International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2023/223428 Volume 10, Issue 5, September – October (2023)

RESEARCH ARTICLE

Performance Enhancement of Mobility-Enabled Wireless Sensor Network Using Sophisticated Eagle Search Optimization-Based Gaussian Ad Hoc On-Demand Distance Vector (SESO-GAODV) Routing Protocol

V. Veerakumaran
Department of Computer Science, Nallamuthu Gounder Mahalingam College, Pollachi, Tamil Nadu, India.
veerakumaranvvk@gmail.com

Aruchamy Rajini
Department of Computer Science, Nallamuthu Gounder Mahalingam College, Pollachi, Tamil Nadu, India.
saruchamyrajini@gmail.com

Received: 25 July 2023 / Revised: 16 September 2023 / Accepted: 23 September 2023 / Published: 30 October 2023

Abstract – The research focuses on enhancing the performance of Mobility Enabled Wireless Sensor Networks (ME-WSNs) through the introduction of a novel routing protocol named Sophisticated Eagle Search Optimization-Based Gaussian Ad Hoc On-demand Distance Vector (SESO-GAODV). ME-WSNs pose unique challenges due to their dynamic and rapidly changing network topologies. To address these challenges, SESO-GAODV leverages the intelligent optimization techniques of Sophisticated Eagle Search Optimization and the dynamic route discovery capabilities of Gaussian Ad Hoc On-demand Distance Vector (GAODV). The proposed protocol undergoes extensive evaluations and comparisons with other existing routing protocols. Through comprehensive performance analysis, SESO-GAODV demonstrates superior results, including reduced delay, increased throughput, minimized packet loss, and lower energy consumption. The protocol's adaptability to changing network conditions and efficient handling of node mobility contribute to its energy-efficient nature, making it a promising solution for enhancing data transmission efficiency and reliability in ME-WSNs. SESO-GAODV's ability to optimize energy consumption ensures a prolonged network lifetime, facilitating seamless communication and optimized network performance in dynamic and challenging environments.

Index Terms – AODV, Eagle Search Optimization, Gaussian, ME-WSNs, Routing, Sensor Network.

1. INTRODUCTION

Wireless Sensor Networks (WSNs) have emerged as a powerful technology in various fields, including environmental monitoring, industrial automation, healthcare, and smart cities. These networks consist of small, autonomous devices called wireless sensor nodes that communicate with each other and the base station using wireless communication protocols. Traditional WSNs often face limited coverage, connectivity, and energy consumption challenges [1]. To address these issues, the Mobility-Enabled Wireless Sensor Networks (ME-WSNs) concept has gained attention. ME-WSNs incorporate mobility into the sensor nodes, allowing them to move within the network’s operating area [2]. This mobility introduces a dynamic element to the network, enabling better coverage, connectivity, and data collection. The nodes can be deployed on mobile vehicles, robots, or animals. By leveraging the mobility of these entities, ME-WSNs offer several advantages over traditional WSNs [3].

One significant advantage of ME-WSNs is enhanced network coverage. In traditional WSNs, fixed sensor nodes may struggle to cover large areas, leading to coverage gaps or insufficient data collection. By introducing mobility, ME-WSNs can adapt to the changing environment and ensure better coverage [4]. The nodes can move to areas where data collection is required or follow the movement of an object of interest. Connectivity is another crucial aspect addressed by ME-WSNs.

Traditional WSNs often suffer from connectivity issues due to obstacles or node failures. In ME-WSNs, mobile nodes can act as intermediaries, relaying data between disconnected nodes and maintaining a connected network [5]. When a node moves out of range, a mobile node can bridge the communication gap by moving closer to both nodes, enabling seamless data transfer. This dynamic connectivity ensures reliable and efficient data transmission, enhancing the network's performance [6].

Energy efficiency is a critical consideration in WSNs due to the limited power resources of sensor nodes. ME-WSNs can optimize energy consumption by strategically deploying mobile nodes. For instance, a mobile node can move closer to energy-depleted nodes to collect their data, reducing the need for frequent recharge or replacement [7]. ME-WSNs can employ energy-aware routing protocols that leverage the mobility patterns of nodes to minimize energy consumption while maintaining connectivity. By intelligently managing energy resources, ME-WSNs can extend the network's lifetime and reduce maintenance efforts. ME-WSNs enable targeted data collection. Due to their static nature, traditional WSNs often generate a vast amount of redundant or irrelevant data [8]. With mobility, ME-WSNs can focus on specific areas or objects of interest, collecting data where it is most valuable. For example, in disaster response scenarios, mobile nodes can be deployed to gather data from affected regions or monitor the movements of rescue teams. This targeted data collection enhances the network's efficiency and reduces the burden of processing and storing irrelevant information [9].

Using bio-inspired optimization in routing for Wireless Sensor Networks (WSNs) brings several positive benefits. Firstly, it significantly improves energy efficiency by finding optimal paths for data transmission, leading to extended network lifetime and reduced maintenance costs. Secondly, these techniques enhance network reliability by adapting to dynamic environmental changes and network disruptions [10]–[12]. They provide robust and scalable solutions for efficient data delivery in large-scale WSN deployments. Bio-inspired optimization enables WSNs to perform better in data throughput, latency, and packet delivery ratio, making them ideal for real-world applications such as environmental monitoring, disaster management, and precision agriculture. Figure 1 provides the overall process of bio-inspired optimization in routing.

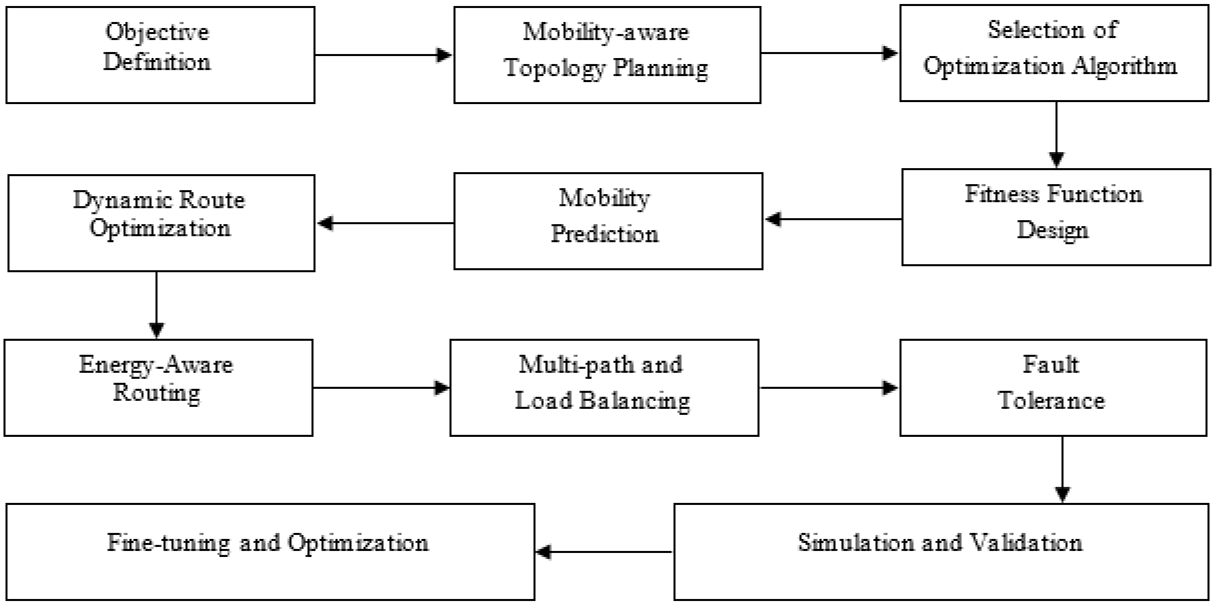
Objective Definition
Mobility-aware Topology Planning
Selection of Optimization Algorithm
Fitness Function Design
Mobility Prediction
Dynamic Route Optimization
Energy-Aware Routing
Multi-path and Load Balancing
Fault Tolerance
Simulation and Validation
Fine-tuning and Optimization

Figure 1 Overall Bio-Inspired Optimization Routing in ME-WSN

Despite the numerous benefits, ME-WSNs also present challenges. Some complex tasks involve designing efficient mobility patterns, developing robust localization algorithms, and managing node mobility [13]. Ensuring secure and reliable communication between mobile nodes and the base station becomes crucial to prevent data loss or unauthorized access [14]. Mobility-Enabled Wireless Sensor Networks offer significant advantages over traditional WSNs by introducing mobility into the network. Enhanced coverage, improved connectivity, energy efficiency, and targeted data collection are some of the critical benefits of ME-WSNs. By leveraging the mobility of sensor nodes, these networks can adapt to dynamic environments and overcome the limitations of static networks. While challenges exist, ME-WSNs hold great potential [15].

