematical computational technique used to determine the optimal solution for a given objective function. It is primarily used in the engineering field, particularly in the network field. Optimization algorithms execute many iterations to obtain appropriate solutions. Many existing optimization algorithms are available, including the cuckoo search (CS), honey bee mating optimization (HBMO), cat swarm optimization (CSO), whale optimization algorithm (WOA), harmony search (HAS), tabu search algorithm (TSA), and deferential evolution algorithm (DEA). However, most optimization algorithms require slightly longer convergence times during many rounds of iteration. To solve this problem, we propose a COA optimization algorithm to enhance network lifetime and throughput.

This study mainly focused on the following major research contributions:

- A substantial analysis was conducted on the proposed EACH-COA technique to obtain the best CH in a network. The effectiveness of the EACH-COA strategy was tested using various metrics: throughput, latency, and network longevity.
- EACH-COA performs CH selection and cluster formation. Using the coati optimization method, the best CH is selected, and clusters are grouped together using the nodes that are closest to each other.
- Existing optimization techniques are compared with the proposed EACH-COA methodology to demonstrate that EACH-COA outperforms them in network longevity.
- The fitness function is computed using residential energy (RER) and distance parameters in the CH selection process.
- The proposed EACH-COA technique was simulated using MATLAB 2019a. The overall network lifetime and throughput were improved by 8–15% and 5–10%, respectively.

The remainder of this article is organized as follows: Section 2 discusses existing works on CH selection-based optimization algorithms. Section 3 presents the system model, including the COA-network and COA-energy models. Section 4 describes the CH selection process using the coati optimization algorithm. Section 5 discusses the simulation, results from the discussion, and a comparison with existing optimization techniques. Section 6 presents conclusions and directions for future work.

2. Background

This section mainly focused on CH selection techniques for WSNs.

Rajkumar et al. [24] proposed a CH selection approach for a livestock industry WSN. The rabbit optimization algorithm (ROA) was used to select the designated CH in the network. The MATLAB 2021a platform was used for simulation in this study. The performance of EECHS-ARO was examined by comparing it with those of the QOBOA, ALO, and TLBO. EECHS-ARO improved the network lifespan and packet delivery ratio by 15% and 5%, respectively. However, EECHS-ARO requires convergence time while selecting the CH. The ROA requires considerably less convergence time than QOBOA, ALO, and TLBO.

Soni et al. [25] proposed a metaheuristic-optimized CH selection-based routing algorithm for WSNs. It uses the dragonfly optimization algorithm (DOA) to minimize energy consumption and achieve fast data transmission. MOCRAW has two phases: CH selection algorithm (CHSA) and routing selection algorithm (RSA). CHSA selects the optimal CH based on distance, RER, and node density metrics. MOCRAW was implemented using the NS2 simulation tool. The efficacy of MOCRAW was examined using E-FUCA, EAFTC-RIS, GAPSO-H, ECRP-UCA, and HMBCR. MOCRAW improved the overall energy efficiency by 10–15% compared with E-FUCA, EAFTC-RIS, GAPSO-H, and ECRP-UCA. However, convergence time is required to select the optimal CH.

Sankar et al. [26] proposed a sandpiper optimization algorithm (SOA) based on CH selection in IoT. SOA focuses on both cluster formation and CH selection using Euclidean distance. A MATLAB 2019a simulation environment was used for the SOA. The effectiveness of the SOA technique was improved compared to the EECHS-ARO, IABC, PSO, and ABC-Cd techniques. Compared with similar methods, the proposed SOA increased the network lifetime by 3–18% and throughput by 6–10%. Consequently, the SOA decreases the overall energy consumption while selecting the CH. Therefore, it transmits more packets with less delay during CH selection.

Anil et al. [27] proposed a hybrid swarm optimization algorithm for CH selection in a WSN. To select the CH, the harmony optimization algorithm (HOA) and competitive swarm optimization (CSO) approaches were combined. A network simulator (NS2) was used to simulate HAS-CSO. The performance of HAS-CSO was computed based on various metrics: RER, count of dead nodes, and throughput. As a result, the HAS-CSO approach surpassed the overall network lifetime of ABC-SD, FAMACROW, and Bee Swarm by 5–10%. However, the fitness function requires more computation time for CH selection.

Rakesh et al. [28] proposed a hybrid optimization-based energy-aware CH selection method for WSNs to improve overall network performance in terms of energy and network lifetime. A MATLAB simulation environment was used to implement the proposed PDU-SLnO energy-efficient protocol. The PDU-SLnO approach was developed using PSO and sea lion optimization (SLno) techniques. The efficacy of PDU-SLnO was analyzed with respect to various metrics, such as delay, energy, quality of service (QoS), and distance. The proposed PDU-SLnO enhanced the overall normalized energy by 32.90% compared to GA, PSO, ALO, GAL-LF, GOA, FGF, and CS-PS. However, this protocol can only be used in specific applications.

Panimalar et al. [29] proposed an energy-efficient CH selection protocol for WSNs. This paper presented an improved sparrow search algorithm for choosing the best CH possible in a cluster. The proposed EECHS-ISSADE approach was computed using various parameters, such as the number of active nodes, inactive nodes, throughput, and unused energy. The EECHS-ISSADE approach was simulated using MATLAB R2018a. The proposed EECHS-ISSADE improved the network longevity by 6–8%. However, it required a long convergence time during CH selection. The energy efficiency of EECHS-ISSADE was superior to that of EECHS-ABC, TABU-PSO, and LEACH.

Sengathir et al. [30] proposed a hybrid model to select the optimal CH in a WSN to ensure network consistency. The proposed HMABCFA approach combines a modified artificial bee colony (ABC) and firefly algorithm (FA) to find the optimal CH. The FA updating position feature was merged with the ABC algorithm to avoid low convergence speed during the CH selection process. HMABCFA was implemented using Python 3.6 with supporting libraries. The performance of the proposed HMABCFA technique was evaluated in terms of energy stability, network lifetime, and latency. As a result, the proposed HMABCFA improved the lifespan by 23.21%, reduced latency by 22.88%, and improved energy stability by 19.84% compared with MBABCOA, KHOGACP, and GSAC. However, a longer time was required for convergence owing to the hybrid approach.

