

A comparative study on energy efficient clustering based on metaheuristic algorithms for WSN

S. Sowndeswari^{1, 2*} and E. Kavitha³

Research Scholar, Department of Electronics and Telecommunication Engineering, Sir M Visvesvaraya Institute of Technology, Bangalore, Karnataka, India¹

Assistant Professor, Department of Electronics & Communication Engineering, Sambhram institute of Technology, Bangalore, Karnataka, India²

Professor & Head, Department of Electronics & Telecommunication Engineering, Sir M Visveswaraya Institute of Technology, Bangalore, Karnataka, India³

Received: 20-September-2021; Revised: 19-January-2022; Accepted: 20-January-2022

©2022 S. Sowndeswari and E. Kavitha. This is an open access article distributed under the Creative Commons Attribution (CC BY) License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

One of the important technologies in wireless sensor networks (WSN) is an efficient and dependable routing system. In WSN, energy is the key resource for extending network lifespan. Nowadays, WSN is utilized for a variety of applications, and there is always an issue with energy consumption. As a result, to find out the best energy efficiency model is the primary focus of this research in order to extend the life of the network. Different energy-efficient clustering (EEC) methods based on artificial bee colony optimization (ABC) are applied for the WSN in this comparative research to increase the network's energy efficiency. To increase energy efficiency during the communication process, the enhanced memetic artificial bee colony (EMABC), global artificial bee colony algorithm based on the cross over and tabu search (CGTABC), Energy-efficient clustering using artificial bee colony (EC-ABC), memetic artificial bee colony algorithm (MeABC), and randomized memetic artificial bee colony algorithm (RMABC) are implemented. The EEC method involves forming the appropriate quantity of clusters and selecting cluster heads in a dynamic manner. Furthermore, relay nodes are chosen with energy efficiency objectives, and they are computed afterwards. The implemented methods perform EEC. In terms of assessment and validation, the implemented models are compared. As a consequence, the proposed EMABC model performed well in terms of energy efficiency, with an efficiency of 82.44%, an end-to-end delay of 99.68ms, a packet drop rate of 152, a throughput of 680.28Kbps, packet delivery ratio of 98.05%, and network lifetime of 91%.

Keywords

WSN, Energy efficiency, Network lifetime, Clustering, Routing, Enhanced memetic artificial bee colony.

1. Introduction

A wireless sensor network is a network that connects sensor units wirelessly to observe and monitor environmental factors [1]. Wireless sensor networks (WSNs) are a subset of ad-hoc networks. The energy limitation on WSN nodes is critical, and it is thoroughly analyzed to make WSN robust, reliable, and useful in real-world cases. Sensor nodes operate on a limited quantity of energy supplied by a small direct current (DC) source within the sensor node unit, which cannot be changed once the network is deployed.

In addition to energy, other important restrictions of WSN include limited sensing range, transmission range, sensor node processing capabilities, and sensor node memory capacity [2]. A sensor node's sensing range describes the location around the sensor that it can observe or monitor. In contrast, the transmitting range defines the location around the sensor till the sensor node may communicate data gathered by it. Because detecting and transmitting are the two most costly activities of a sensor node, these two restrictions are critical. The design of the network has a serious effect on the scalability and performance of the WSN. The architecture of WSN can be designed on two types: flat and two-tier or hierarchical [1].

* Author for correspondence

Saving node energy while preserving desired network behaviour is the most challenging problem in WSN design. Any WSN can only carry out its task for as long as it is considered alive and not after that. As a result, any energy-saving approach seeks to increase network lifespan. At first, each sensor has a finite quantity of energy. Nonetheless, while communication takes place, energy is drained. Because the target location is generally unattended, replacing or recharging sensor nodes is not always possible. In this situation, if a single node dies, the network may become inoperable. As a result, reducing energy usage and increasing network lifetime are two significant challenges in a WSN [3]. The cluster head (CH) in the network is the main node in the clustering protocol, and it is dependable for gathering data sensed by the member node and transmitting it to the base stations (BS). Member node just require to interact with their relevant CHs over a small distance, requiring minimum energy [2]. The CH can interact with the BS in various methods, including directly transmitting the data to the BS or utilizing various nodes as following hop to indirectly transfer information to the BS. The clustering methods attempt to find appropriate CH sets and rotates the function of CH across every node in order to equalize node's consumption of energy [4]. A clustered wireless sensor network is depicted in *Figure 1*.

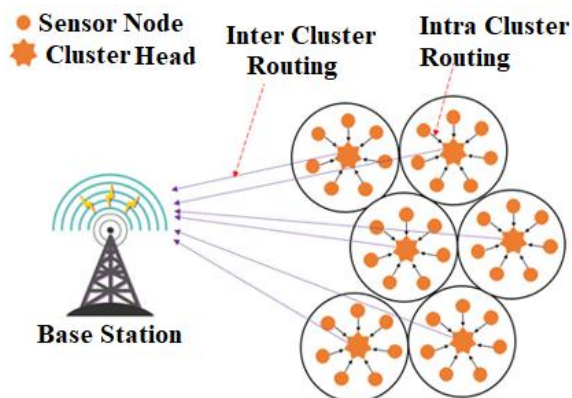


Figure 1 Clustered wireless sensor network

Routing is one of the most difficult issues for which a deterministic algorithm cannot be used. As a result, optimization techniques are utilized to offer low-cost pathways among a large number of options. A routing protocol is required among sensor nodes (routers) of the WSN to describe a path among a source and the destination nodes for effective communication. Routing protocols would provide communication channels among sensor and the sink

nodes. These routings are designed to maximize the network's lifespan. Several energy-efficient routing methods are designed to disperse load among every sensor node, lowering energy consumption in the WSN. Routing protocols are divided into two types: flat routing protocols and hierarchical routing systems. Each sensor node in a flat-based protocol was at the equivalent level and has an individual global address, but some sort of hierarchy is used in a hierarchical method like tree-based clustering (TBC), low-energy adaptive clustering hierarchy (LEACH), and centralized-low-energy adaptive clustering hierarchy (C-LEACH).

To produce a well-functional design that meets the criteria, optimization is required. To accomplish the intended aim of decreasing energy consumption or improving network lifetime, network optimization is necessary. Consumption of energy, security, network longevity, and packet delivery ratio were all important routing problems in a WSN. Data routing protocols in WSNs that are based on bio-inspired swarm intelligence (BISI) models are extremely effective owing to the similarities in how specific agents in every collection operate, which has finally result in a lot of research efforts in their creation and optimizations. The main objective of this work is to perform a comparative analysis on energy efficient clustering (EEC) using meta-heuristic algorithms and to find the best algorithm by performance comparison. In this work, a comparative study of metaheuristic swarm intelligent algorithms for the performance of WSNs energy-efficient clustering and routing is carried out. The performances of different metaheuristic algorithms are compared to find out the best algorithm for energy-efficient clustering and routing of WSN.

2.Related works

A clustering technique that used an improved artificial bee colony ABC algorithm to identify CHs was proposed in [4]. An effective improved ABC method was proposed based on the conventional ABC technique. Then, to tackle the clustering issue in WSN, the network CH energy, CH density, CH location, and other comparable parameters were included in the improved ABC algorithm theory. To determine the best clustering approach, the improved ABC algorithm was employed for optimizing fuzzy C-means clustering. This work was done on the basis of a fixed network, the proposed method has limitations. Firefly algorithm with particle swarm optimization (PSO), a hybrid method for determining the selection of best CH in the C-LEACH algorithm

was proposed in [5]. Using PSO, the hybrid algorithm enhanced the global search behaviour of fireflies and obtained optimal CH placement. In [6] an optimized energy efficiency in the wireless environment by combining the Gaussian elimination approach with distributed energy-efficient clustering (DEEC-Gauss) was implemented. However, this model has a node localization issue that has to be solved [6].