1.1. Problem Statement

The dynamic nature of Mobility-Enabled Wireless Sensor Networks (ME-WSNs) presents a significant challenge in ensuring application Quality of Service (QoS) requirements. With nodes constantly moving, maintaining reliable and efficient data transmission becomes complex, especially while meeting diverse QoS metrics such as low latency, high reliability, and real-time delivery. The intermittent connectivity disruptions and varying link qualities caused by node mobility further complicate providing consistent and predictable QoS. Additionally, the limited power resources of sensor nodes add an energy efficiency dimension to the problem, requiring routing protocols that conserve energy while still meeting QoS requirements. Moreover, scalability becomes a concern in large-scale ME-WSNs, necessitating routing solutions that can handle the increasing complexity of network structures while maintaining QoS performance.

1.2. Motivation

The increasing demand for reliable and efficient data transmission in dynamic environments stems from the motivation to address the challenges of ensuring Quality of Service (QoS) in Mobility-Enabled Wireless Sensor Networks (ME-WSNs). ME-WSNs find applications in diverse domains such as healthcare, environmental monitoring, industrial automation, and smart cities, where real-time and accurate data delivery is crucial. Enabling QoS in ME-WSNs is essential for applications that rely on specific performance guarantees, such as timely patient monitoring in healthcare or real-time control and monitoring in industrial automation. The dynamic nature of ME-WSNs, characterized by node mobility and intermittent connectivity disruptions, necessitates the development of QoS-aware routing protocols that can adapt to changing network conditions while conserving energy and maintaining scalable performance. By addressing the challenges associated with QoS in ME-WSNs, we can unlock the full potential of these networks, enabling timely and accurate data delivery, enhancing system performance, and empowering various sectors with reliable and efficient wireless sensor network deployments.

1.3. Objective

The objective of addressing the Quality of Service (QoS) challenges in Mobility-Enabled Wireless Sensor Networks (ME-WSNs) is to develop efficient and adaptive routing strategies and protocols that ensure reliable and efficient data transmission in dynamic environments. The objective includes:

- Designing QoS-aware routing algorithms: Develop innovative routing algorithms that can adapt to the dynamic nature of ME-WSNs, considering node mobility, intermittent connectivity disruptions, and varying link qualities. These algorithms should optimize the selection of routing paths based on QoS requirements such as low latency, high reliability, and real-time data delivery.

- Energy-efficient routing: Develop energy-aware routing strategies that conserve the limited power resources of sensor nodes while still meeting the desired QoS metrics. This involves optimizing routing decisions to minimize energy consumption, prolong the network lifetime, and reduce maintenance efforts.

- Scalable routing solutions: Design scalable routing protocols capable of handling large-scale ME-WSNs while maintaining QoS performance. The objective is to develop routing techniques that efficiently manage the increasing complexity and number of mobile nodes within the network.

By achieving these objectives, ME-WSNs can provide reliable and efficient data transmission, enabling applications that rely on specific QoS requirements. Developing QoS-aware routing solutions in ME-WSNs improves system performance, energy efficiency, and scalability, ultimately enhancing the overall effectiveness and usability of wireless sensor network deployments.

1.4. Organization of the Paper

In section 1, this paper discussed ME-WSN, Routing, and Bio-inspired Optimization along with the problem statement, highlighting the challenges faced in mobility-enabled wireless sensor networks, followed by the motivation behind addressing these challenges and the specific objectives of the study. Section 2, "Literature Review," provides a comprehensive overview of existing research in the field. Section 3 delves into the novel routing protocol proposed in this paper, offering an in-depth explanation of its design and functioning. In Section 4, "Simulation Settings and Performance Metrics", this paper details the experimental setup and the metrics used to evaluate the proposed protocol. Section 5, "Results and Discussion," presents the empirical findings and provides an extensive analysis, allowing for a better understanding of the protocol's performance enhancements. Lastly, the paper concludes in Section 6, summarizing the key findings and their implications and suggestions for future research directions in this domain.

2. LITERATURE REVIEW

“Fuzzy Logic Scheme” [16] is proposed for clustering the network using various factors in WSN. Experiments were conducted to prove its efficiency in energy usage and network lifetime with calculated accuracy. The node density and distance among base stations and nodes are measured for selecting the cluster. “Node Stability-based Routing” [17] is proposed to define the node’s stability and select the proper gateway. The entropy function defines the node, and the simulation output is retrieved through experimental study. The performance is compared with other techniques regarding Expected Transmission count, channel switching, and Reinforcement learning-based best path to the best gateway. **“**Reverse Glow-Worm Swarm Optimization” [18] is proposed for better energy consumption in the sensor network. The efficiency of the sensor network is also implemented based on the sensor movement and is positioned using the grid points. Simulation-based results are generated to prove its outperformance over existing algorithms. **“**Butterfly

Optimization Algorithm" [19] is proposed for selecting the cluster head for a group of nodes. The energy of the residual node and distance to the base station is optimized, and the routes are detected using the Ant Colony Optimization algorithm. Performance metrics are used to measure the efficiency of the proposed technique and output compared with baseline algorithms. "Multimodal Wireless Sensor Networks" [20] is proposed to track applications with low bandwidth using IoT. With the incorporation of an indoor-based testbed, the performance was evaluated, and experiments were carried out. The data and energy transmission were efficiently saved using the broadcast-based wake-up structure portrayed using an experimental study.

"Depth Based Routing (DBR)" [21] is proposed for handling acoustic communication in Underwater Wireless Sensor Networks. The routing protocol retrieves the performance-based indices and measures the transmission rate. Optimization is carried out for trade-offs among energy usage, end-to-end delay, and delivery rate. "Multi-Radio Multi-Channel Optimization" [22] is proposed for allocating the resources, power control, and channel distribution in WSN. Two-stage allocation of resources is performed to detect the dependency of different resources. A graph colouring algorithm is used for assigning the time slot, and a multi-objective optimization issue is formulated to enhance the network's efficiency. A multi-objective hybrid particle swarm optimization algorithm retrieves the optimal solutions through simulation output. "Application-Specific Routing Protocol" [23] is proposed for designing the model for managing health care applications. The performance increases in packet delivery ratio, packet loss, network lifetime, and end-to-end delay. The network's lifetime is also enhanced by maximizing the throughput in data transmission. Comparison is done using the baseline techniques to prove its efficiency over traditional techniques. "Heat-Diffusion Collection Protocol" [24] is proposed for WSN to guarantee routing techniques in WSN. Evaluation is done by comparing the analysis using Collection Tree Protocol and Backpressure Collection Protocol. Findings are generated by comparing the results with traditional algorithms to prove their performance over other techniques. "Improved Gene Algorithm" [25] is proposed to position passengers at the airport in WSN. Optimization is performed for validating the passengers with data as input. Results are generated for accurate positioning of the flow of passengers, and performance is measured with the comparison of traditional algorithms.

"Packet Forwarding" [26] is proposed for modelling the reporting event in the queuing network. The events are measured using jitter, and the minimum value is recorded. The queuing network is prioritized, and privacy is preserved, demonstrated using a simulator that portrays the temporal information about the traffic to the node. "Routing protocol using virtual infrastructure" [27] is proposed for solving the routing problem in sensor networks. The position of the sink node is updated and saved by creating the routing among clusters. Extensive experimental analysis is conducted using a simulator, and results are generated to prove its performance. "Video Encoding" [28] is proposed for optimizing the data generated for routing in Wireless Multimedia Sensor Networks. The network's lifetime is maximized, and the shortest path routing protocol is used to maximize the network's lifetime. Evaluation is carried out to prove its efficiency over other techniques, and simulation results are portrayed. **"**Clustered QoS Routing Protocol" [29] is proposed for handling the issue in wireless networks in which the nodes are grouped, and the cluster head is assigned with the link to avoid the issue of link failures. The cluster members communicate using the cluster head, and experimental results are generated to prove its efficiency. **"**Energy-Efficient and Reliable Routing Scheme" [30] is proposed for enhancing the reliability and stability of constrained Wireless Body Area Networks (WBAN). The adaptive static clustering method enhances the network lifetime, reliability, and stability through experimental evaluation.

"Cluster Sub-graph Selection based Routing (CSSR)" [31] has been developed as a routing protocol tailored for wireless Ad Hoc networks, focusing on facilitating cluster-based communication. CSSR divides the network into clusters, each assigning a specific cluster head responsible for inter-cluster communication. This approach has resulted in reduced overhead of routing updates and improved scalability of the network. By leveraging the concept of sub-graph selection, CSSR optimizes routing paths, leading to decreased transmission delays and efficient data delivery. "Energy-efficient Adaptive cum Cooperative Routing (EEACR)" [32] has been specifically designed to balance energy consumption and extend network lifetime in wireless sensor networks. EEACR employs adaptive techniques to adjust transmission power levels based on node proximity, thereby conserving energy and extending the operational time of individual nodes. Moreover, EEACR promotes cooperative communication among closely situated nodes, reducing the necessity for long-distance transmissions and enhancing energy efficiency. The adaptive and cooperative mechanisms of the protocol have contributed to the efficient utilization of energy, rendering it suitable for resource-constrained sensor networks.