Various benchmark studies on CH selection in WSN have been conducted by adapting different optimization techniques. However, the limitations of the related works have been observed, which include a longer network time, increased energy consumption, and poor network stability. The limitations of various CH selection optimization algorithms are listed in Table 1.

Table 1. CH selection using optimization algorithms.

S.No	Authors	Proposed CH Selection Optimization Techniques	Advantages	Limitations
1	Ramalingam et al. [24]	ARO	Network lifetime and packet delivery ratio are improved by 15% and 5%	It takes more time to form clusters in the network
2	Chaurasia et al. [25]	DA	Minimized energy consumption by 0.0014 J	It consumes more energy during CH selection
3	Sankar et al. [26]	SOA	Improved throughput and network lifetime by 6–10% and 3–18%	Sensor nodes deplete energy early during CH selection process
4	Kumar et al. [27]	HSA-CSO	Prolongs network lifetime and minimizes energy consumption	Convergence takes time for CH selection process in network
5	Yadav et al. [28]	PDU-SLnO	Increased network lifetime and consumes less energy	Sensor nodes deplete energy early during CH selection process
6	Kathiroli and Selvadurai [29]	SSA	Extended lifetime of sensor nodes	Takes more time to converge
7	Sengathir et al. [30]	EABC-FA	Prolonged network lifespan by 23.21% and energy stability by 19.84% and reduced network delay 22.88%	It takes more time to select CH selection

To address these concerns, this paper proposes a novel approach for selecting energy-efficient cluster heads in a WSN using the COA. The proposed EACH-COA method aims to enhance the overall lifespan of the network.

3. System Model

3.1. Network Model

The network is formed by randomly deploying n nodes across the network area. Each node has a unique ID and an array of properties, including initial energy, processing capacity, and communication energy. All network node positions are fixed and cannot be moved from one area to another, thereby reducing the intervention between nodes. The sink node serves as the central point of the network. The CH node establishes communication between all sensor nodes and collects data from each sensor node for transmission to a higher level. The sink node selects the CH node by executing the CH selection algorithm. The sink node adopts the COA algorithm to select the optimal CH node. Eventually, the CH creates the cluster using Euclidean distance. The CH obtains information from the cluster nodes and transmits it to the sink node. The network model of COA is illustrated in Figure 1.

3.2. Energy Model

The COA protocol transmits data using a standard channel model [31]. The energy consumption of the node was measured based on the amount of energy required to transmit a data packet from the participant to the sink node. Figure 2 depicts the COA energy consumption model.

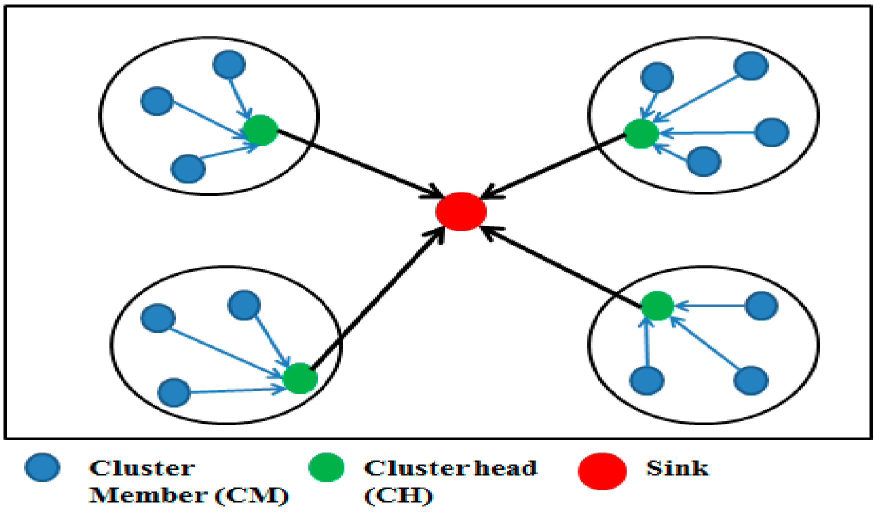

Figure 1. COA network model.

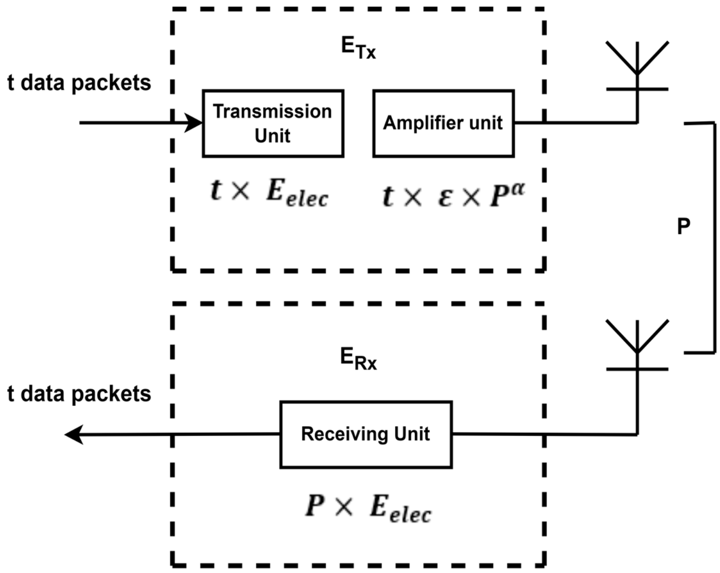

Figure 2. COA energy model.

In COA, distance 'p' between the nodes 's' and 'r' is represented within the coverage area. It employs either a multipath channel fading model or the free space channel. The evaluation of the energy consumption for transmitting t bits of data from node 's' to node 'r' is conducted as follows:

$$\begin{aligned} E_{TX}(t,p) &= tE_{elec} + m\varepsilon \mathrm{p}(\mathrm{s},\mathrm{r})^{\alpha} \\ &= \begin{cases} t\times E_{elec} + t\varepsilon_{\mathrm{ft}}\mathrm{p}(\mathrm{s},\mathrm{r})^2 \ \text{ where } \mathrm{p}(\mathrm{s},\mathrm{r}) < \mathrm{p}_0 \\ tE_{elec} + t\varepsilon_{\mathrm{mp}}\mathrm{p}(\mathrm{s},\mathrm{r})^4 \text{ where } \mathrm{p}(\mathrm{s},\mathrm{r}) \geq \ \mathrm{p}_0 \end{cases} \end{aligned} \tag{1}$$

where $t\varepsilon_{\mathrm{fr}}\mathrm{y}(\mathrm{s},\mathrm{r})^2$ or $t\varepsilon_{\mathrm{mp}}\mathrm{y}(\mathrm{s},\mathrm{r})^4$ represents amplifier unit energy consumption.