A unique CH selection method based on a grey wolf optimization algorithm was developed in [7]. Also, the objective functions and weight factors for effective CH selection and cluster formation were developed. The method was run in a variety of test scenarios with varied numbers of sensors and CHs. However, several concerns, such as load balancing and fault tolerance in WSN, might have been taken into account. An energy efficient clustering (EEC) leader selecting method based on ABC optimization and k-means clustering was proposed in [8]. Clusters were formed by k-mean clustering, and then optimum cluster leaders were selected by ABC optimization. The sensor's energy, the distance among a cluster leaders and sink, and the distance among members and cluster leaders comprised the optimization objective function. It was discovered that placing the sink in the center of the WSN resulted in the highest energy efficiency.

An efficient approach called chaotic whale metaheuristic energy-optimized data gathering (CWMEODG) is used to improve the data collection by using least amount of energy in WSN was proposed [9]. The Chaotic tent map mathematical approach was used for the CWMEODG approach to discover the optimal global solution and quick convergence rate. As a result, the delay at the sink was reduced [9]. The integration of the Butterfly Optimization and Ant Colony Optimizations to minimize overall energy usage and enhance network lifespan was proposed in [10]. The bat optimization algorithm (BOA) was utilized to elect the best cluster heads (CHs) from the set of nodes, and the ant colony optimization (ACO) was used to determine the path between the CHs and the base station (BS). Optimizing ACO with residual energy, node degrees, and distance parameters yielded the most energy-efficient routes selection.

Using LEACH, an energy-efficient routing protocol, combined with an optimization method genetic algorithm (O-LEACH) to improve energy efficiency with sensor node longevity was proposed in [11]. The

simulation of a WSN became faster, and the total transmission latency was decreased by using the O-LEACH routing protocol. However, throughput may be enhanced while energy usage was lowered. Multi-objective coverage optimizations method (MOCADMA) for WSN that models WSN coverage controls as a multi-objective optimization was developed [12]. MOCADMA optimized WSN coverage using the memetic algorithms and the dynamic local search approach, achieved objectives like higher networks coverage, efficient node usage, and higher residual energy.

An EEC protocol based on a multi-weight chicken swarm evolutionary algorithm to increase network life span and energy efficiency was proposed in [13]. This protocol contributed to increased energy efficiency during network communication. There was a potential of receiving communication delay while using this protocol in a big scale-based application. An efficient and dependable WSNs clustering method was developed [14] based on the quantum ABC algorithm that targeted imbalanced load in WSN clustering without taking into account node residual energy, node location, node intensity, and so on. This method could better adapt to network topology, analyze network node balanced distribution, and ensure that every round might pick the CH with the correct ratio, reducing node energy consumption and extending network lifespan. A secure and energy-efficient optimization approach based on the Dij-Huff Method (DHM) was proposed [15]. Dijkstra's algorithm (DA) and Huffman coding (HC) were combined to form the DHM. This method was capable of delivering data to nodes with greater energies while also providing a secure connection. This approach reduced packet loss and latency while increasing network longevity.

In [16], a dual-cluster-head mechanism was proposed, which minimized the energy overhead during the rotation of Cluster Heads (CHs). In addition, a non-cooperative game model for balancing energy consumption among Cluster Heads was presented. Furthermore, the game model's Nash Equilibrium Point (NEP) was presented, along with the related proof. Following that, the Energy-efficient Clustering algorithm, which incorporated Game theory and the Dual-cluster-head (ECGD) mechanism, was detailed, taking into account energy efficiency in both intra-cluster and inter-cluster communication.

Hassan et al. [17] presented an improved energy-efficient clustering protocol (IEECP) to extend the lifetime of the WSN. The IEECP was divided into three sections sequentially. First, an ideal number of clusters for the overlapping balanced clusters was identified. The balanced-static clusters were then produced by combining a modified fuzzy C-means algorithm with a method to reduce and balance the energy consumption of the sensor nodes. Finally, cluster heads were chosen at ideal positions by rotating the CH function among cluster members using a new CH selection-rotation algorithm that combined a back-off timing mechanism for CH selection and a rotation mechanism for CH rotation. The protocol, in particular, lowered and balanced node energy consumption by strengthening the clustering structure, while IEECP was ideal for networks requiring a long lifetime. A selective-path priority table energy-balanced (EB) and energy-efficient (EE) clustering-based routing system with a mobile sink was proposed in [18]. The priority table was created by ranking the two shortest paths to the cluster head or sink using some simple but effective algorithms. Some routing characteristics, such as transmission range/power and residual energy, were used to derive the rules.

A hybrid energy-efficient clustering based on fuzzy C means and differential evolution method (EEFCM-DE) based on fuzzy clustering and fuzzy-based evolutionary technique was proposed in [19]. The objective was to use FCM to create clusters and then apply an evolutionary process DE to select the best node as a CH inside each cluster generated. The fitness of each node was calculated for CH selection using a specially constructed fuzzy inference method. In [20], an Ant Colony Optimization (ACO) integrated Glowworm Swarm Optimization (GSO) approach (ACI-GSO) model was proposed to identify the optimal CH for energy-efficient routing protocol in WSN. ACI-GSO chose the CH node based on three parameters: distance, latency, and energy. Finally, the adopted model efficiently determined the performance of the work.

In [21], the BM-BWO with fuzzy logic-based hybrid energy efficient distributed clustering (HEED) protocol (BMBWFL-HEED) was proposed. To select the highest residual energy, the Boosted Mutation-based Black Widow Optimization (BMBWO) method was used. The Fuzzy Logic System was used to identify the best cluster head (FLC). The fuzzy logic system chooses the most appropriate and optimal cluster heads. Energy Efficient Centroid-

based Ant colony Optimization (EECAO) protocol was proposed in [22] to increase sensor network performance in WSN-assisted Internet of Things (IoT) contexts. The protocol used centroid-based clustering to collect information from local clusters and ant colony optimization to communicate that information to the base station. To reduce long-distance communication, an ant routing model was used to optimize the path between the super cluster heads and the base station.

Overall, most of the research works were based on improving energy efficiency and network lifetime. Here in this comparative analysis based on the performances of the discussed literatures were compared and validated in terms of energy efficiency, end-to-end delay, network throughput, packet drop rate and packet delivery ratio. Finally, the best performed technique will be suggested for the further analysis and research in future.

3.Methodologies for comparative study

The technique of addressing NP-hard issues is known as optimization. Deterministic methods cannot solve NP-hard problems in a reasonable amount of time. The objective function, which could be maximized or reduced based on the issue, is used in optimization approaches. Because some high-level or metaheuristic processes can be utilized, this optimization methodology is sometimes known as metaheuristic optimization. Optimization strategies can have either a multi-objective or single-objective function as their objective function. In a single-objective function, all points converge to a single point, which was the optimal solution. When particles combine at more than two points, it indicates that more than two optimum solutions exist. In the event of a multi-objective function, however, the best one among them is chosen. Some optimization algorithms utilize local search to identify the optimal solution, while others use global search. Some techniques utilize local search to improve a single solution, which was exploitations-oriented, in order to identify the best solution among several solutions. Simultaneously, the other is population-based, and it investigates the global search for its functioning.

3.1Design challenges in WSN

WSN can be differentiate by a variety of features. Initially, WSN has unique traffic patterns such as many to one and one to many trees. Second, sensor nodes in WSNs have severe operating resource restrictions, like limited memory, battery power, low bandwidth connections, and so on. Third, because

each sensor node in a WSN might continually create an excess amount of data, there is a greater requirement for data aggregation at different levels in the network to identify and remove superfluous data, increasing network traffic. The specific design difficulties of WSNs will be explored one by one in the following section:

3.1.1 Memory and computation requirements

Agents in WSNs are small computers with low-processing-power CPUs and limited onboard memory. To ensure that functionalities are executed efficiently, the routing algorithms designed must have a low processing overhead.