In the domain of ME-WSN, a significant research gap exists in developing routing protocols that can adeptly handle the dynamic mobility of sensor nodes, ensuring efficient data transmission while optimizing energy usage. Current routing strategies often struggle to adapt to the rapid node movements within these networks. Additionally, scalability poses a substantial challenge as ME-WSNs continue to expand in size and complexity, necessitating the creation of scalable routing solutions capable of accommodating large-scale networks while maintaining reliability. Moreover, the absence of standardized evaluation metrics and benchmark datasets for ME-WSN [33] routing protocols hampers fair performance comparisons and inhibits the establishment of best practices. Addressing these research gaps is crucial to enhancing the capabilities and practicality of ME-WSNs in various real-world applications.

3. SOPHISTICATED EAGLE SEARCH OPTIMIZATION-BASED GAUSSIAN AD HOC ON-DEMAND DISTANCE VECTOR (SESO-GAODV)

3.1. Gaussian Ad Hoc On-demand Distance Vector (GAODV)

3.1.1. Gaussian-Based Distance Estimation

The predicted travel time from one node to another, denoted as D_{est}, can be represented mathematically. Eq.(1) models the probability distribution of D_{est} using a Gaussian function.

$$P(D_{est}) = (1/sqrt(2 * pi * sigma^2)) * exp\big(-(D_{est} - mu)\big)^2/(2 * sigma^2)) \quad (1)$$

Here, mu represents the mean, and $sigma^2$ represents the variance of the distribution.

3.1.2. Selective Flooding with Gaussian Probabilities

Nodes selectively flood Route Request (RREQ) packets based on the estimated distance. Each node calculates the probability of forwarding the RREQ packet, denoted as $P_{forward}$. The probability can be derived from the Gaussian-based distance estimation, and Eq.(2) represents the same.

$$P_{forward} = P(D_{est}) \quad (2)$$

Nodes closer to the estimated distance have a higher probability of forwarding the packet, while nodes farther away have a lower probability.

3.1.3. RREQ Processing at Intermediate Nodes

Intermediate nodes receiving an RREQ packet evaluate available routes based on link quality, residual energy, or latency metrics. Gaussian functions can be used to calculate probabilities for each metric, such as:

3.1.3.1. Link Quality Probability (P_{link})

The Link Quality Probability is denoted as P_{link}, is a metric representing the likelihood of a reliable and high-quality link between nodes. It is calculated using a Gaussian function based on the link quality metric. The link quality metric can incorporate signal strength, packet error rate, and signal-to-noise ratio. The Gaussian function assigns a probability value to the link quality metric, indicating the probability that the link offers a good communication channel. A higher P_{link} signifies a higher probability of having a robust and reliable link. P_{link} is calculated using Eq.(3).

$$P_{link} = Gaussian(link\ quality) \quad (3)$$

3.1.3.2.. Residual Energy Probability (P_{energy})

The Residual Energy Probability is denoted as P_{energy}, is a metric that reflects the remaining energy in a node's power source (e.g., battery). It is calculated using a Gaussian function based on the residual energy metric. The residual energy metric represents a node's power source's remaining energy level or capacity. By applying a Gaussian function to the residual energy metric, P_{energy} is determined, indicating the probability of having sufficient energy reserves in a node. A higher P_{energy} implies a higher probability of having nodes with ample energy to support communication and routing operations. P_{energy} is calculated using Eq.(4).

$$P_{energy} = Gaussian(residual\ energy) \quad (4)$$

3.1.3.3. Latency Probability ($P_{latency}$)

The Latency Probability, denoted as $P_{latency}$, is a metric representing the expected delay or latency in transmitting packets across the network. It is calculated using a Gaussian function based on the latency metric. The latency metric can encompass transmission, queuing, and processing delays. By applying a Gaussian function to the latency metric, $P_{latency}$ is determined, indicating the probability of experiencing lower latency for packet transmission. A higher $P_{latency}$ suggests a higher probability of achieving lower delays and faster communication across the network. $P_{latency}$ is calculated using Eq.(5).

$$P_{latency} = Gaussian(latency) \quad (5)$$

The intermediate node chooses the path with the highest overall probability, considering the metrics evaluated using Gaussian probabilities.

3.1.4. RREP Propagation and Route Maintenance

When the RREQ reaches its final resting place or an intermediate node that knows how to reach the final resting place, the Route Reply (RREP) is returned to the sending node. The selected route is established and maintained for future data transmission. While no specific equations exist in this step, the Gaussian-based probabilities calculated in the previous step can determine the preferred route during the RREP propagation and subsequent route maintenance process.

3.1.5. Dynamic Adjustment of Gaussian Parameters

To adapt to the changing network conditions, the parameters of the Gaussian functions, such as mean (mu) and variance ($sigma^2$), can be dynamically adjusted. This dynamic adjustment allows the routing protocol to adapt to variations in distance, link quality, energy levels, and latency, ensuring efficient route discovery and selection. Although no specific equations are mentioned in this step, the dynamic adjustment process involves updating the values of mu and $sigma^2$ based on real-time network measurements and feedback. Algorithm 1 provides the pseudocode of Gaussian-based AODV Routing.

Step 1: Initialization

Initialize network parameters, including the mean (mu) and variance ($sigma^2$) for Gaussian distance estimation and other relevant metrics.

Step 2: Route Request (RREQ) Broadcasting

When a node wants to initiate route discovery, it broadcasts an RREQ packet.

Step 3: Gaussian-based Distance Estimation

Upon receiving the RREQ packet, each node estimates its distance to the destination using a Gaussian-based approach by calculating the estimated distance, D_{est}, based on a probabilistic model considering the mean and variance.

Step 4: Selective Flooding with Gaussian Probabilities

Determine the probability of forwarding the RREQ packet based on the estimated distance. Assign a probability, $P_{forward}$, to the RREQ packet, enabling selective flooding where nodes closer to the estimated distance have a higher probability of forwarding the packet.

Step 5: RREQ Processing at Intermediate Nodes

Using a probabilistic approach, intermediate nodes receiving the RREQ packet evaluate available routes based on metrics like link quality, residual energy, or latency.

Step 6: Gaussian-based Route Selection

Select the route with the highest overall probability, considering the evaluated metrics. Choose the route that maximizes the combined probabilities of link quality, residual energy, and latency.

Step 7: Route Reply (RREP) Propagation and Maintenance

An RREP is sent back to the sending node after the RREQ reaches its final stopping point or an intermediate node that knows how to reach the final node, establishing and maintaining the selected route for future data transmission.

Step 8: Dynamic Adjustment of Gaussian Parameters

Periodically monitor the network conditions and adapt the parameters (such as mean and variance) used in the probabilistic models, allowing the routing protocol to adapt to variations in distance, link quality, energy levels, and latency.

Step 9: Repeat Route Discovery Process

Repeat the process as needed for subsequent route discovery requests.

Algorithm 1 Gaussian-Based AODV Routing

3.2. Sophisticated Eagle Search Optimization (SESO) for Routing

3.2.1. Eagle Search Optimization (ESO)

Eagle Search Optimization (ESO) is a search algorithm that draws inspiration from the hunting techniques and cooperative intelligence observed in real eagles. It encompasses three distinct predation phases, namely the choosing phase, the search phase, and the swooping phase. Each phase contributes to the overall efficiency and effectiveness of the algorithm in various domains, including routing in WSN.

3.2.1.1. Choosing Phase

In the choosing phase (CHP), ESO aims to identify promising solutions within the search space based on their fitness values. The search space is typically represented as a set of candidate solutions, each encoded as a vector or a set of variables. The fitness function evaluates the quality or objective value of each solution. This function captures the problem-specific criteria and constraints that need to be optimized. ESO employs various selection strategies, such as tournament or roulette wheel selection, to choose the most promising solutions for further exploration. These strategies are based on fitness values, where solutions with higher fitness values are more likely to be selected. Algorithm 2 provides the pseudocode of CHP.

In CHP, eagles initially choose the search region before zeroing in on the optimal predation spot based on the density and distribution of potential meals. As Eq.(6) implies, eagles will hunt in this region.

$$M_{s,new} = M_{best} + \delta * b(M_{mean} - M_s), \tag{6}$$

Wherein δ is a position-changing parameter with a range of 1.2 to 2.3 and a random integer b between (0, 1), and M_{best} indicates the best place found during the last search for eagles to choose an area. The eagle distribution average after the last search is located at M_{mean}. The coordinates for the sth pair of eagles are M_s. The eagles have moved to $M_{s,new}$.