The distance threshold value p_0 is computed using Equation (2).

$$\mathrm{p}_0 = \sqrt{\frac{\varepsilon_{ft}}{\varepsilon_{mp}}} \tag{2}$$

The energy expenditure by node 'r' in the reception of 'q' bits of data from node 's' is expressed by Equation (3).

$$E_{RX}(P) = PE_{elec} \tag{3}$$

4. Proposed COA Protocol

The COA-based energy-aware CH selection protocol proposed in this paper significantly enhances network lifetime and improves throughput in WSNs. The COA is responsible for two processes: CH selection and cluster formation. The CH is selected by executing the COA algorithm, and a cluster is then constructed based on the neighboring nodes of the CH. Thus, the overall lifetime of the network increases.

4.1. Coati Optimization Algorithm-Based Cluster Head Selection

The COA is a population-based metaheuristic optimization algorithm [32]. Coatis are considered sensor nodes in a WSN. The sensor nodes are randomly distributed at the beginning of the initialization according to Equation (4). The values of the decision variables are defined by the positions of the coatis in the search space. Consequently, the positions of the coatis represent the proposed solution to the problem. It comprises two phases: exploration and exploitation.

$$C_i : \ c_{i,j} = lob_j + ran \cdot \left(upb_j - lob_j\right) \ i = 1, 2, \ldots, n, \ j = 1, 2, \ldots, m, \tag{4}$$

which represents the position of the *i*-th coati in the search space and indicates the value of the decision variable, and n represents the number of coatis in the search space, m indicates the number of decision variables, ran indicates a random real value in the range of [0, 1], and upb_j and lob_j are the lower and upper bounds of the *j*-th decision variable, respectively.

The mathematical representation of the coati population matrix C is represented as

$$C = \begin{bmatrix} C_1 \\ \vdots \\ C_i \\ \vdots \\ C_n \end{bmatrix}_{n \times m} = \begin{bmatrix} C_{1,1} & \cdots & C_{1,j} & \cdots & C_{1,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ C_{i,1} & \ldots & C_{i,j} & \ldots & C_{i,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ C_{n,1} & \ldots & C_{n,j} & \ldots & C_{n,m} \end{bmatrix}_{n \times m} \tag{5}$$

The objective function obtains various values for the candidate solution from the decision variables. These values are expressed by Equation (6).

$$V = \begin{bmatrix} V_1 \\ \vdots \\ V_i \\ \vdots \\ V_n \end{bmatrix}_{n \times 1} = \begin{bmatrix} V(C_1) \\ \vdots \\ V(C_i) \\ \vdots \\ V(C_n) \end{bmatrix}_{n \times 1} \tag{6}$$

where V indicates the vector of objective functions and V_i represents the value of the i-th coati objective value. In COA, the candidate solution for the objective function is defined by the position of the members in the population. During the iterations of the algorithm, the best member position and candidate solution may change.

4.1.1. Exploration Phase

The group of coatis hunt prey by scaring and climbing trees. Some coatis wait behind trees for their prey to fall to the ground. When prey fall from a tree, the coatis chase and attack it. The hunting process leads to different positions of coatis in the search space and defines their ability to explore the problem-solving space. The position of the prey is assumed to be the best member of the population. Half of the coatis climb the tree and the other half wait under the tree for the prey to fall to the ground. The position of the coatis expressed mathematically by Equation (7).

$$C_i^{\boldsymbol{pos1}} : x_{\boldsymbol{i,j}}^{\boldsymbol{pos1}} = x_{i,j} + ran \cdot \left(prey_j - I \cdot x_{i,j}\right) \text{ , } for\ i = 1, 2, \ldots, \frac{n}{2}\ and\ j = 1, 2, \ldots, m. \tag{7}$$

The prey is placed randomly in the search space when it falls from the tree. Based on the random position of the prey, the coatis' positions move in the search space, which is evaluated using Equations (8) and (9):

$$prey^{\boldsymbol{G}} : prey_J^G = lob_j + ran \cdot \left(upb_j - lob_j\right)\ i = 1, 2, \ldots, m, \tag{8}$$

$$C_i^{pos1} : c_{i,j}^{pos1} = \begin{cases} c_{i,j} + ran \cdot \left(prey_j^G - I \cdot c_{i,j}\right), \ V_{prey^G} < V_i, \\ c_{i,j} + ran \cdot \left(x_{i,j} - prey_j^G\right), \ else, \end{cases} \tag{9}$$

$$for\ i = \left[\frac{n}{2}\right] + 1,\ \left[\frac{n}{2}\right] + 2, \cdots, n\ and\ j = 1, 2, \cdots, m.$$

The updated positions of the coatis within the search space are contingent on the improvement in the value of the objective function. In contrast, if the value fails to improve, the coatis' positions will remain fixed at their previous values. The new coati position is determined by Equation (10).

$$C_i = \begin{cases} C_i^{pos1}, V_i^{pos1} < V_i, \\ C_i,\ else. \end{cases} \tag{10}$$

where $\boldsymbol{C_i^{pos1}}$ indicates the new position of the i-th coati; $\boldsymbol{c_{i,j}^{pos1}}$ is the coati's position in the j-th dimension; $\boldsymbol{V_i^{pos1}}$ indicates the value of the objective function; ran represents a random value in the range of 0 to 1; prey indicates the position of prey in the search space, which is considered as the best member; $\boldsymbol{prey_j}$ represents the jth dimension; I indicates a random integer value selected from the set [1,2]; $\boldsymbol{prey^G}$ indicates the position of prey on the ground; $\boldsymbol{prey_J^G}$ represents the jth dimension; and $\boldsymbol{V_{prey^G}}$ gives the value of the objective function.