3.1.2 Self-organization and automaticity

A WSN is a dynamic network; many nodes may become non-operational throughout its operation, while many new nodes are included to the networks. The routing protocols employed should be proactive; in such a continuously changing network, it should be adaptable enough to ensure the longer-term availability of important services of network.

3.1.3 Energy efficiency

Because sensor nodes have constrained battery power, efficient power usage was important in WSNs in order to enable extended network operating times. Hence, the routing protocol is required to:

- Reduce the total transmissions involved in route discovery and data delivery.
- Forwarding data in the protocol should be done in multi-paths to ensure that all nodes' energy usage was uniform across the operation.

3.1.4 Scalability of the network

WSN applications often install thousands of sensor nodes in a dense dispersed pattern over a testbed. Because the nodes in these dense networks have limited communication range, signal interference causes a significant failure rate. As a result, a routing system must be resistant to difficulties such as intense radio interferences and unpredictability in longer data forwarding pathways

3.1.5 Data security in a network

Data security is a significant problem in WSN. Unauthorized access owing to insufficient network security might breach quality of service (QoS) agreements. Because of the data broadcasting nature in WSN, it may result in increased data exposure. Because the physical route of transmission was inherently unsafe, there is a requirement for security-aware data routing algorithms to be developed.

WSNs are made up of nodes that range in number from a few to millions. Clustering such millions of nodes and determining the optimum path or route between them is a difficult process. The key needs in WSN are clustering and selecting the best available

channel for data transfer in order to reduce energy consumption and enhance packet delivery ratio. Consequently, the overall network lifetime was increased, and suspicious nodes in data transfer can be avoided. In comparison to traditional clustering algorithms, a better clustering based on population-based optimization approach is necessary to obtain improved WSN performance. Enhanced Memetic Artificial Bee Colony (EMABC) optimization is one such strategy, which is a population-based metaheuristic approach for achieving optimization in WSN.

3.2 Review of ABC algorithm

Artificial bee colonies are divided into three groups: employed, onlookers, and scouts. The employed bee's task is to find food or a food source. After discovering a food supply, a bee must communicate or share with other bees about the nectar. The observer bee is waiting for the food. The duration of their leap is determined by the amount of nectar consumed. A large supply of nectar will attract more bees, and many additional bees will follow. The ABC algorithm emulates the same phenomenon. Employed bees seek for solutions at random, and the quality of the solutions is determined by the amount of nectar available, i.e., fitness value.

3.2.1 Initialization stage

In the conventional ABC, an employed bee was produced at random within its boundaries and the Equation 1 will be evaluated as.

$$x_{ij} = l_j + rand(0,1) \cdot (u_j - l_j) \quad (1)$$

Where, $x_{ij}, i = 1 \dots SN, j = 1 \dots D$ was a component of the employed bee X . The number of employed bees is SN , and every bee contains D components. The lower and upper limits of the search space are denoted by l_j and u_j , respectively.

3.2.2 Employed bee stage

Employed bees were utilized to find better food locations inside their search area. By altering one of its components, an employed bee x_{ij} could be updated and the Equation 2 will be evaluated as.

$$v_{ij} = x_{ij} + \phi_{ij} \cdot (x_{ij} - x_{kj}) \quad (2)$$

Where, v_{ij} is a component of the newly created food site v_i . x_{kj} is a random parent bee's element x_i that must be distinct from x_{ij} . ϕ_{ij} is a real number having a uniform distribution between $[-1, 1]$. It regulates the movement of the new source of food relative to the food source of parent. The following Equation 3 can be used to assess the fitness of the v_i .

$$fit_i = \begin{cases} 1/(1 + f_i) & \text{if } f_i \geq 0 \\ 1 + abs(f_i) & \text{if } f_i < 0 \end{cases} \quad (3)$$

Where, f_i is the v_i cost value. The more fitness of the two v_i and x_i would survive the following generations. Finally, information about the location of food sources would be shared with onlooker bees.

3.2.3 Onlooker bee stage

With a roulette wheel selection, onlooker bees were liable for utilizing the food sources communicated by employed bees. The food location's probability being chosen by the following Equation 4:

$$p_i = \frac{fitness}{\sum_{i=1}^{SN} fitness_i} \quad (4)$$

The better the value of fitness of a food item, the more similar it is that onlooker bees will take use of it. Equations (2) and (3) were also utilized to keep the onlooker bees up to date.

3.2.4 Scout bee stage

According to the ABC, if a food supply's fitness value cannot be adjusted after a "limit" number of generations, the bee would abandon it and replace it with a new food source. In the search space, the scout bee produces the new source of food randomly. The basic ABC algorithm iteratively continues the preceding searching procedure until the specified stop condition is reached.

ABC is useful for slow convergences and may be utilized with other methods easily. Its energy usage has to be improved further because it is not as excellent as other techniques.

In the ABC method, each food source represents a feasible route between the sink and source nodes. Colonies of artificial bees change the position of food sources. Each colony is skilled at developing intellect that can be utilized to find food. The food-seeking operation is split into three key parts: employed food seekers who create new food sources, onlooker food seekers who renew food sources based on nectar amount, and scout food seekers who locate new food sources in place of abandoned food sources. Step size in ABC is a combination of an arbitrary integer ϕ_{ij} , the current solution, and an arbitrarily determined solution. The quality of the updated solution is determined by the size of the step. If the size of step was very large, the upgraded solution exceeds the true solution; if the size of step is very small, the ABC's convergence rate may be considerably reduced. As a result, adding memetic computation into conventional ABC avoids the issue of exceeding

the true solution while preserving convergence speed [23].

3.3 Enhanced memetic artificial bee colony (EMABC) algorithm

To enhance quality, the memetic algorithm (MA) is a problem-specific local search evolutionary algorithm that balances algorithm exploitation and exploration. The method utilizes a local search strategy to reduce the possibility of premature convergence. The fitness function in an MA is a function that takes a potential solution to the issue as input and outputs how well the solution fits the problem. Memetic computation (MC) emerged from MA to reach a higher milestone in discovering solutions. To solve the problem, the memetic computation employs a combination of memes, which represent a set of information encrypted in complicated structures that interact with one another.

To enhance the WSN, the EMABC algorithm develops a secure cluster-based routing. This work is divided into three stages: clustering, secure CH selection, and secure routing path creation. The K-means clustering technique is initially used to split the network into clusters. The EMABC method is then used to identify the near-optimal secure CH. This EMABC takes into account four distinct fitness values: node trust value, residual energy, distance, and node degree. The EMABC's trust value consideration is utilized to avoid black hole attacks during CH selection. The black-hole attack is one of the many current attacks that degrade network speed and dependability by causing the malicious node to discard all incoming packets.

Furthermore, the EMABC algorithm establishes a safe route between the source and destination nodes. The EMABC routing, like the ad-hoc on-demand distance vector (AODV), generates the routing path using route request and reply messages. The route request messages are broadcast to the adjacent nodes by the source node during the route discovery process. Similarly, the secure data transmission channel is created between the source and destination nodes. Data packets are transferred to the target node when the transmission path has been determined. This work considered the best fitness values in the EMABC algorithm. The route reply message is then sent to the source node by the node with the highest fitness. As a result, the EMABC method outperforms any other population-based heuristic algorithm.

The original ABC algorithm divides food seekers into three categories: employed food seekers, onlooker food seekers, and scout food seekers. Only one employed food seeker discovers the availability of food. Onlooker food seekers use this information to determine which food source to visit. When the food supply is depleted, the unused food seekers become scout food seekers. In this functional system, each food source represents one of the N possible routes between every pair of nodes in WSNs. The location of a food supply reveals a potential routing solution. A food source's nectar amount is equivalent to the strength of the associated solution as evaluated by a fitness function. In addition, every node maintains two information in the neighbour table: the total time necessary for a packet to reach the next node, and unused node energy. To obtain the most accurate path in the search process, a local search space that is trailing in the original ABC algorithm should be enhanced. As a result of introducing memetic computation into the original ABC Algorithm, the local search space is improved by adding one more step to the original ABC Algorithm [24].