Input:

- Search space: A set of candidate solutions
- Fitness function: Evaluates the quality or objective value of each solution
- Parameters: δ (position-changing parameter), b (random integer)
- Previous best solution: M_{best}

- Average distribution: M_{mean}

Output:

- Selected promising solutions for further exploration.

Algorithm:

```
Step 1: Initialize an empty set to store the selected promising solutions.
Step 2: For each solution, $M_s$ in the search space:
    a). Calculate the new position for the solution using Eq.(6).
    b). Evaluate the fitness value of the solution using the fitness function.
    c). Add the solution with its fitness value to the set of selected promising solutions.
Step 3: Apply a selection strategy to choose the most promising solutions from the selected solutions. The selection probability for each solution can be based on its fitness value.
Step 4: Return the selected promising solutions for further exploration in the subsequent phases of the ESO algorithm.
```

Algorithm 2 CHP

3.2.1.2. Searching Phase

The Searching Phase (SHP) explores and refines the selected solutions. ESO uses various exploration and exploitation techniques to navigate the search space and discover better solutions. Exploration techniques, such as randomization or perturbation, introduce randomness into the search process to escape local optima. They enable ESO to explore new regions of the search space and diversify the solutions. Exploitation techniques, such as local search or crossover operators, exploit the information gained from the current solutions to refine and improve them. Algorithm 3 provides the pseudocode of SHP. Local search explores the neighbourhood of a solution by making minor modifications, while crossover combines information from two or more solutions to create new offspring solutions. ESO iteratively applies these exploration and exploitation techniques to update and refine the solutions. The process involves evaluating the fitness of the new solutions and replacing the existing solutions if the new ones are better. Throughout the SHP, eagles continuously spiral and change directions in the air as they hunt for prey in the region they previously selected. Eq.(7) to Eq.(10) express the new coordinates.

$$M_{s,new} = M_s + q(s) * (M_s - M_{s+1}) + p(s) * (M_s - M_{mean}), \quad (7)$$

$$p(s) = \frac{pb(s)}{max(|pb|)}, q(s) = \frac{qb(s)}{max(|qb|)} \quad (8)$$

$$pb(s) = b(s) * sin(\rho(s)), qb(s) = bs * cos(\rho(s)) \quad (9)$$

$$\rho(s) = d * \pi * rand\ amd\ b(s) = \rho(s) + B * rand \quad (10)$$

Wherein the polar angle is indicated as $\rho(s)$ and the polar diameter is $b(s)$. The spiral trajectory is determined by the values of the parameters d and B, which can vary between 5.2 and 9.8 and 0.6 and 2.3, respectively. The polar coordinates of the eagle are given by $p(s)$ and $q(s)$, both range from -1 to +1, and the subsequently updated location of the sth eagle is given by M_{s+1}, $rand$ is a random number that ranges from 0 to 1. In absolute terms, pb is equal to $|pb|$.It's the same as writing $|qb|$ or $|pb|$.

Input:

- Selected solutions from the Choosing Phase: A set of candidate solutions
- Exploration and exploitation parameters: d, B

Output:

- Updated and refined solutions

Algorithm:

```
Step 1: For each solution, $M_s$ in the selected set:
Step 2: Calculate the polar angle $\rho(s)$ and the polar diameter b(s) using Eq.(10).
Step 3: Calculate the exploration factor $p(s)$ and exploitation factor q(s) using Eq.(8) and Eq.(9).
Step 4: Update the coordinates of the solution $M_s$ using Eq.(7).
Step 5: Evaluate the fitness value of the new solution $M_{(s+1)}$ using the fitness function.
Step 6: Replace the existing solution $M_s$ with the updated solution $M_{(s+1)}$ if the new solution has a better fitness value.
Step 7: Return the updated and refined solutions.
```

Algorithm 3 SHP

3.2.1.3. Swooping Phase

The Swooping Phase (SWP) focuses on decision-making using the information gathered during the search phase. ESO leverages the knowledge acquired to guide the selection and modification of solutions. ESO may incorporate strategies such as elitism or adaptive mechanisms in this phase. Elitism preserves the best solutions so far and ensures their inclusion in subsequent generations. Adaptive mechanisms dynamically adjust the exploration and exploitation rates based on the progress of the optimization process. These strategies help maintain diversity in the solution set and prevent premature convergence. Algorithm 4 provides the pseudocode of SWP. ESO balances exploration and exploitation by adjusting the search parameters, such as mutation rates or crossover probabilities, based on the problem's complexity and characteristics. By intelligently combining these strategies, ESO aims to converge to high-quality solutions while maintaining diverse solutions. Eagles in ESO are in continual motion throughout the swooping Stage, with some eagles swooping in from the most advantageous position found during the prior search for prey while others do the same. Eq.(11) to Eq.(14) simulate the eagle behaviour.

$$M_{s,new} = rand * M_{best} + p1(s) * (M_s - u_1 * M_{mean}) \\ +q1(s) * (M_s - u_2 - M_{best}), \tag{11}$$

$$p1(s) = \frac{pb(s)}{max(|pb|)}, q1(s) = \frac{qb(s)}{max(|qb|)} \tag{12}$$

$$xr(s) = \prod_{s=0}^{i} \left(r(s) \times \frac{1}{cosec\ (h(i))} \right) \tag{13a}$$

$$yr(s) = \sum_{s=0}^{i} \left(r(s) \times \frac{1}{\sec(h(i))} \right) \tag{13b}$$

$$pb(s) = d * \pi * rand \text{and } b(s) = \rho(s) \tag{14}$$

Wherein u_1 and u_2 all take values of one to two and enhance the eagles' movement intensity towards the best and centre places.

```
Input:
•   Selected and updated solutions from the Searching Phase: A set of candidate solutions
•   Best solution found so far: $M_{best}$
•   Average distribution: $M_{mean}$
•   Parameters: $u_1$, $u_2$
Output:
•   Updated and refined solutions
Algorithm:
Step 1:   Initialize an empty set to store the updated solutions.
Step 2:   For each solution M_s in the selected set:
        a).   Generate random numbers r(s), xr(s), and yr(s) to determine the movement intensity.
        b).   Calculate the exploration factor p1(s) and exploitation factor q1(s) based on Eq.(12) and Eq.(14).
        c).   Calculate the new position for the solution using a combination of the best solution, the current solution, and the average distribution.
        d).   Add the updated solution to the set of updated solutions.
Step 3:   Apply elitism or adaptive mechanisms to maintain diversity and prevent premature convergence.
Step 4:   Return the updated and refined solutions.
```

Algorithm 4 SWP

3.2.2. Sophisticated Eagle Search Optimization (SESO)

The predation behaviour of eagles, characterized by their spiralling trajectory as they search for and attack prey, motivates the development of the SESO algorithm. In SESO, a polar coordinate system is defined and initialized for each individual in the population. The representation of each individual is done using a binary array, denoted as (φ_t, ρ_t), where φ_t represents the polar angle and ρ_t represents the polar diameter. By incorporating this representation, the SESO algorithm aims to capture the essence of the eagles' predatory behaviour. During the initialization phase, the population is established by including more individuals based on the polar coordinate system representation. This allows for diverse solutions to be considered in the search process. Unlike other algorithms, SESO does not involve coordinate system transformations during the fitness function calculation. Instead, the fitness value is directly computed in polar coordinates using the SESO method. This method includes updating each individual's polar angle and diameter based on their fitness values and the algorithm's specific mechanisms.

The utilization of the polar coordinate system and direct computation of fitness values in polar coordinates in SESO contribute to its unique approach to optimization. By incorporating the eagles' spiralling trajectory and adapting it to the problem at hand, SESO aims to enhance the search and optimization process. This allows for the exploration of different regions in the search space and the identification of high-quality solutions. Using polar coordinates adds a distinct perspective to the algorithm and provides a framework for efficient and effective fitness evaluation.

3.2.2.1. Mathematical Modelling of SESO

During the setup step of the SESO algorithm, the polar angle (φ) and polar width (ρ) are immediately assigned and stored as a list in the coordinate system for the polar regions. As part of this process, the original points in polar coordinates are transformed into Cartesian coordinates, leading to uneven distribution.

To address the distortions introduced by the coordinate transformation, the SESO algorithm incorporates techniques based on the Archimedes theorem and the cumulative density function (CDF). These methods allow for accounting and compensating for the distortions, ensuring a more balanced distribution of points in the search space.

The Archimedes theorem, which relates the area of a sector to the length of the corresponding arc, can be utilized to calculate the appropriate polar width (ρ) for each point. By considering the desired density of points and the size of the search space, an appropriate adjustment can be made to ensure a more uniform distribution. The cumulative density function (CDF) also plays a crucial role in compensating for the uneven distribution induced by the coordinate transformation. By inverting the CDF, an initialization formula can be derived to distribute the points in a manner that counteracts the distortions and achieves a more balanced representation in the Cartesian coordinate system.