4.1.2. Exploitation Phase

Coati position is updated in the search space when the coati encounters a predator or a predator attacks the coati, which is mathematically modeled based on the natural behavior of the coati. The coati moves to a safe position when the predator attacks it. Random positions of the coatis are generated in a local search to keep the coatis near the current positions, which is calculated using Equations (11) and (12):

$$\boldsymbol{lob_j^{loc} = \frac{lob_j}{z}, upb_j^{loc} = \frac{upb_j}{z}, where\ z = 1, 2, \ldots, Z.} \tag{11}$$

$$\boldsymbol{C_i^{pos2} : c_{i,j}^{pos2} = c_{i,j} + (1 - 2ran) \cdot \left(lob_j^{loc} + ran \cdot \left(upb_j^{loc} - lob_j^{loc}\right)\right)} \tag{12}$$

$$i = 1, 2, \ldots, n,\ j = 1, 2, \ldots, m$$

The position of coatis in the search space is updated if the value of the object function is improved, which is calculated using Equation (13).

$$C_i = \begin{cases} C_i^{pos2},\ V_i^{pos2} < V_i, \\ C_i,\ else, \end{cases} \tag{13}$$

where C_i^{pos2} represents the updated i-th new position in the exploitation phase of the COA; $c_{i,j}^{pos2}$ indicates the j-th direction; V_i^{pos2} indicates the objective function value; and ran represents a random value between 0 and 1.

4.1.3. Fitness Function

The fitness function is used to optimize the candidate solution for an objective problem. It considers both the Residual Energy (RER) and distance between nodes (Distance) to find an optimal solution. In COA, the fitness function determines whether the coati reaches the prey location. The coati evaluates the fitness value in each iteration to find the prey. If a coati reaches the best position, that node is considered the CH in that iteration. The CH selection process is presented in Algorithm 1.

Algorithm 1: COA-based CH Selection Algorithm

```
Input: Number of nodes 'n'
Output: Best position of coati acts as CH
1: initialize the position of nodes using Equation (4)
2: For z = 1 to Z do
3: prey position is updated based on best member position
//exploration phase
4:          For z = 1 to [Z/2]
5:               the new position of coati is calculated using Equation (7)
6:               update position of i-th coati using Equation (10)
7:          END for
8:          For z = 1 + [Z/2]: Z
9:               prey random position is computed using Equation (8)
10:         coati new position is computed using Equation (9)
11: updated position of i-th coati using Equation (10)
12: END for
//exploitation phase
13:      For z = 1 to Z
14:           Update the position of the ith coati using Equations (11) to (13)
15:      END for
16: compute the fitness value using Equation (16)
17: If coati reaches best position, then
18:      Best coati acts as CH
19: else
20:      Go to step 1
21: END for
22: return optimal CH
```

- Residual Energy (RER)

The sensor nodes are equipped with limited battery life, affecting the network's longevity. The node with the highest RER is considered the CH in the network for long-term network sustainability before replacing the node with others. The RER metric plays an essential role in a WSN, which can be used to determine the difference between the initial and available energy after some operation time [33]:

$$RER(n) = \frac{Energy_{avail}}{Energy_{intial}} \tag{14}$$

- Computation of Distance

A shorter distance between CH and CM may reduce communication overhead and energy consumption for transmitting data over the network. The distance parameter plays a significant role in real-time scenarios where nodes are equipped with limited battery power. The distance between a sensor node n_i and a sink node is computed by utilizing the Euclidean distance, as stated in reference [34]:

$$dis(n_i, sink) = \sqrt{\sum_{i=1}^{n} (sink - n_i)^2} \tag{15}$$

The present position of the coati's fitness function $C_{i(fitness)}$ is determined through Equation (16).

$$C_{i(fitness)} = 0.5 \times (1 - RER(C_i) + 0.5 \times (1 - dis(C_i)) \tag{16}$$

The CH selection workflow of EACH-COA is represented in Figure 3.