The population-based method EMABC of size N has candidate solutions a_i that are vectors of M design variables within S , the decision space. The following is how the initial population is produced at random and the Equation 5 is evaluated as:

$$a_{i,j} = rand(0,1) \cdot (u_b - l_b) + l_b \text{ for } j = 1,2, \dots M \tag{5}$$

Where, r and $(0,1)$ creates a random number between 0 and 1, and u_b , l_b indicate the upper and lower limits of a candidate solution a_i .

The following are the exploration tasks in EMABC:

1. Stochastic long-distance type (comparable to employed food seekers in ABC)
2. Stochastic moderate distance type (comparable to onlooker food seekers in ABC)
3. Deterministic long-distance type
4. Random long-distance type (comparable to scout food seekers in ABC)

3.3.1 Stochastic long-distance type

To obtain a trial solution, three operations are done on each of the candidate solutions a_i : mutations, crossover, and selections. The trial solutions are calculated using Equation 6.

$$u_i^{(t)} = a_{r_1}^{(t)} + (a_{r_2}^{(t)} - a_{r_3}^{(t)}) \tag{6}$$

where $a_{r_1}^{(t)}$, $a_{r_2}^{(t)}$, $a_{r_3}^{(t)}$ are randomly chosen candidate solutions, and t is the number of iterations. Equation

7 is used to compute the number of alterations allowed to the trial solution.

$$v_{i,j}^{(t)} = \begin{cases} u_{i,j}^{(t)} & \text{if } (rand_j(0,1) \leq CR || j = j_{r_4}) \\ a_{i,j}^{(t)} & \text{otherwise} \end{cases} \tag{7}$$

Where CR is the crossover probability and j_{r_4} is a random variable from the trial solution. Equation 8 is used to find the optimum solution between the candidate and trial solutions.

$$a_i^{(t+1)'} = \begin{cases} v_i^{(t)} & \text{if } f(v_i^{(t)}) \leq f(a_i^{(t)}) \\ a_{i,j}^{(t)} & \text{otherwise} \end{cases} \tag{8}$$

Where f denotes the fitness function.

3.3.2 Stochastic moderate distance type

Similar to the stochastic long-distance type, the procedure in getting the trial solution incorporates mutations, crossover, and selections using Equation 9.

$$u_i^{(t)} = a_i^{(t)} + (a_{sup}^{(t)} - a_i^{(t)}) + (a_{r_1}^{(t)} - a_{r_2}^{(t)}) \tag{9}$$

Where $a_{sup}^{(t)}$ indicates the best solution.

Equations (7) and (8) are used to execute the crossover and selection processes (8). At the end of this step, the best route is achieved by the onlooker food seekers based on the quality of the nectar amount.

3.3.3 Deterministic short distance type

This step attempts to put the candidate solution into local optima, with the main goal of preserving variety and avoiding the scenario of transcending the real solution.

3.3.4 Random long-distance type

When the food supply is depleted, the food seekers transform into scouts, and the search process is repeated in three distinct exploration tasks as stated above.

The Enhanced Memetic ABC algorithm is as follows: Step 1: Create a population of N equally distributed individual. Every individual x_{ij} was a food source (i.e., a needed solution) with D features. The dimension of the problem was indicated as D . The i^{th} solution in the j^{th} dimension was represented by x_{ij} .

Step 2: Compute the fitness of every individual solution.

Step 3: All employed bees, when put at different sources of food than others, seeks in the vicinity of their present place for a best source of food. Use the enhanced search phase, which was inspired by the Golden Section Search method.

Repeat until the termination requirements are met.

Determine the function's value using f_1 and f_2 .

If $f_{1.val} < f_{2.val}$ then

$b = f_2$ and solution is in interval $[a, b]$.

else

$a = f_1$ and solution is in interval $[a, b]$.

Using following equation, change the location of the solution.

$$x'_{ij} = x_{ij} + (x_{ij} - x_{kj}) \times f_1$$

Here $k=rand [0,1]$ *Food Number, $j=rand [0,1]$ * dimensions and $l= \{1,2\}$.

Indices $j \in \{1, 2, \dots, D\}$ and $k \in \{1, 2, \dots, N\}$ were selected at random. N was the number of bees in service. ϕ_{ij} was a homogeneous number randomly between $[-1, \text{and } 1]$.

Step 4: Determine the fitness of x_{ij} and v_{ij} . Use the greedy selections technique to choose the best one among them.

Step 5: Using the following formula, compute and normalize the values of probability, p_{ij} , for every solution x_i .

$$p_{ij} = \phi \times \left(\frac{fit_i}{\max_fitness} \right) + (1 - \phi) \times \left(\frac{fit_i}{\sum_{i=1}^N fit_i} \right)$$

Here ϕ was the number randomly between $[0,1]$.

Step 6: At random, allot every observer bee to the solution, x_i , with a probability proportional to p_{ij} . Use a better search phase inspired by the Golden Section Search method.

Repeat until the termination requirements are met.

Determine the function's value using f_1 and f_2 .

If $f_{1.val} < f_{2.val}$ then

$b = f_2$ and solution is in a range $[a, b]$

else

$a = f_1$ and solution is in a range $[a, b]$.

Using following equation, change the location of the solution.

$$x'_{ij} = x_{ij} + (x_{ij} - x_{kj}) \times f_1$$

Here $k=rand [0,1]$ *Food Number, $j=rand [0,1]$ * dimensions and $l= \{1,2\}$

Step 7: Provide fresh source of food for every observer bee, x'_{ij} .

Step 8: Determine the fitness of all observer bees, x_{ij} as well as the new solution, x'_{ij} . Using greedy selection development, choose the best fitness.

Step 9: If a certain solution x_{ij} was not enhanced after a predetermined number of iterations, it is flagged for condemnation. Change the solution with a scout bees placed at a source of food created at random inside the search area using

$$x_{ij} = x_{minj} + rand[0,1](x_{maxj} - x_{minj})$$

for $j = 1, 2, \dots, D$

Step 10: Note the better solutions discovered thus far.

Step 11: Review the end category. If the better solution discovered so far was adequate or has reached the highest number of iterations, end and return the best solutions discovered. Else, return to the step 2 and repeat.

3.4 Different artificial bee colony algorithms for comparison

3.4.1 Global artificial bee colony based on the crossover and tabu search (CGTABC)

This energy-balanced and protocol of energy efficient clustering (EEC) routing chose the best cluster heads (CHs) using an improved ABC algorithm. The member nodes connect to the relevant CHs, splitting the network into clusters. To reduce excessive energy consumption by CHs located distant from the base station (BS), an efficient routing algorithm was utilized to determine the best route among the CHs and the BS. To communicate the fused data to the BS, the CH employs a multi-hop communication technique [25].

Many clustering protocols employ the time division multiple access (TDMA) method for intra-cluster communication, which was extremely advantageous under ideal conditions where all nodes can always gather reliable data. In practice, however, achieving this ideal condition is frequently challenging. For a time, certain nodes may not detect valuable data. They will, however, be woken within their time slots, resulting in needless energy use and time loss. To address this issue, during the stabilization phase, the polling control approach based on idle/busy node was incorporated within intra-cluster communications to equalize energy usage and enhance network's performance.

Initialization of parameters: The number of employed bees and the number of spectator bees are both source node (SN), as is the highest number of times the source of nectar may be mined continually. Limit, the maximum iterations $maxCycle$, and cr , cross coefficient

Step 1: Set the populations by generating the populations of SN people at random, with the size of a vector defined by the optimum number of CH

Step 2: Compute and recall the optimum answer for each individual's fitness function value for iterations ranging from one to $maxCycle$ bees

Step 3: for $i=1$ to SN .