The initialization formula, derived from inverting the CDF, provides a systematic approach to distribute the points in the Cartesian coordinate system based on their desired density and the characteristics of the search space. This compensation mechanism ensures that the transformed points maintain a more even and representative distribution, enabling the SESO algorithm to explore the search space effectively. By incorporating these techniques, SESO overcomes the challenges posed by coordinate transformation-induced distortions and ensures a more uniform distribution of points. This enables the algorithm to make informed decisions and effectively navigate the search space, improving optimization results. Eq.(15) and Eq.(16) express the same.

$$\varphi = rand * (ov - zb) + zb, \qquad (15)$$

$$\rho = \gamma * \cos^{-1}(2 * rand - 1), \qquad (16)$$

Wherein ov is the maximum and zv is the minimum in SESO, and the rand is an arbitrary value between 0.02 and 1. The coefficient of disturbance, denoted by γ, can have values between 1.1 and 2. During initialization, the SESO algorithm must also maintain control over the border, and the population as a whole is free to be dispersed wherever in the search space. As a result, the values in the range $(0,2\pi)$ correspond to the polar angle ρ. For the SESO method to stay inside the boundary throughout the optimization phase, the polar diameter φ must also be established using boundaries.

After the initial setup, the current locations of all individuals in SESO are recorded. In SESO, the refreshing of the polar angle (φ) and polar width (ρ) corresponds to updating the positions of each individual. Similar to the three phases in ESO, there are three updates to the polar angle (φ) in SESO. The current formula for updating the polar angle (φ) in SESO is Eq.(17).

$$\varphi_{new} = \varphi_{old} + \{\alpha * \sin(\beta * \varphi_{old})\} + \{\gamma * \cos(\delta * \varphi_{old})\} \qquad (17)$$

where φ_{new} represents the updated polar angle, φ_{old} represents the current polar angle of an individual, and α, β, γ, and δ are coefficients that control the magnitude and direction of the update. The sinusoidal term, $\alpha * \sin(\beta * \varphi_{old})$, introduces oscillations and allows for exploration and diversification of the search space. It encourages individuals to explore different regions and helps prevent premature convergence. The cosine term, $\gamma * \cos(\delta * \varphi_{old})$, influences the direction and bias of the search by guiding individuals towards promising areas. It can assist in exploiting regions of the search space that show potential for optimal solutions. By updating the polar angle using this formula, SESO ensures that individuals explore and exploit the search space effectively. Combining the sinusoidal and cosine terms allows for a balanced exploration-exploitation trade-off, promoting diversity and convergence towards high-quality solutions.

3.2.2.2. Enhanced CHP (ECHP)

The ECHP is the update providing strategy to the polar angle (φ) in the SWP of the SESO algorithm. The formula for updating the polar angle based on Eq.(18) is as follows:

$$\varphi_{s,new} = \varphi_{best} + \delta * b(\varphi_{mean} - \varphi_s), \qquad (18)$$

where $\varphi_{s,new}$ represents the updated polar angle for the sth individual, φ_{best} represents the ideal polar angle determined as the best angle found in the previous search, φ_{mean} represents the average polar angle calculated from the individuals' polar angles, φ_s represents the current polar angle of the s th individual, δ represents the position change parameter, and b is a random integer with a value ranging from zero to one.

The update of the polar angle is calculated by taking the difference between the average polar angle and the current polar angle of the individual, multiplied by the position change parameter δ and the random integer b. This update allows the individual to adjust its polar angle towards the ideal angle determined by the best angle found during the previous search. The position change parameter δ controls the magnitude of the position change, influencing how much the individual's polar angle is adjusted towards the ideal angle. The random integer b introduces some randomness into the update process, adding diversity and exploration to the search.

By updating the polar angle using Eq.(18), the SESO algorithm enables individuals to adapt their positions based on the best angle found, the average angle, and their current angle. This adjustment helps guide the individuals towards more promising regions in the search space and contributes to the optimization process of the algorithm. Algorithm 5 provides the pseudocode of ECHP.

Input:

- Current polar angle of the sth individual: φ_s
- Ideal polar angle determined from the previous search: φ_{best}
- Average polar angle calculated from the individuals' polar angles: φ_{mean}

```
• Position change parameter: $\delta$
• Random integer b between 0 and 1
Output:
• Updated polar angle for the sth individual: $\varphi_{(s,new)}$
Algorithm:
Step 1: Generate a random integer $b$ between 0 and 1.
Step 2: Calculate the updated polar angle using Eq.(18).
Step 3: Return the updated polar angle $\varphi_{(s,new)}$.
```

Algorithm 5 ECHP

3.2.2.3. Robust Searching Phase

In the SESO algorithm's Searching Phase, the ECHP is a crucial step for adjusting the polar angles of the individuals. The following equations (19) to (22) guide this adjustment:

$$\varphi_{s,new} = \varphi_s + c_1 * (\varphi_s - \varphi_{mean}) + t_1 * (\varphi_s - \varphi_{s+1}), \qquad (19)$$

$$c_1 = \frac{\pi_1}{max(|\pi_1|)}, t_1 = \frac{\alpha_1}{max(\alpha_1)}, \qquad (20)$$

$$\pi_1 = b_1 * sin(\rho), \alpha_2 = b_1 * cos(\rho), \qquad (21)$$

$$b_1 = d * \pi * rand, \qquad (22)$$

In these equations, ρ represents the polar angle, d is the control parameter for the spiral trajectory (ranging from 4 to 12), π is the mathematical constant π_1, and $rand$ is a random number between 0 and 1. The parameter b_1 is calculated by multiplying $d, \pi,$ and $rand$. It controls the range of the polar diameter update, adding variability to the process. The control parameters π_1 and α_1 are derived from b_1 by taking the sine and cosine of the polar angle ρ , respectively. These parameters reflect the spiral trajectory and influence the adjustment of the polar angle. To ensure a balanced adjustment, the control parameters π_1 and α_1 are normalized by dividing them by their maximum absolute values. The resulting coefficients, c_1 and t_1, fall within the range of -1 to 1, allowing for controlled adjustments of the polar angle.

By incorporating these equations, the SESO algorithm updates the polar angles of the individuals based on the influence of the average angle (represented by c_1) and the angle of the following individual (represented by t_1). The randomness introduced through the control parameter b_1 contributes to exploration and diversity in the search process.

The ECHP in the Searching Phase enables the algorithm to explore and refine the solutions by adjusting the polar angles based on the average angle, the angle of the following individual, and the randomness introduced by b_1 . This iterative process aims to converge towards high-quality solutions in the routing optimization problem in WSN. Algorithm 6 provides the pseudocode of robust searching phase.

```
Input:
• Population: A set of individuals with their initial polar angles.
• Control parameter d.
• Termination condition: Specifies when to stop the iterative process.
Output:
• Updated population: Individuals with adjusted polar angles.
Algorithm:
Step 1: Initialize the population with initial polar angles for each individual.
Step 2: Calculate the average polar angle ($\varphi_{mean}$) based on the polar angles of all individuals in the population.
Step 3: Repeat until the termination condition is met:
        a). Normalize the control parameters:
        • Compute $c_1$ and $t_1$ using Eq.(20).
        b). Update the polar angle for the individual:
        • Compute $\varphi_{s,new}$ using Eq.(19) for updating the polar angle.
        • Calculate the new average polar angle ($\varphi_{mean}$) based on the updated polar angles of all individuals in the population.
        c). For each individual (s) in the population, perform the following steps:
        • Calculate the control parameters $\pi_1$ and $\alpha_1$:
        • Compute $\pi_1$ and $\alpha_2$ using Eq.(21).
        • Compute $b_1$ using Eq.(22).
Step 4: Return the updated population with adjusted polar angles.
```

Algorithm 6 Robust Searching Phase

3.2.2.4. Enhanced SWP (ESWP)

In EWSP, the focus is on refining the individuals' positions using their polar angles (φ) and introducing disturbances through the polar angle disturbance coefficient (ρ). ESWP aims to improve the fitness values of individuals and approach the global ideal fitness value.