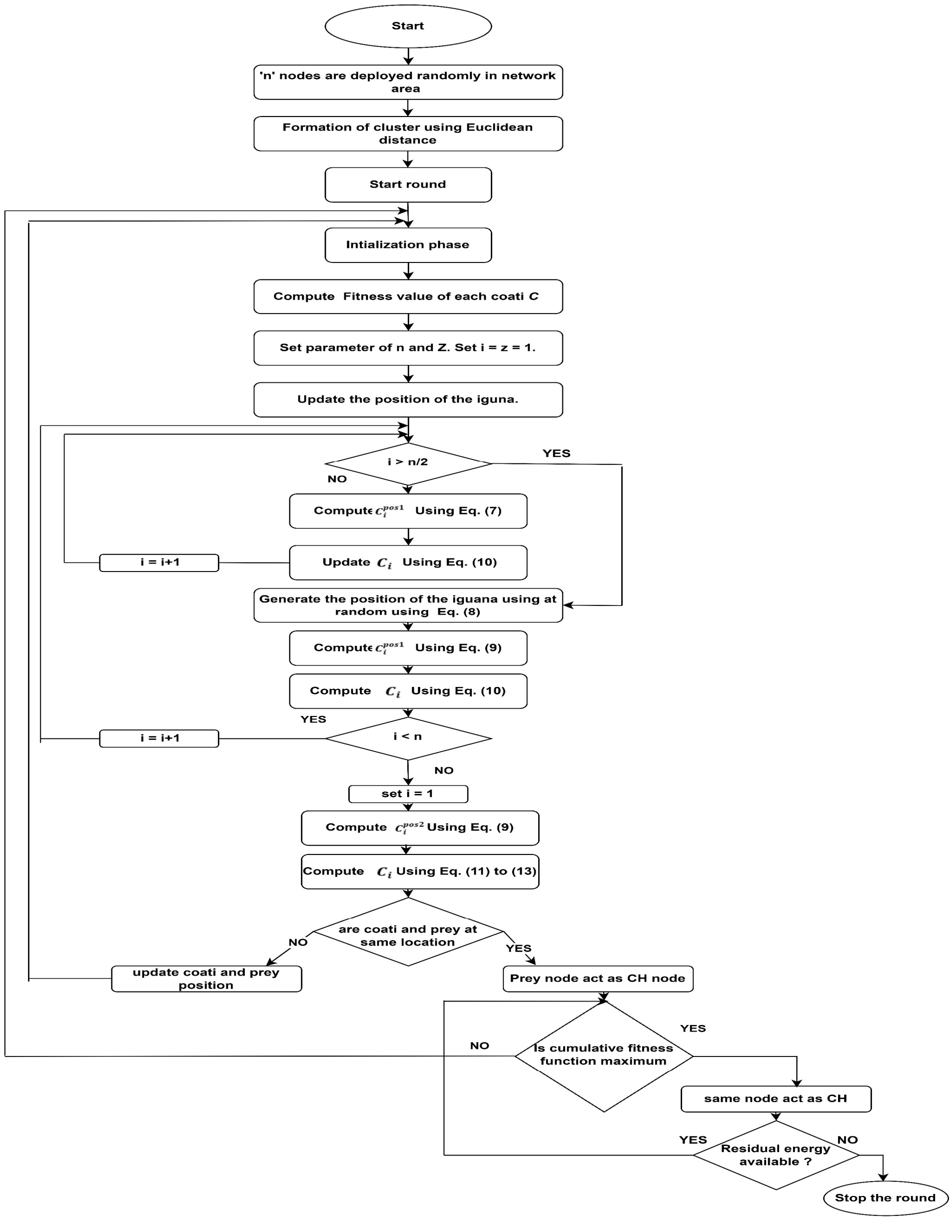

Figure 3. EACH-COA workflow.

5. Results and Discussion

The proposed EACH-COA technique was evaluated using the MATLAB 2019a simulation framework [35]. Numbers of nodes of 100, 200, and 300 were considered to evaluate the effectiveness of EACH-COA compared with EECHS-ARO, EECHS-ISSADE, and PDU-SLno. The distribution of nodes within the network was randomized. All nodes were initialized with equal energy in the network. A network area of 500 m $\times$ 500 m was considered for the simulation. The sink node was positioned at 250 m $\times$ 250 m, i.e., the center of the network. To gauge the effectiveness of EACH-COA, parameters such as the energy consumption of nodes, network throughput, and lifespan were compared with the benchmark studies available in the literature. The simulation parameters and values are listed in Table 2.

Table 2. Simulation environment.

Parameter	Value
Network area	500×500 m^2
Sink location	(250, 250)
Number of sensor nodes	100, 200, 300
CH percentage	5–10%
Control packet size	200 bits
Data packet size	4000 bits
Free space energy $\in_{fs}$	10 PJ/bit/m^2
Multipath energy $\in_{mp}$	0.0013 PJ/bit/m^4

5.1. Network Longevity

Figure 4 illustrates the network longevity by running different sets of nodes. The simulation was conducted using three sets of nodes: 100, 200, and 300. In the 100-node scenario, the performances of the EECHS-ARO, EECHS-ISSADE, PDU-SLno, and EACH-COA approaches in terms of dead nodes were 1500, 1600, 1700, and 1800, respectively. For 200 nodes, the performances of the EECHS-ARO, EECHS-ISSADE, PDU-SLn0, and EACH-COA approaches in terms of dead nodes were 1600, 1500, 1750, and 1850, respectively. In the 300-node scenario, the performances of the EECHS-ARO, EECHS-ISSADE, PDU-SLn0, and EACH-COA approaches in terms of dead nodes were 1750, 1700, 1850, and 2000, respectively. The overall network lifetime was improved by adopting the EACH-COA approach. It also reduces the convergence time when selecting a CH in the network.

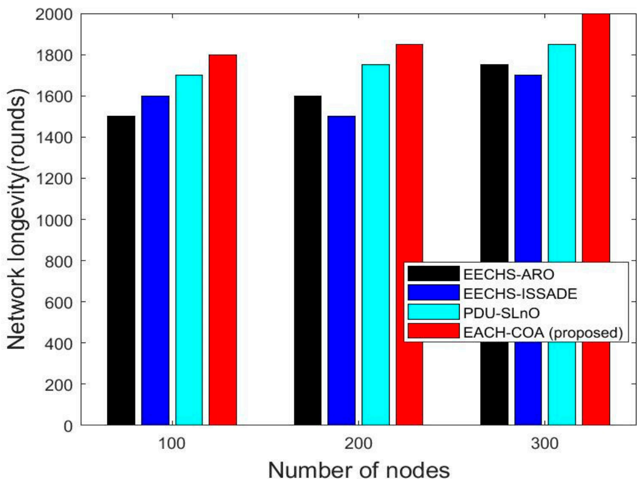

Figure 4. Network longevity vs. number of nodes.

Table 3 presents the network longevity based on the number of sensor nodes. The network performance in terms of lifetime was extended in different scenarios when the network sizes were 100, 200, and 300. The COA outperformed similar methods by reducing the convergence period during CH selection.

Table 3. Network longevity vs. number of nodes.

Number of Nodes	Network Longevity			
	EECHS-ARO	EECHS-ISSADE	PDU-SLnO	EACH-COA
100	1500	100	1500	100
200	1600	200	1600	200
300	1750	300	1750	300

5.2. Throughput

The amount of data determines the throughput transmitted to other nodes in the network. In Figure 5, the throughput is displayed for various network sizes, including 100, 200, and 300. The number of data packets received by the sink node from the CH in the network size 100 scenario is 130,000, 140,000, 170,000, and 180,000 for EECHS-ARO, EECHS-ISSADE, PDU-SLno, and EACH-COA, respectively. The number of data packets received by the sink node from the CH in the network size 200 scenario is 220,000, 200,000, 240,000, and 260,000 for EECHS-ARO, EECHS-ISSADE, PDU-SLno, and EACH-COA, respectively. The number of data packets received by the sink node from the CH in the network size 300 scenario: 310,000, 300,000, 360,000, and 400,000 for EECHS-ARO, EECHS-ISSADE, PDU-SLno, and EACH-COA, respectively. It is clear that the COA transmitted more packets to the sink node in the three different scenarios compared to other similar algorithms: EECHS-ARO, EECHS-ISSADE, and PDU-SLno. This is also a reason for the enhanced throughput and network lifetime compared to other algorithms.