Step 4: Employed bees search

Step 5: A new source of nectar was discovered by performing a cross operation on the source of nectar

explored by the employed bees and iterative optimum source of nectar

Step 6: Ascertain that the new source of nectar falls in the zone of the optimized search

Step 7: The new source's fitness value of nectar was computed, and the superior nectar source is kept using the greedy approach

Step 8: end

Step 9: for $i=1$ to SN

Step 10: Determine the probability $P_{CGTAB(i)}$ that an onlooker bee selects source of nectar i

Step 11: end

Step 12: Onlooker bee

Step 13: for $i=1$ to SN

Step 14: According to the probability $P_{CGTAB(i)}$, an onlooker bee selects which employed bee to follow

Step 15: Onlooker bee transforms into employed bee and performs actions that correspond to employed bee

Step 16: end

Step 17: Put a source of nectar on tabu list if it satisfies the requirements for abandonment

Step 18: Scout bee

Step 19: A fresh nectar source is created by the scout

Step 20: Assure this new source of nectar was accessible and that the old source of nectar was replaced

Step 21: Note the present best source of nectar

Step 22: end

Step 23: Global optimal solutions consist of best CHs chosen for the present round.

3.4.2 Energy-efficient clustering using ABC

The combination of ABC optimization and k-means clustering was used to create this energy-efficient cluster leader selection technique. The sink chooses k cluster leaders, and the k-means clustering technique is used to generate k clusters. After the clusters were established, the ABC optimization was applied to find the best cluster leaders. The sensors, distance from sensors inside the cluster, distance to sink were the objective functions evaluated for optimization. The ABC performs to improve the functions in order to create WSN more efficient.

Step 1: Input: N is the count of sensors in WSN, c was the count of cluster leaders, and Z was the number of rounds.

Step 2: Output: Cluster leader's optimal position.

Step 3: Randomly initialize the food resources, X_{nc} , where n was between 1 and N and c was between 1 and k ;

Step 4: for $z=1$ and Z ; do

Step 5: for every employed bee; do

Step 6: Select a new leader of cluster candidate;

Step 7: Determine the fitness functionality f ;

Step 8: For leader selection, use a greedy method.

Step 9: end

Step 10: for all onlooker bees to choose a leader of cluster based on probability;

Step 11: end

Step 12: if the leader does not optimize; then

Step 13: produce new cluster leader;

Step 14: end

Step 15: Maintain the finest cluster leaders;

Step 16: End

3.4.3 Memetic ABC algorithm

The memetic algorithm employs the concept of meme(s) as unit of information stored in computational representation, and the targets of genetic manipulations were not any individuals in the populations' space, but some locally optimal representatives chosen via local search and from local region. The rate of convergence of the Randomized Memetic Artificial Bee Colony optimization method for WSNs may be considerably accelerated utilizing the inherent parallelism of the memetic algorithm [26–30].

Initialize the parameters.

Repeat until the termination requirements are met.

Step I: Generate new solutions for the employed bees.

Step II: Create and assess new solutions for onlooker based on the solutions chosen based on the pi.

Step III: If the scout's abandoned solution exists, replace it with a new randomly generated solution.

Step IV: Note the best solution obtained so far.

3.4.4 Randomized memetic ABC algorithm

Step 1: Begin the population of solutions

Step 2: Compute the population

Step 3: Repeat until the termination requirements are met

Step 4: Create and compute new solutions for the employed bees

Step 5: For the employed bees, use the greedy selection method

Step 6: Grade the population and quantify the fitness

Step 7: Calculate the likelihood of the solutions using fitness

Step 8: Create new solution for onlooker bee based on the solutions chosen and evaluate them

Step 9: for the scout if a discarded solution exists, change it with a new generated solution

Step 10: Keep the best solution

Step 11: Memetic search space

Step 12: Set a and b to their initial values.

Step 13: Continue while $(a-b) < \epsilon$

Step 14: Calculate f_1 and f_2

Step 15: Create two new solutions using f_1 and f_2 based on MeABC.

Step 16: Determine $f(X_{new1}) < f(X_{new2})$ for the objective functions.

Step 17: if $(f(X_{new1}) < f(X_{new2}))$ then $b=f_2$

Step 18: if $(f(X_{new1}) < f(X_{best}))$

Step 19: then $X_{best} < X_{new1}$

Step 20: else

Step 21: $a=f_1$

Step 22: if $(f(X_{new2}) < f(X_{best}))$

Step 23: then $X_{best} < X_{new2}$

According to the relevant studies, mostly all the routing protocols were developed with the goal of decreasing power consumption in order to increase WSN lifespan with minimum power consumptions and lower overhead costs. Most of the protocols presume that nodes have a general understanding of the networks. The optimization approaches' termination period, optimum parameters, and convergence period were not thoroughly explored. Scalability and fault-tolerance are not extensively explored in present optimization approaches. The majority of prior research articles utilize energy consumption, distance, network lifespan, delay, packet delivery rate as performance measures.

4.Experimental analysis

This section discusses the approach and experiments for implementing the proposed EMABC. The proposed model's performance is evaluated using the network simulation (NS-2) testbed. In addition to EMABC, the following previous techniques are utilized for performance evaluation: Enhanced

Memetic artificial bee colony (EMABC), global artificial bee colony algorithm based on the cross over and tabu search (CGTABC), Energy-efficient clustering using artificial bee colony (EC-ABC), memetic artificial bee colony algorithm (MeABC), and randomized memetic artificial bee colony algorithm (RMABC). Simulation is a cost-effective way to test the model multiple times using different situations. Table 1 represents the simulation parameters and their values of simulation parameters.

Table 1 Simulation Parameters

Parameters	Values
Simulator	NS-2.34
Simulation Period	100 ms
Coverage Area	1000 x 1000
No of Nodes	100
Initial Energy	0.5 J
E_{DA}	5 nJ/bit/signal
d_0	87 m
Packet Size	4000 bits
E_{energy}	50 nJ/bit
Free space (E_{fm})	10 p J/bit/m ²
Multi-path (E_{am})	0.00013 pJ/bit/m ⁴
Percentage of CHs	0.05

4.1Energy efficiency

The leftover energy at the end of network data transmission is referred to as energy efficiency. Figure 2 compares the energy efficiency performance of nodes for several models, including CGTABC, EC-ABC, MeABC, RMABC, and EMABC. The simulation experiment is conducted out using 100 nodes in this scenario.

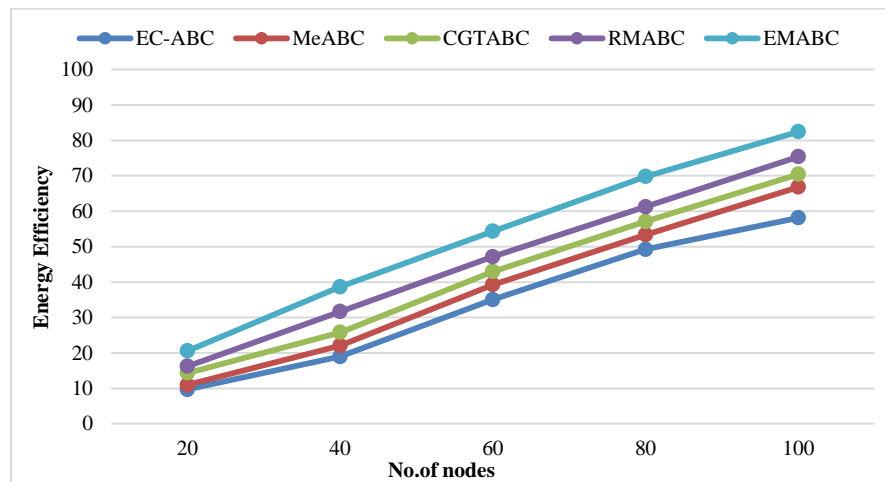


Figure 2 Graphical plot of energy efficiency comparison

The EMABC model has an energy efficiency performance of 82.44 percent. Earlier protocols, such

as EC-ABC, MeABC, CGTABC, and RMABC, yield energy efficiency of 58.12 percent, 66.83 percent, 70.42 percent, and 75.38 percent, respectively. The

results show that the EMABC model outperforms the others in terms of performance.