(i). Update of the Polar Angle

The new polar angle (φ_{snew}) for each individual (s) is determined by the sum of three terms, which is mathematically expressed in Eq.(23). The first term, $rand * \varphi_{best}$ introduces randomness and exploration by considering a random value between 0 and 1 (randomly) multiplied by the best polar angle (φ_{best}) found so far. In the second term, $c_2 * (\varphi_s - u_1 * \varphi_{mean})$, adjusts the current polar angle (φ_s) towards the average polar angle (φ_{mean}) of the population, while u_1 represents a coefficient (usually set to 2) that controls the intensity of this adjustment. In the third term, $t_2 * (\varphi_s - u_2 * \varphi_{best})$, guides the individual's polar angle towards the best polar angle (φ_{best}) found in the previous search, where u_2 is another coefficient (also set to 2) determining the strength of this guidance. Eq.(23) expresses the same.

$$\varphi_{snew} = rand * \varphi_{best} + c_2 * (\varphi_s - u_1 * \varphi_{mean}) + t_2 * (\varphi_s - u_2 * \varphi_{best}) \qquad (23)$$

(ii). Normalization of Coefficients

To ensure a balanced adjustment of the polar angle, the control parameters π_2 and α_2 (computed using Eq.(24)) are normalized. The coefficients c_2 and t_2 are calculated by dividing π_2 and α_2, respectively, by their maximum absolute values. This normalization restricts their values to -1 to 1, controlling the adjustments and avoiding extreme polar angle changes.

$$c_2 = \frac{\pi_2}{max(|\pi_2|)}, t_2 = \frac{\alpha_2}{max(\alpha_2)} \qquad (24)$$

(iii). Control Parameters

The control parameters π_2 and α_2 are derived from the polar angle ρ, which represents a disturbance coefficient. π_2 is computed as the product of a randomly generated value b_1 (determined by the control parameter d in Eq.(22)) and the sine of ρ, while α_2 is obtained similarly using the cosine of ρ. These parameters influence the magnitude and direction of the polar angle adjustments, enabling exploration and exploitation in the search space which is mathematically expressed in Eq.(25).

$$\pi_2 = b_1 * sinl(\rho);\ \alpha_2 = b_1 * cosl(\rho), \qquad (25)$$

(iv). Disturbance Coefficient

The disturbance coefficient ρ_s influences the polar angle adjustments by incorporating randomness. In each iteration, ρ_s is multiplied by a disturbance coefficient γ (ranging from 0 to 2), and then a random value is added or subtracted (±) to the result. This randomness introduces diversity in the polar angle adjustments and prevents the algorithm from getting stuck in local optima. Eq.(26) mathematically expresses the disturbance coefficient.

$$\rho_{s+1} = \gamma * \rho_s \pm 2 * \cos^{-1}(2 * rand - 1), \qquad (26)$$

By recalculating an individual's position using the most recent values of φ and ρ, we compute the fitness function values for each member of the population and compare them to the global ideal fitness value of G.

The two possibilities are replacement and retention. Suppose the new fitness function value is better than the present fitness value. In that case, the first action is taken otherwise. Instead of using Cartesian coordinates, users of the SESO update their position using polar coordinates, namely φ and ρ, to determine their exact location. This will significantly increase the convergence efficiency and the update speed of people.

3.3. Analysis of Complexity for SESO

The SESO algorithm employs various parameters to control its behaviour and optimize the search process. Let's discuss the time and space complexities of SESO based on these parameters:

3.3.1. Time Complexity:

Time complexity is mainly influenced by the number of iterations (f) required for the polar diameter modifications during the searching phase. The complexity is proportional to the product of the population size (T), problem dimension (y), and the number of iterations (f), represented as $K(T * y * f)$.

3.3.1.1. Configuration of Population and Parameters:

During the initialization phase, SESO sets up the population with a specified number of search agents (T) and defines the problem dimension (y) based on the number of variables or dimensions in the search space. The time required for this configuration is denoted as $K(T * y)$.

Configuring the parameters, such as control and disturbance coefficients, is also done during this phase. The complexity for parameter setup is typically a constant time operation and does not significantly impact the overall time complexity.

3.3.1.2. Modification of Polar Diameters

The core of the SESO algorithm lies in the searching phase, where it iteratively updates the polar diameters of individuals to explore and exploit the search space. Each iteration involves all the search agents (T) undergoing polar diameter modifications.

The time complexity of modifying the polar diameters for each search agent is proportional to the problem dimension (y) since each search agent's polar angle is represented by y variables. Additionally, the number of iterations (f) influences the total number of polar diameter modifications. As a result, the time complexity for polar diameter updates is $K(T * y * f)$, and it dominates the overall time complexity of SESO.

3.3.1.3. Termination Criteria

SESO utilizes a termination criterion to decide when to stop the iterative search process. This criterion typically involves evaluating the convergence of the population or reaching a predefined number of iterations. The termination criteria are generally evaluated constantly, represented as $K(1)$, since they do not depend on the problem dimension or population size.

3.3.2. Space Complexity

Space Complexity is determined by the space required to store the search agents (population) and their positions in the search space. The space complexity depends on the number of search agents (T) and the problem dimension (y), leading to a space level of $K(T * y)$.

3.3.2.1. Search Agents

The space complexity of SESO is mainly determined by the number of search agents (T), which corresponds to the population size. Each search agent's position in the search space requires memory allocation to store the polar angles (φ) and potentially other relevant information related to the solution. The space required to store each search agent's position is directly proportional to the problem dimension (y) since it represents the number of variables needed to describe the solution accurately.

3.3.2.2. Problem's Size

The space complexity is also influenced by the problem's size (y), which directly relates to the number of variables required to represent the solutions in the search space. As the problem dimension increases, the space required to store the individuals' positions and other relevant data grows proportionally.

4. SIMULATION SETTINGS AND PERFORMANCE METRICS

Table 1 Simulation Settings

Setting	Value
Bandwidth	100Hz
Initial energy level at nodes	10J
MAC Protocol Version	CW-MAC802.11DCF
Network Boundary Limit	1.5km x 1.5km x 1.5km
Node density	350
Packet size	74 bytes
Rate of data transmission	10kbps
Runtime	300s
Sensor node's transmission range	≈350m
Sink density	4
Size of packet header	10 bytes
Transmission power	20W

A comprehensive analysis of routing protocols in Mobility Enabled Wireless Sensor Networks (MEWSN) involves the utilization of various simulations. To evaluate the effectiveness of the proposed routing protocol against the existing ones, NS3 simulations are employed. Researchers have faced significant challenges in modelling and implementing protocols in MEWSN, particularly regarding the network's overall performance. Consequently, the study aims to thoroughly examine the design, strengths, and limitations of both the proposed and currently employed routing protocols. The study reveals that the NS3 simulator performs better when integrated with C++.Table 1 provides the simulation settings used for evaluating the protocols.

5. RESULTS AND DISCUSSION

5.1. Delay Analysis

Figure 2 comprehensively compares the delay performance among three routing protocols, CSSR, EEACR, and SESO-GAODV, across node density scenarios. In Ad Hoc networks, delay is a critical metric, representing the time data packets travel from source to destination, directly influencing real-time communication efficiency.

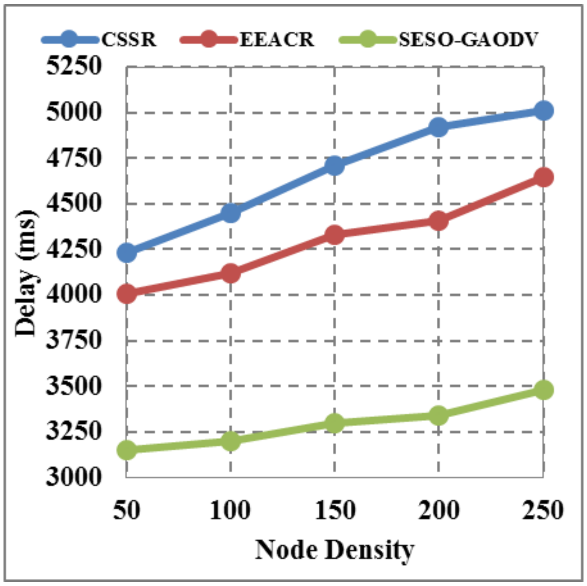

Figure 2 Delay

Upon analyzing the average delay values obtained from Table 2, it is evident that CSSR exhibits the highest average delay of 4664.6 ms. The delay escalates with increasing node density, making CSSR less favourable for dense networks. This can be attributed to the cluster-based approach of CSSR, which introduces overhead in selecting sub-graphs and may result in increased packet delivery times, particularly in scenarios with numerous potential paths for packets to traverse. EEACR demonstrates a relatively better average delay of 4301.8 ms. It effectively balances energy-efficient routing with delay performance, showing promising results across node densities. The adaptability of EEACR in identifying energy-efficient routes contributes to its lower delay values than CSSR. However, it is worth noting that in some instances, the cooperative communication among nodes in EEACR may introduce slight overhead, affecting the delay performance.

SESO-GAODV emerges as the most efficient protocol, boasting the lowest average delay of 3294.4 ms. By optimizing route selection and packet forwarding with excellent efficiency, SESO-GAODV achieves superior delay performance, even in high-density scenarios. This can be attributed to the intelligent optimization capabilities of the sophisticated eagle search algorithm, which enables SESO-GAODV to efficiently explore the solution space and identify optimal routes.