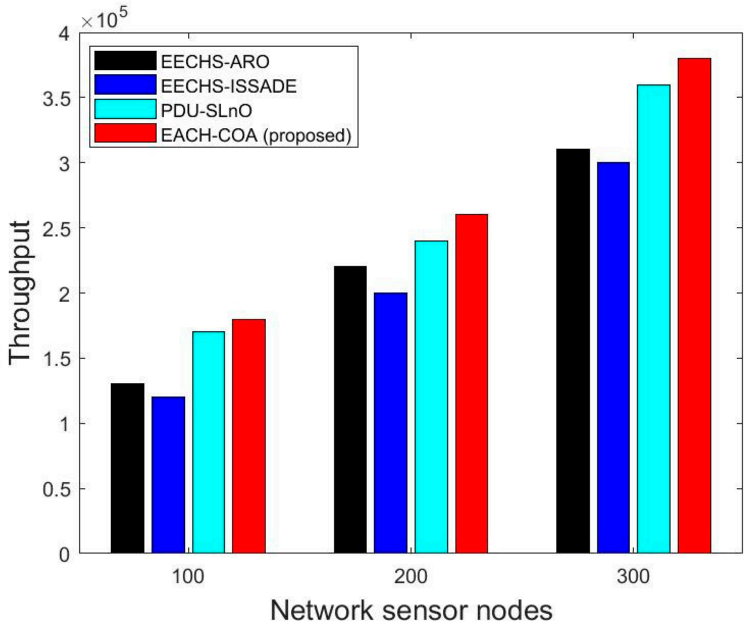

Figure 5. Throughput vs. number of nodes.

Table 4 lists the throughput of the proposed EACH-COA with respect to the various node numbers. The sink node received a more significant number of packets in comparison to EECHS-ARO, EECHS-ISSADE, and PDU-SLnO. The proposed COA requires less time to converge during CH rotation within the network.

Table 4. Throughput vs. number of nodes.

Number of Nodes	Network Longevity			
	EECHS-ARO	**EECHS-ISSADE**	**PDU-SLnO**	**EACH-COA**
100	130,000	140,000	170,000	180,000
200	220,000	200,000	240,000	260,000
300	310,000	300,000	360,000	400,000

5.3. Average Energy Consumption

The amount of energy consumed during data transmission is determined by the energy consumption. Figure 6 presents the average energy consumption for several network rounds. The average energy consumptions of the EECHS-ARO, EECHS-ISSADE, PDU-SLnO, and EACH-COA were 0.55 J, 0.50 J, 0.47 J, and 0.39 J, respectively. The proposed

EACH-COA approach minimizes energy consumption by 16%, 6%, and 8% compared to EECHS-ARO, EECHS-ISSADE, and PDU-SLnO, respectively, for the 3000th network round. The proposed approach has a shorter convergence time for CH selection. This is because the COA considers the RER and distance parameters to compute the fitness function. It has been noted that there is an escalation in energy consumption as the number of network rounds continues to escalate.

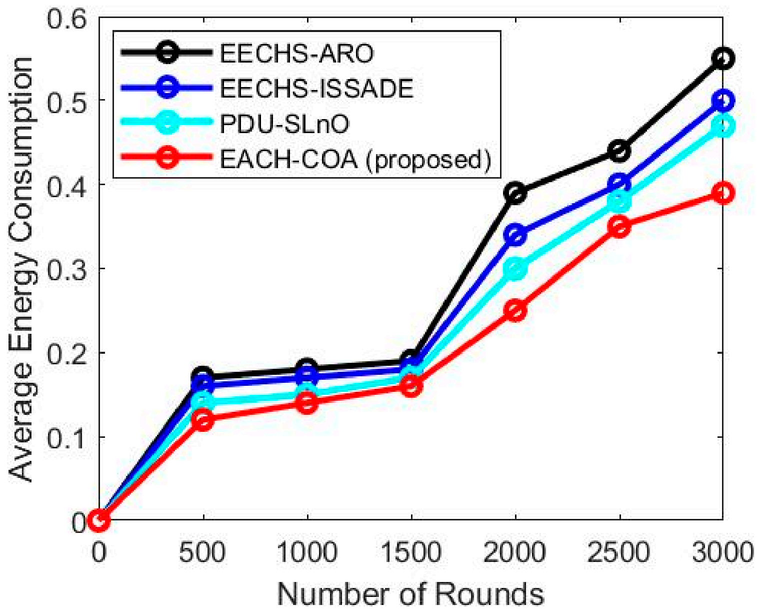

Figure 6. Average energy consumption vs. number of network round.

The average energy utilization of several techniques in association with several network rounds is listed in Table 5. The proposed EACH-COA consumed less energy than the other algorithms. It was found that energy consumption increased as the number of network rounds increased.

Table 5. Average energy consumption vs. number of rounds.

Number of Rounds	Average Energy Consumption (Joule)			
	EECHS-ARO	EECHS-ISSADE	PDU-SLnO	EACH-COA
0	0	0	0	0
500	0.17	500	0.17	500
1000	0.18	1000	0.18	1000
1500	0.19	1500	0.19	1500
2000	0.39	2000	0.39	2000
2500	0.44	2500	0.44	2500
3000	0.55	3000	0.55	3000

5.4. Network Stabilization Period

Figure 7 shows the network stability period for various network sizes. The proposed EACH-COA achieved a stable network state compared to EECHS-ARO, EECHS-ISSADE, and PDU-SLno. The COA converges rapidly. The reduction in the energy consumption within the network nodes was attributed to the use of this technique. This approach is best suited for networks that exhibit high node density.

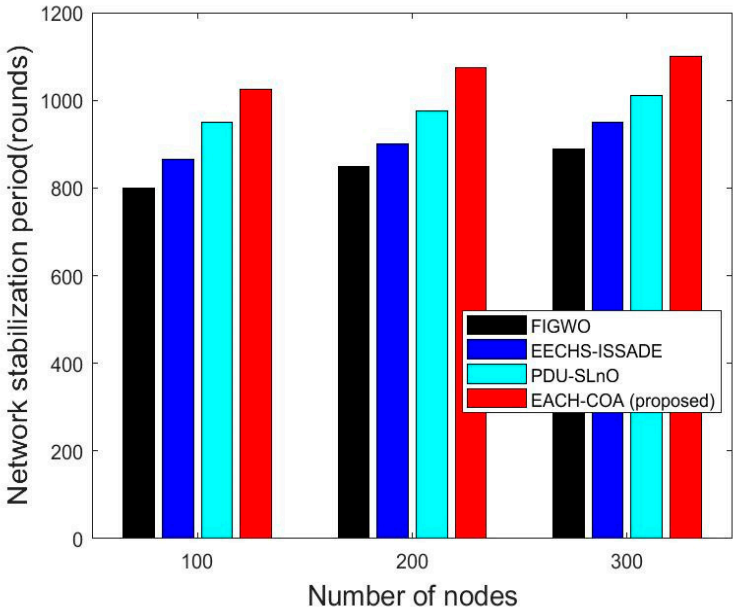

Figure 7. Network stabilization vs. number of nodes.