4.2 End-to-end delay

The entire network latency is the end-to-end delay. Figure 3 compares the end-to-end delay performance of nodes for several models, including CGTABC, EC-ABC, MeABC, RMABC, and EMABC. The

simulation experiment is conducted out using 100 nodes in this scenario.

The EMABC model has an end-to-end delay performance of 99.68 ms. EC-ABC, MeABC, CGTABC, and RMABC generate end-to-end delays of 276.21 ms, 234.53 ms, 158.64 ms, and 129.48 ms, respectively. When compared to other protocols, the result shows that the EMABC protocol provides a minimal latency.

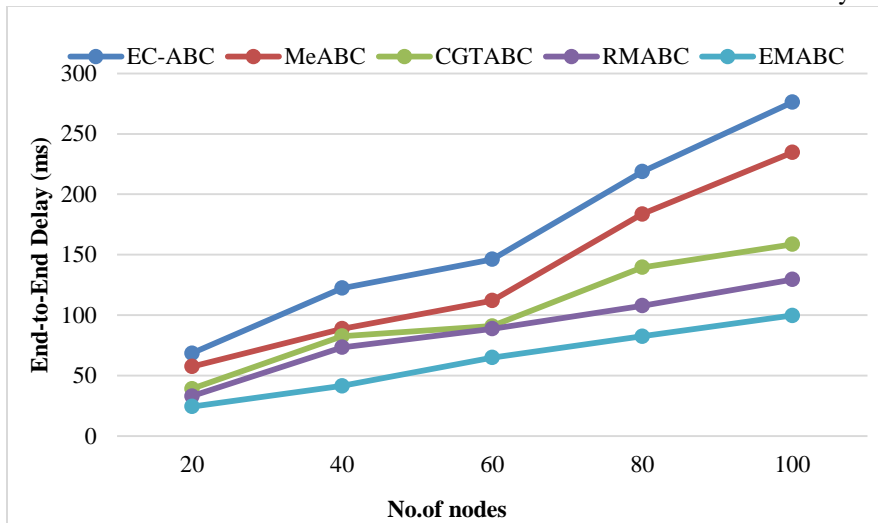


Figure 3 Graphical plot of end-to-end delay comparison

4.3 Packet drop rate

The packet drop computation is used to calculate the number of packets dropped (number of packets sent—number of packets received). Figure 4 compares the packet drop performance of the nodes for several models, including CGTABC, EC-ABC, MeABC, RMABC, and EMABC. The simulation experiment is conducted out using 100 nodes in this scenario.

The results show that the performance of the EMABC model yields fewer packet drop when compared to the others. EMABC has a packet drop performance of 152 packets. Earlier models, such as EC-ABC, MeABC, CGTABC, and RMABC, generate packet drop ratios of 638, 545, 337, and 256 packets, respectively.

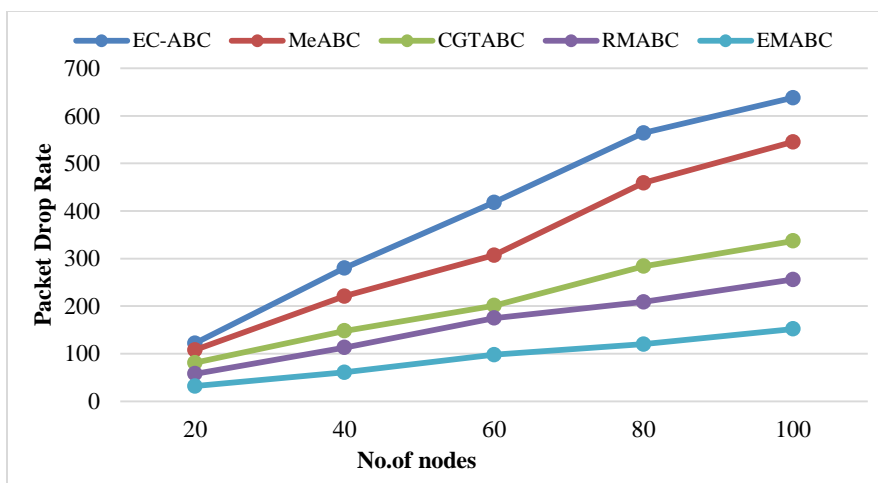


Figure 4 Graphical plot of packet drop rate comparison

4.4 Network throughput

The number of packets maximum received within a certain period is known as network throughput. *Figure 5* compares the throughput performance of the nodes for several models, including CGTABC, EC-ABC, MeABC, RMABC, and EMABC. The simulation experiment is conducted out using 100

nodes in this scenario. The EMABC model has a throughput performance of 680.28 Kbps. Earlier versions, such as the EC-ABC, MeABC, CGTABC, and RMABC, produce throughput of 192.35 Kbps, 345.73 Kbps, 483.52 Kbps, and 561.85 Kbps, respectively. The results show that the performance of the EMABC model outperforms the others.

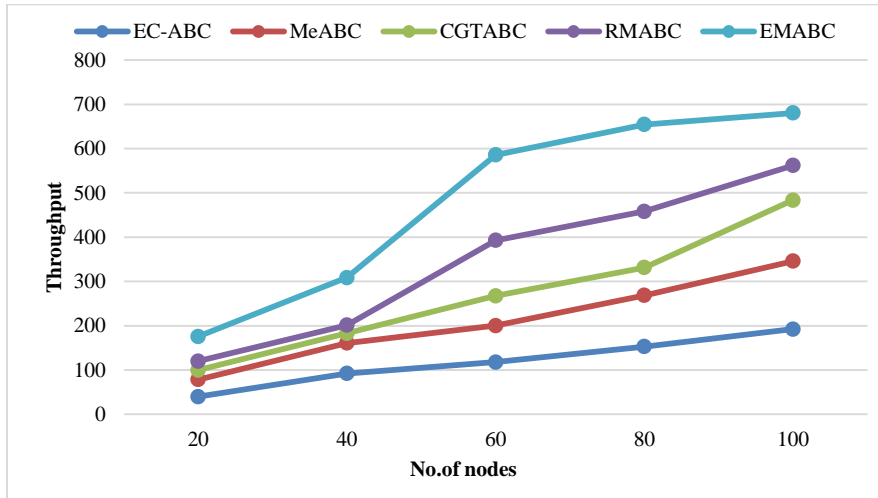


Figure 5 Graphical plot of network throughput comparison

4.5 Packet delivery ratio

The packet delivery ratio is the number of packets received by the destination divided by the number of packets sent by the sender node.

Figure 6 compares the packet delivery ratio performance of nodes for several models, including CGTABC, EC-ABC, MeABC, RMABC, and EMABC. The simulation experiment is conducted

out using 100 nodes in this scenario. The EMABC model has a packet delivery ratio of 98.05 percent. EC-ABC, MeABC, CGTABC, and RMABC models provide PDR of 77.63 percent, 84.28 percent, 89.46 percent, and 92.85 percent, respectively. The aforementioned results demonstrate that the performance of the EMABC model outperforms the others.