The analysis demonstrates that SESO-GAODV outperforms both CSSR and EEACR in minimizing delays. Due to higher contention and interference, CSSR faces challenges in dense networks, leading to elevated average delay values. While EEACR balances energy efficiency and delay, SESO-GAODV stands out with its advanced capabilities. It is the most promising choice for achieving low delays and efficient communication in Ad Hoc networks, particularly in scenarios with high node density. Further improvements in CSSR and EEACR might be necessary to enhance their delay performance and competitiveness with SESO-GAODV.

Table 2 Results of Delay Analysis

Node Density	CSSR	EEACR	SESO-GAODV
50	4231	4008	3152
100	4452	4117	3203
150	4708	4330	3297
200	4923	4409	3341
250	5009	4645	3479
Average	4664.6	4301.8	3294.4

5.2. Packet Delivery Ratio

Figure 3 presents a comparative analysis of the PDR for three routing protocols, CSSR, EEACR, and SESO-GAODV, in different node density scenarios.

The PDR represents the percentage of successfully delivered packets from the source to the destination node, a critical measure of the protocols' performance in ensuring reliable data transmission in Ad Hoc networks. Analyzing the average PDR values from Table 3, we observe the following:

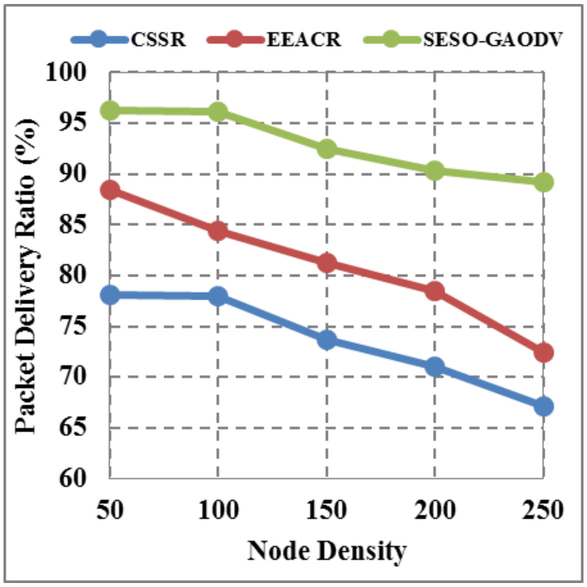

Figure 3 Packet Delivery Ratio

The CSSR protocol achieves an average PDR of 73.580%. CSSR creates clusters of nodes and selects efficient sub-graphs for routing. However, the average PDR values for CSSR are lower than the other two protocols. This may be attributed to the inherent challenges of cluster-based approaches handling increased node density, resulting in reduced packet delivery efficiency. The EEACR protocol exhibits an average PDR of 80.998%. EEACR operates on the principles of Energy-efficient Adaptive cum Cooperative Routing. It balances energy-efficient routing decisions with cooperative communication among nodes. The average PDR values for EEACR are relatively higher than CSSR, suggesting its ability to handle packet delivery more effectively, even in scenarios with moderate node density.

The SESO-GAODV protocol stands out with an impressive average PDR of 92.877%. SESO-GAODV employs the Sophisticated Eagle Search Optimization-based Gaussian Ad Hoc On-demand Distance Vector mechanism, leveraging intelligent optimization techniques. This enables SESO-GAODV to efficiently explore optimal routes and achieve superior packet delivery performance, particularly in high-density node scenarios.

The PDR analysis reveals that SESO-GAODV outperforms CSSR and EEACR regarding successful packet delivery. While CSSR and EEACR demonstrate reasonable performance, their average PDR values are surpassed by the advanced optimization capabilities of SESO-GAODV. The intelligent optimization techniques in SESO-GAODV make it highly desirable for ensuring reliable and successful packet delivery in Ad Hoc network environments, even under challenging conditions of high node density. CSSR and EEACR could benefit from further improvements in their mechanisms to enhance their packet delivery performance and competitiveness with SESO-GAODV.

Table 3 Results of Packet Delivery Ratio Analysis

Node Density	CSSR	EEACR	SESO-GAODV
50	78.143	88.435	96.262
100	77.987	84.365	96.114
150	73.657	81.224	92.446
200	70.998	78.492	90.343
250	67.114	72.472	89.221
Average	73.580	80.998	92.877

5.3. Packet Loss Ratio

Figure 4 presents an evaluation of the Packet Loss Ratio (PLR) for three routing protocols: CSSR, EEACR, and SESO-GAODV, across different node density scenarios. The PLR represents the percentage of lost packets during transmission, a crucial metric indicating the protocols' efficiency in preserving data integrity and minimizing data loss in Ad Hoc networks. Analyzing the average PLR values from Table 4, we observe the following:

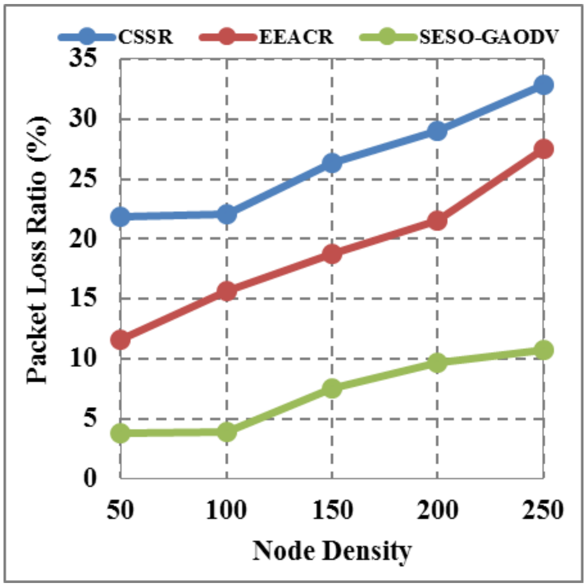

Figure 4 Packet Loss Ratio

CSSR shows an average PLR of 26.420%. Operating on Cluster Sub-graph Selection based Routing, CSSR creates node clusters and selects sub-graphs for routing. The higher average PLR values for CSSR suggest challenges in handling packet losses, particularly in scenarios with denser node populations. The cluster-based approach might increase contention and interference, leading to packet loss during transmission. EEACR exhibits an average PLR of 19.002%. Based on Energy-efficient Adaptive cum Cooperative Routing, EEACR balances energy efficiency and cooperative communication among nodes. The lower average PLR values for EEACR indicate better packet loss handling and data integrity maintenance, even in scenarios with moderately dense node distributions.

SESO-GAODV stands out with an impressive average PLR of 7.123%. Leveraging Sophisticated Eagle Search Optimization-based Gaussian Ad Hoc On-demand Distance Vector mechanism, SESO-GAODV employs intelligent optimization techniques. The lower average PLR values for SESO-GAODV demonstrate its superior ability to minimize packet loss during transmission, even in high-density node scenarios. The PLR analysis reveals that SESO-GAODV outperforms both CSSR and EEACR in minimizing packet loss. While CSSR and EEACR demonstrate reasonable performance, their average PLR values are comparatively higher than SESO-GAODV. The advanced optimization techniques in SESO-GAODV make it highly effective in reducing packet loss and preserving data integrity in Ad Hoc network environments. Further enhancements in the mechanisms of CSSR and EEACR might be beneficial to improve their ability to handle packet loss and match the superior performance of SESO-GAODV.

Table 4 Results of Packet Loss Ratio Analysis

Node Density	CSSR	EEACR	SESO-GAODV
50	21.857	11.565	3.738
100	22.013	15.635	3.886
150	26.343	18.776	7.554
200	29.002	21.508	9.657
250	32.886	27.528	10.779
Average	26.420	19.002	7.123

5.4. Throughput Analysis

Figure 5 illustrates the Throughput analysis of three distinct routing protocols: CSSR, EEACR, and SESO-GAODV, under varying node density scenarios. Throughput is a critical performance metric in Ad Hoc networks, quantifying the data transmission rate from the source node to the destination node. Upon analyzing the average Throughput values obtained from Table 5, the following insights can be derived:

CSS Rexhibits an average Throughput of 217.313 Kbps. While CSSR demonstrates moderate data transmission efficiency, its throughput experiences a slight decline in denser node density situations. This is due to the inherent challenges of cluster-based routing approaches in handling increased interference and contention, leading to a comparatively lower Throughput. EEACR achieves an average Throughput of 221.483 Kbps. EEACR balances energy efficiency and cooperative communication among nodes, leading to a slightly improved data transmission rate compared to CSSR. It maintains relatively stable Throughput values across node densities, showcasing its adaptability to different network conditions.