Table 6 shows the network stability period concerning different network nodes. The EACH-COA approach keeps the network more stable than the EECHS-ARO, EECHS-ISSADE, and PDU-SLno methods. It was discovered that the network is stabilized at the 1100th round of EACH-COA with a 300-node network and uses less energy in the network nodes.

Table 6. Network stabilization period vs. number of nodes.

Number of Nodes	Network Longevity			
	EECHS-ARO	**EECHS-ISSADE**	**PDU-SLnO**	**EACH-COA**
100	800	100	800	100
200	850	200	850	200
300	900	300	900	300

6. Conclusions and Future Work

Energy conservation has emerged as a crucial undertaking in WSNs, owing to the limited-capacity batteries installed in the network nodes. Clustering is the most efficient technique for conserving energy by retrieving data from all participants and relaying them to the sink in a cluster. This paper introduced EACH-COA, an energy-aware CH selection-based COA designed to increase the longevity of WSNs and decrease their energy consumption. The proposed EACH-COA approach first performs cluster formation using Euclidean distance, and then the CH is selected by adopting the COA. The performance of the EACH-COA protocol is verified for different network sizes. Therefore, the network longevity and throughput are enhanced by 8-15% and 5-10% compared with the EECHS-ARO, EECHS-ISSADE, and PDU-SLno approaches. As a result, energy conservation is maximized.

In future work, the performance of the proposed EACH-COA approach will be evaluated in real-time settings. A hybrid approach will be developed to select the optimal CH by reducing the convergence time in the network. To enhance the overall performance of the network, several other network parameters will be considered when determining the fitness function.

Author Contributions: Conceptualization, R.S.; methodology, R.S.; validation, R.S., Y.C. and B.K.M.; formal analysis, R.S.; investigation, R.S.; writing—original draft preparation, R.S.; writing—review and editing, R.S., Y.C. and B.K.M.; supervision, Y.C.; project administration, Y.C. and B.K.M.; funding acquisition, Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2023-2020-0-01489) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation). This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (RS-2023-00259703) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation). This research was supported by Interdisciplinary Program in IT-Bio Convergence System (BK21 Plus), 21Sunchon National University, 255, Jungang-ro, Suncheon-si, Jeollanam-do 57922, Republic of Korea.

Data Availability Statement: The dataset used for this work was randomly generated in MATLAB.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Sennan, S.; Ramasubbareddy, S.; Balasubramaniyam, S.; Nayyar, A.; Kerrache, C.A.; Bilal, M. MADCR: Mobility aware dynamic clustering-based routing protocol in Internet of Vehicles. *China Commun.* **2021**, *18*, 69–85. [CrossRef]
2. Nauman, A.; Qadri, Y.A.; Amjad, M.; Zikria, Y.B.; Afzal, M.K.; Kim, S.W. Multimedia Internet of Things: A comprehensive survey. *IEEE Access* **2020**, *8*, 8202–8250. [CrossRef]
3. Sennan, S.; Ramasubbareddy, S.; Balasubramaniyam, S.; Nayyar, A.; Abouhawwash, M.; Hikal, N.A. T2FL-PSO: Type-2 Fuzzy Logic-based Particle Swarm Optimization Algorithm used to Maximize the Lifetime of Internet of Things. *IEEE Access* **2021**, *9*, 63966–63979. [CrossRef]
4. Kassab, W.A.; Darabkh, K.A. A–Z survey of Internet of Things: Architectures, protocols, applications, recent advances, future directions and recommendations. *J. Netw. Comput. Appl.* **2020**, *163*, 102663. [CrossRef]
5. Thuluva AS, S.; Somanathan, M.S.; Somula, R.; Sennan, S.; Burgos, D. Secure and efficient transmission of data based on Caesar Cipher Algorithm for Sybil attack in IoT. *EURASIP J. Adv. Signal Process.* **2021**, *2021*, 1–23. [CrossRef]
6. Roy, S.; Mazumdar, N.; Pamula, R. An optimal mobile sink sojourn location discovery approach for the energy-constrained and delay-sensitive wireless sensor network. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 10837–10864. [CrossRef]
7. Palanisamy, S.; Sankar, S.; Somula, R.; Deverajan, G.G. Communication Trust and Energy-Aware Routing Protocol for WSN Using DS Theory. *Int. J. Grid High Perform. Comput. (IJGHPC)* **2021**, *13*, 24–36. [CrossRef]
8. Kandris, D.; Nakas, C.; Vomvas, D.; Koulouras, G. Applications of wireless sensor networks: An up-to-date survey. *Appl. Syst. Innov.* **2020**, *3*, 14. [CrossRef]
9. Shahraki, A.; Taherkordi, A.; Haugen, Ø.; Eliassen, F. Clustering objectives in wireless sensor networks: A survey and research direction analysis. *Comput. Netw.* **2020**, *180*, 107376. [CrossRef]
10. Goyat, R.; Kumar, G.; Alazab, M.; Saha, R.; Thomas, R.; Rai, M.K. A secure localization scheme based on trust assessment for WSNs using blockchain technology. *Future Gener. Comput. Syst.* **2021**, *125*, 221–231. [CrossRef]
11. Nabavi, S.R.; Eraghi, N.O.; Torkestani, J.A. WSN routing protocol using a multiobjective greedy approach. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 6664669. [CrossRef]
12. Sah, D.K.; Amgoth, T. Renewable energy harvesting schemes in wireless sensor networks: A survey. *Inf. Fusion* **2020**, *63*, 223–247. [CrossRef]
13. Priyadarshi, R.; Gupta, B.; Anurag, A. Deployment techniques in wireless sensor networks: A survey, classification, challenges, and future research issues. *J. Supercomput.* **2020**, *76*, 7333–7373. [CrossRef]
14. Ravi, G.; Kashwan, K.R. A new routing protocol for energy efficient mobile applications for ad hoc networks. *Comput. Electr. Eng.* **2015**, *48*, 77–85. [CrossRef]
15. El Khediri, S.; Fakhet, W.; Moulahi, T.; Khan, R.; Thaljaoui, A.; Kachouri, A. Improved node localization using K-means clustering for Wireless Sensor Networks. *Comput. Sci. Rev.* **2020**, *37*, 100284. [CrossRef]
16. Reddy, M.; Babu, M.R. Implementing self adaptiveness in whale optimization for cluster head section in Internet of Things. *Clust. Comput.* **2019**, *22*, 1361–1372. [CrossRef]
17. Amutha, J.; Sharma, S.; Sharma, S.K. Strategies based on various aspects of clustering in wireless sensor networks using classical, optimization and machine learning techniques: Review, taxonomy, research findings, challenges and future directions. *Comput. Sci. Rev.* **2021**, *40*, 100376. [CrossRef]
18. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
19. Sennan, S.; Balasubramaniyam, S.; Luhach, A.K.; Ramasubbareddy, S.; Chilamkurti, N.; Nam, Y. Energy and delay aware data aggregation in routing protocol for Internet of Things. *Sensors* **2019**, *19*, 5486. [CrossRef]