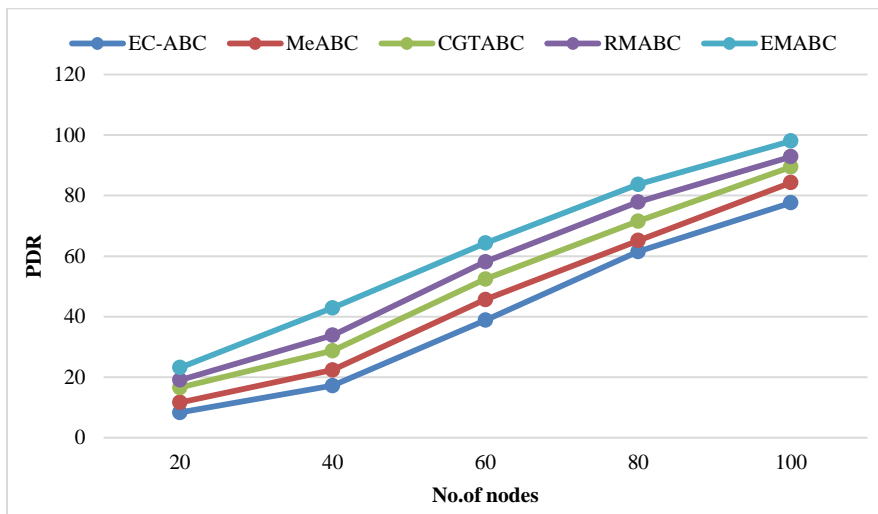


Figure 6 Graphical plot of packet delivery ratio comparison

4.6 Network lifetime

Figure 7 portrays the estimated lifetime of a network, which is computed by calculating the resource utilization at a particular time instance from source to destination from the total energy available. Figure 7 represents the network lifetime performance of nodes for several models, including CGTABC, EC-ABC,

MeABC, RMABC, and EMABC. The simulation experiment is conducted out using mobility (bps) in this scenario. The EMABC model has a network lifetime of 91 percent. EC-ABC, MeABC, CGTABC, and RMABC models provide network lifetime of 65 percent, 73.50 percent, 84 percent, and 86 percent respectively.

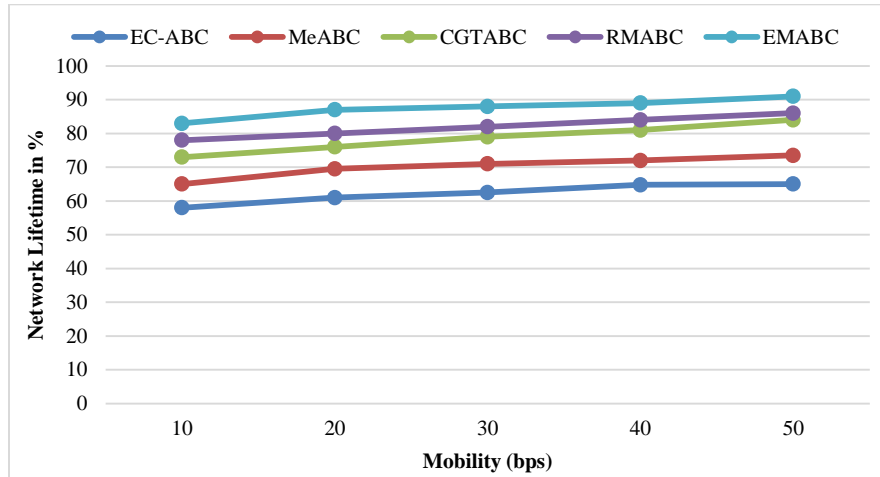


Figure 7 Graphical plot of network lifetime comparison

Based on the comparison, the EMABC model has achieved

- 82.44% Energy Efficiency
- 99.68ms End-to-End Delay
- 152 Packet drop rate
- 680.28 Kbps Network Throughput
- 98.05% Packet Delivery Ratio
- 91% Network Lifetime

The proposed model has achieved better performances in all the evaluation metrics like energy efficiency, end-to-end delay, packet drop rate, network throughput, packet delivery ratio, and network lifetime. The EMABC model has achieved an energy efficiency of 82.44%, which is 7.06% to 24.3% improved than other compared models. The end-to-end delay of the EMABC model was 99.68ms, which is 29.8ms to 176.53ms better than the compared models. The packet drop rate of EMABC model was 152 packets, which is 10.4% to 48.6% lesser than the compared models. The network throughput was 680.28Kbps, which is 11.8% to 48.7% higher than the other compared models. The packet delivery rate was 98.05%, which is 5.2% to 20.4% better than the other compared models. Finally, the network lifetime was 91%, which is 5% to 26% improved than the other compared models. The main limitation of the proposed research is the

optimization and the computational time. To overcome this issue, optimization algorithms with better performance can be implemented to solve this issue. A combination of two optimization algorithms can be a better choice to solve this issue and to achieve better performances. A complete list of abbreviations is shown in *Appendix I*.

5. Conclusion

Different energy-efficient clustering (EEC) methods based on ABC optimization were applied for the WSN in this comparative research to increase the network's energy efficiency. The energy-efficient clustering method consisted of forming the appropriate number of clusters and selecting CHs in a dynamic manner. Furthermore, relay nodes were chosen with energy efficiency objective, and they were prioritized afterwards. The implemented methods perform energy-efficient clustering. To increase energy efficiency during the communication process, the enhanced memetic artificial bee colony (EMABC), global artificial bee colony algorithm based on the cross over and tabu search (CGTABC), Energy-efficient clustering using artificial bee colony (EC-ABC), memetic artificial bee colony algorithm (MeABC), and randomized memetic artificial bee colony algorithm (RMABC) were implemented. In terms of assessment and validation, the implemented

models were compared. As a consequence, the proposed EMABC model performed well in terms of energy efficiency, with an efficiency of 82.44 percent, an end-to-end delay of 99.68ms, a packet drop rate of 152, a throughput of 680.28Kbps, packet delivery ratio of 98.05 percent, and network lifetime of 91%. Here in this analysis, based on the comparison of research, the main limitation in most of techniques include energy efficiency and network lifetime. This study can be expanded to mobile networks in the future based on the energy-efficient routing algorithms for WSNs.

Acknowledgment

The authors would like to thank the Department Of Electronics & Telecommunication Engineering, Sir M Visveswaraya Institute of Technology, Bangalore, Karnataka, India and Sambhram Institute of Technology, Bangalore, Karnataka, India for their valuable research guidance and support throughout the completion of this project.

Conflicts of interest

The authors have no conflicts of interest to declare.

Authors contribution statement

S. Sowndeswari: Responsible for survey, algorithm design and article writing. **E. Kavitha:** Responsible for algorithm design, verification and proof reading.