20. Sankar, S.; Ramasubbareddy, S.; Luhach, A.K.; alnumay, W.S.; Chatterjee, P. NCCLA: New caledonian crow learning algorithm based cluster head selection for Internet of Things in smart cities. *J. Ambient Intell. Humaniz. Comput.* **2022**, *13*, 4651–4661. [CrossRef]
21. Behera, T.M.; Mohapatra, S.K.; Samal, U.C.; Khan, M.S.; Daneshmand, M.; Gandomi, A.H. Residual energy-based cluster-head selection in WSNs for IoT application. *IEEE Internet Things J.* **2019**, *6*, 5132–5139. [CrossRef]
22. Khan, M.A.R.; Shavkatovich, S.N.; Nagpal, B.; Kumar, A.; Haq, M.A.; Tharini, V.J.; Karupusamy, S.; Alazzam, M.B. Optimizing hybrid metaheuristic algorithm with cluster head to improve performance metrics on the IoT. *Theor. Comput. Sci.* **2022**, *927*, 87–97. [CrossRef]
23. Savaglio, C.; Fortino, G. A simulation-driven methodology for IoT data mining based on edge computing. *ACM Trans. Internet Technol. (TOIT)* **2021**, *21*, 1–22. [CrossRef]
24. Ramalingam, R.; Saleena, B.; Basheer, S.; Balasubramanian, P.; Rashid, M.; Jayaraman, G. EECHS-ARO: Energy-efficient cluster head selection mechanism for livestock industry using artificial rabbits optimization and wireless sensor networks. *Electron. Res. Arch.* **2023**, *31*, 3123–3144. [CrossRef]
25. Chaurasia, S.; Kumar, K.; Kumar, N. MOCRAW: A Meta-heuristic Optimized Cluster head selection based Routing Algorithm for WSNs. *Ad Hoc Netw.* **2023**, *141*, 103079. [CrossRef]
26. Sankar, S.; Ramasubbareddy, S.; Dhanaraj, R.K.; Balusamy, B.; Gupta, P.; Ibrahim, W.; Verma, R. Cluster Head Selection for the Internet of Things Using a Sandpiper Optimization Algorithm (SOA). *J. Sens.* **2023**, *2023*, 3507600. [CrossRef]
27. Kumar, A.; Webber, J.L.; Haq, M.A.; Gola, K.K.; Singh, P.; Karupusamy, S.; Alazzam, M.B. Optimal cluster head selection for energy efficient wireless sensor network using hybrid competitive swarm optimization and harmony search algorithm. *Sustain. Energy Technol. Assess.* **2022**, *52*, 102243. [CrossRef]
28. Yadav, R.K.; Mahapatra, R.P. Hybrid metaheuristic algorithm for optimal cluster head selection in wireless sensor network. *Pervasive Mob. Comput.* **2022**, *79*, 101504. [CrossRef]
29. Kathiroli, P.; Selvadurai, K. Energy efficient cluster head selection using improved Sparrow Search Algorithm in Wireless Sensor Networks. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 8564–8575. [CrossRef]
30. Sengathir, J.; Rajesh, A.; Dhiman, G.; Vimal, S.; Yogaraja, C.A.; Viriyasitavat, W. A novel cluster head selection using Hybrid Artificial Bee Colony and Firefly Algorithm for network lifetime and stability in WSNs. *Connect. Sci.* **2022**, *34*, 387–408. [CrossRef]
31. Abu Salem, A.O.; Shudifat, N. Enhanced LEACH protocol for increasing a lifetime of WSNs. *Pers. Ubiquitous Comput.* **2019**, *23*, 901–907. [CrossRef]
32. Dehghani, M.; Montazeri, Z.; Trojovská, E.; Trojovský, P. Coati Optimization Algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems. *Knowl.-Based Syst.* **2023**, *259*, 110011. [CrossRef]
33. Sankar, S.; Srinivasan, P.; Luhach, A.K.; Somula, R.; Chilamkurti, N. Energy-aware grid-based data aggregation scheme in routing protocol for agricultural internet of things. *Sustain. Comput. Inform. Syst.* **2020**, *28*, 100422. [CrossRef]
34. Sahoo, B.M.; Amgoth, T.; Pandey, H.M. Particle swarm optimization based energy efficient clustering and sink mobility in heterogeneous wireless sensor network. *Ad Hoc Netw.* **2020**, *106*, 102237. [CrossRef]
35. Sathyamoorthy, M.; Kuppusamy, S.; Dhanaraj, R.K.; Ravi, V. Improved K-means based q learning algorithm for optimal clustering and node balancing in WSN. *Wirel. Pers. Commun.* **2022**, *122*, 2745–2766. [CrossRef]