References

- [1] Soua R, Minet P. A survey on energy efficient techniques in wireless sensor networks. In joint IFIP wireless and mobile networking conference 2011 (pp. 1-9). IEEE.
- [2] Al AZ, Khedr AM, Osamy W, Arif I, Agrawal DP. Routing in wireless sensor networks using optimization techniques: a survey. *Wireless Personal Communications*. 2020; 111:2407-34.
- [3] Singh S, Sharma RM. Optimization techniques in wireless sensor networks. In proceedings of the second international conference on information and communication technology for competitive strategies 2016 (pp. 1-7).
- [4] Wang Z, Ding H, Li B, Bao L, Yang Z. An energy efficient routing protocol based on improved artificial bee colony algorithm for wireless sensor networks. *IEEE Access*. 2020; 8:133577-96.
- [5] Pitchaimanickam B, Murugaboopathi G. A hybrid firefly algorithm with particle swarm optimization for energy efficient optimal cluster head selection in wireless sensor networks. *Neural Computing and Applications*. 2020; 32(12):7709-23.
- [6] Aroba OJ, Naicker N, Adeliyi T. An innovative hyperheuristic, gaussian clustering scheme for energy-efficient optimization in wireless sensor networks. *Journal of Sensors*. 2021.
- [7] Sekaran K, Rajakumar R, Dinesh K, Rajkumar Y, Latchoumi TP, Kadry S, et al. An energy-efficient cluster head selection in wireless sensor network using grey wolf optimization algorithm. *TELKOMNIKA*. 2020; 18(6):2822-33.
- [8] Ahmad T. Energy EC: an artificial bee colony optimization based energy efficient cluster leader selection for wireless sensor networks. *Journal of Information and Optimization Sciences*. 2020; 41(2):587-97.
- [9] Sridhar R, Guruprasad N. Energy efficient chaotic whale optimization technique for data gathering in wireless sensor network. *International Journal of Electrical and Computer Engineering*. 2020; 10(4):4176-88.
- [10] Maheshwari P, Sharma AK, Verma K. Energy efficient cluster based routing protocol for WSN using butterfly optimization algorithm and ant colony optimization. *Ad Hoc Networks*. 2021.
- [11] Bhola J, Soni S, Cheema GK. Genetic algorithm based optimized leach protocol for energy efficient wireless sensor networks. *Journal of Ambient Intelligence and Humanized Computing*. 2020; 11(3):1281-8.
- [12] Chen Z, Li S, Yue W. Memetic algorithm-based multi-objective coverage optimization for wireless sensor networks. *Sensors*. 2014; 14(11):20500-18.
- [13] Ajmi N, Helali A, Lorenz P, Mghaieth R. MWCSGA—multi weight chicken swarm based genetic algorithm for energy efficient clustered wireless sensor network. *Sensors*. 2021; 21(3):1-21.
- [14] Wang H, Chen Y, Dong S. Research on efficient-efficient routing protocol for WSNs based on improved artificial bee colony algorithm. *IET Wireless Sensor Systems*. 2017; 7(1):15-20.
- [15] Alghamdi TA. Secure and energy efficient path optimization technique in wireless sensor networks using DH method. *IEEE Access*. 2018; 6:53576-82.
- [16] Lin D, Wang Q. An energy-efficient clustering algorithm combined game theory and dual-cluster-head mechanism for WSNs. *IEEE Access*. 2019; 7:49894-905.
- [17] Hassan AA, Shah WM, Habeb AH, Othman MF, Al-mhiquani MN. An improved energy-efficient clustering protocol to prolong the lifetime of the WSN-based IoT. *IEEE Access*. 2020; 8:200500-17.
- [18] Ogundile OO, Balogun MB, Ijiga OE, Falayi EO. Energy-balanced and energy-efficient clustering routing protocol for wireless sensor networks. *IET Communications*. 2019; 13(10):1449-57.
- [19] Sharma R, Vashisht V, Singh U. EEFCM-DE: energy-efficient clustering based on fuzzy C means and differential evolution algorithm in WSNs. *IET Communications*. 2019; 13(8):996-1007.
- [20] Reddy DL, Puttamadappa C, Suresh HN. Merged glowworm swarm with ant colony optimization for energy efficient clustering and routing in wireless sensor network. *Pervasive and Mobile Computing*. 2021.
- [21] Sheriba ST, Rajesh DH. Energy-efficient clustering protocol for WSN based on improved black widow

optimization and fuzzy logic. Telecommunication Systems. 2021; 77(1):213-30.

- [22] Nalluri PR, Gnanadhas JB. A cognitive knowledged energy-efficient path selection using centroid and ant-colony optimized hybrid protocol for WSN-assisted IoT. Wireless Personal Communications. 2022:1-27.
- [23] Sun Y, Dong W, Chen Y. An improved routing algorithm based on ant colony optimization in wireless sensor networks. IEEE Communications Letters. 2017; 21(6):1317-20.
- [24] Kumar S, Sharma VK, Kumari R. Randomized memetic artificial bee colony algorithm. arXiv preprint arXiv:1408.0102. 2014.
- [25] Chandran R, Kumar SR, Gayathri N. Genetic algorithm-based tabu search for optimal energy-aware allocation of data center resources. Soft Computing. 2020; 24(21):16705-18.
- [26] Nallamuthu SA. A hybrid genetic-neuro algorithm for cloud intrusion detection system. Journal of Computational Science and Intelligent Technologies. 2020; 1(2):15-25.
- [27] Ravi M. A survey on security risks in internet of things (IoT) environment. Journal of Computational Science and Intelligent Technologies. 2020; 1(2):1-8.
- [28] Choubey R, Dubey R, Bhattacharjee J. A survey on cloud computing security, challenges and threats. International Journal on Computer Science and Engineering. 2011; 3(3):1227-31.
- [29] Narmatha C. A new neural network-based intrusion detection system for detecting malicious nodes in WSNs. Journal of Computational Science and Intelligent Technologies. 2020; 1(3):1-8.
- [30] Sasirekha SP, Priya A, Anita T, Sherubha P. Data processing and management in IoT and wireless sensor network. In journal of physics: conference series 2020. IOP Publishing.



S. Sowndeswari obtained her M.Tech., in Digital Communication & Networking from VTU and topped university rank in VTU. Currently pursuing Ph.d., from VTU. Presently working as an Assistant Professor in Sambhram Institute of Technology, Bengaluru. She has over 13 years of

teaching experience in the field of Electronics and Communication Engineering. She has a large number of publications to her credit and also presented several papers in International and National Conferences. NPTEL online certified in Principles of Communication Systems – I (IIT-Karagpur) and Signals & Systems, Part – 1 from IIT Bombay. Interested Fields are Antennas & Communication, Image Processing, Network Security, Wireless Sensor Networks.

Email: sowndeswari@yahoo.co.in



Dr. E. Kavitha obtained her Ph.D., from Annamalai University in the field of Communication systems. Presently working as a professor and Head of the department in the department of Electronics and Telecommunication Engineering in Sir M VIT, Bengaluru. She has over 20 years of teaching experience. She has a large number of publications to her credit and presented several papers in international conferences. Her areas of interests are Wireless Sensor Networks, Communication Systems, Wireless Communications and Networking.
Email: kavimail3@gmail.com

Appendix 1

S. No.	Abbreviation	Description
1	ABC	Artificial Bee Colony Optimization
2	ACO	Ant Colony Optimization
3	ACOIGSOA	Ant Colony Optimization Integrated Glowworm Swarm Optimization Approach
4	AODV	Ad-hoc On-demand Distance Vector
5	BISI	Bio-Inspired Swarm Intelligence
6	BMBWFL-HEED	Boosted Mutation-based Black Widow Optimization with Fuzzy Logic-Based HEED Protocol
10	BMBWO	Boosted Mutation-based Black Widow Optimization
11	BS	Base Station
12	CGTABC	Global Artificial Bee Colony Algorithm Based on the Cross Over and Tabu Search
13	CH	Cluster Head
14	C-LEACH	Centralized-Low-Energy Adaptive Clustering Hierarchy
15	CWMEODG	Chaotic Whale Metaheuristic Energy-Optimized Data Gathering
16	DA	Dijkstra's Algorithm
17	DC	Direct Current
18	DHM	Dij-Huff Method
19	EB	Energy-Balanced
20	EC-ABC	Energy-Efficient Clustering Using Artificial Bee Colony
21	ECGD	Dual-Cluster-Head
22	EE	Energy-Efficient
23	EEC	Energy-Efficient Clustering
24	EECAO	Energy Efficient Centroid-based Ant colony Optimization
25	EEFCM-DE	Energy-Efficient Clustering Based On Fuzzy C Means And Differential Evolution Method
26	EMABC	Enhanced Memetic Artificial Bee Colony
27	GSO	Glowworm Swarm Optimization
28	HC	Huffman Coding
29	HEED	Hybrid Energy Efficient Distributed Clustering
30	IEECP	Improved Energy-Efficient Clustering Protocol
31	IoT	Internet of Things
32	LEACH	Low-Energy Adaptive Clustering Hierarchy

33	MeABC	Memetic Artificial Bee Colony Algorithm
34	MOCADMA	Multi-Objective Coverage Optimizations Method
35	NEP	Nash Equilibrium Point
36	NP-hard	Non-Deterministic Polynomial-Time Hard
37	O-LEACH	Optimization Method Genetic Algorithm
38	QoS	Quality of Service
39	RMABC	Randomized Memetic Artificial Bee Colony Algorithm
40	TBC	Tree-Based Clustering
41	WSN	Wireless Sensor Networks