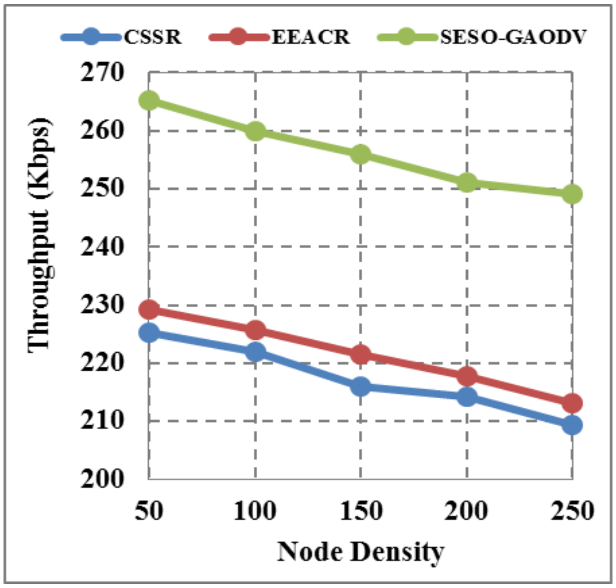

Figure 5 Throughput

Table 5 Results of Throughput Analysis

Node Density	CSSR	EEACR	SESO-GAODV
50	225.244	229.323	265.221
100	221.965	225.659	259.987
150	215.892	221.493	255.998
200	214.186	217.713	251.138
250	209.279	213.227	249.145
Average	217.313	221.483	256.298

SESO-GAODV stands out with an impressive average Throughput of 256.298 Kbps. The protocol's intelligent optimization techniques enable it to efficiently explore optimal routes, leading to significantly higher data transmission rates. SESO-GAODV excels in high-density node scenarios, maintaining superior Throughput performance. The Throughput analysis emphasizes the efficiency of SESO-GAODV in achieving higher data transmission rates compared to CSSR and EEACR. While CSSR demonstrates moderate performance and EEACR exhibits stability, SESO-GAODV outperforms both protocols, especially in dense node populations. The advanced optimization capabilities in SESO-GAODV contribute to its superior throughput and overall network performance in Ad Hoc environments. CSSR and EEACR could benefit from further enhancements in their mechanisms to improve throughput and match the performance of SESO-GAODV in various node-density scenarios.

5.5. Energy Consumption Analysis

Figure 6 presents the Delay analysis of three routing protocols, CSSR, EEACR, and SESO-GAODV, conducted under varying node density scenarios. Delay is a crucial metric in Ad Hoc networks, representing the time data packets traverse from the source to the destination node. A lower delay indicates more efficient and faster data transmission. Analyzing the average Delay values from Table 6, this research observes the following:

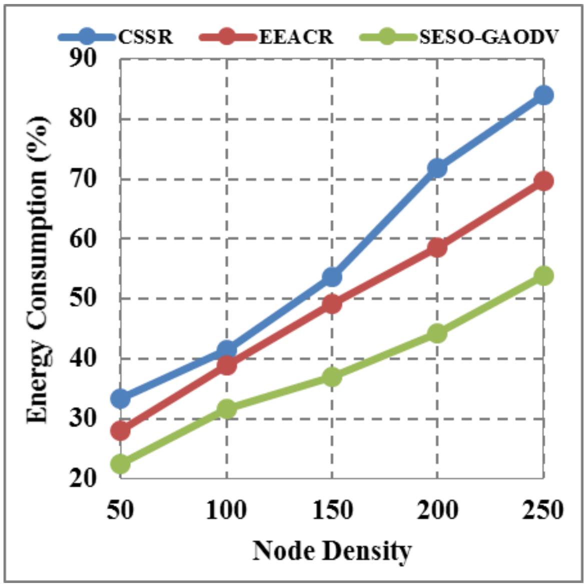

Figure 6 Delay

Analyzing the performance of CSSR at different node densities, as provided in Table 6, this research observes that CSSR exhibits moderate delay performance at lower node densities (50 and 100), with average delays of 33.412 ms and 41.463 ms, respectively. However, as the node density increases to 150, 200, and 250, CSSR encounters significant challenges in handling complex routing decisions, resulting in noticeably higher average delays of 53.696 ms, 71.903 ms, and 84.076 ms, respectively. The cluster-based mechanism increases contention and interference at higher node densities, leading to longer transmission times. Thus, CSSR's performance is limited in scenarios with denser node populations, and it may benefit from further enhancements to handle such challenges effectively.

Table 6 Results of Delay Analysis

Node Density	CSSR	EEACR	SESO-GAODV
50	33.412	27.905	22.362
100	41.463	38.945	31.673
150	53.696	49.244	36.986
200	71.903	58.581	44.170
250	84.076	69.681	53.818
Average	56.910	48.871	37.802

Evaluating the performance of EEACR across different node densities, this research finds that EEACR consistently maintains a relatively stable delay performance. At lower node densities (50 and 100), EEACR achieves average delays of 27.905 ms and 38.945 ms, respectively, showcasing its ability to transmit data efficiently with minimal delays. As the node density increases to 150, 200, and 250, EEACR continues demonstrating stable delay performance, with average delays of 49.244 ms, 58.581 ms, and 69.681 ms, respectively.

However, EEACR's delay values are slightly higher than the advanced SESO-GAODV protocol, which leverages intelligent optimization techniques. Nonetheless, EEACR's cooperative communication and energy-efficient routing mechanism make it a robust choice for ensuring reliable data transmission across various node densities.

SESO-GAODV is a proposed routing protocol that stands out for its impressive delay performance. At node density 50, SESO-GAODV achieves the lowest average delay of 22.362 ms, and this trend continues across higher node densities, with average delays of 31.673 ms (100 nodes), 36.986 ms (150 nodes), 44.170 ms (200 nodes), and 53.818 ms (250 nodes). SESO-GAODV's superior delay performance can be attributed to its intelligent optimization techniques, which enable it to efficiently explore optimal routes and make intelligent routing decisions even in densely populated networks.

As the node density increases, SESO-GAODV maintains lower delays than CSSR and EEACR, making it a highly efficient choice for data transmission in Ad Hoc networks with varying node densities. The protocol's ability to minimize delays even in challenging high-density scenarios positions it as a promising solution for ensuring timely and reliable data delivery.

6. CONCLUSION

The proposed Sophisticated Eagle Search Optimization-Based Gaussian Ad Hoc On-demand Distance Vector (SESO-GAODV) routing protocol showcases its potential to significantly enhance the performance of Mobility Enabled Wireless Sensor Networks (ME-WSNs). By leveraging the intelligent optimization techniques of Sophisticated Eagle Search Optimization and the dynamic route discovery capabilities of Gaussian Ad Hoc On-demand Distance Vector (GAODV), SESO-GAODV effectively addresses the challenges posed by dynamic and rapidly changing network topologies. Through extensive evaluations and comparisons with existing routing protocols, SESO-GAODV demonstrates superior results in reduced delay, increased throughput, minimized packet loss, and lower energy consumption. The protocol's adaptability to changing network conditions and efficient handling of node mobility make it a promising solution for ensuring reliable data transmission and optimized network performance in ME-WSNs. The energy-efficient nature of SESO-GAODV also contributes to a prolonged network lifetime, supporting sustainable and seamless communication in dynamic and challenging environments. SESO-GAODV holds great potential for advancing the efficiency, reliability, and sustainability of ME-WSNs, offering valuable insights for further research and implementation in real-world wireless sensor network applications.

REFERENCES

[1] A. Islam, K. Akter, N. J. Nipu, A. Das, M. Mahbubur Rahman, and M. Rahman, “IoT Based Power Efficient Agro Field Monitoring and Irrigation Control System : An Empirical Implementation in Precision Agriculture,” in 2018 International Conference on Innovations in Science, Engineering and Technology, ICISET 2018, 2018, pp. 372–377. doi: 10.1109/ICISET.2018.8745605.

[2] N. Gharaei, Y. D. Al-Otaibi, S. A. Butt, S. J. Malebary, S. Rahim, and G. Sahar, “Energy-Efficient Tour Optimization of Wireless Mobile Chargers for Rechargeable Sensor Networks,” IEEE Syst. J., vol. 15, no. 1, pp. 27–36, 2021, doi: 10.1109/JSYST.2020.2968968.

[3] J. Martin Sahayaraj and J. M. Ganaseakar, “Relay node selection with energy efficient routing using hidden Markov model in wireless sensor networks,” Int. J. Netw. Virtual Organ., vol. 19, no. 2–4, pp. 176–186, 2018, doi: 10.1504/IJNVO.2018.095420.

[4] L. Rajaoarisoa, N. K. M’Sirdi, M. Sayed-Mouchaweh, and L. Clavier, “Decentralized fault-tolerant controller based on cooperative smart-wireless sensors in large-scale buildings,” J. Netw. Comput. Appl., vol. 214, p. 103605, 2023, doi: 10.1016/j.jnca.2023.103605.

[5] F. Niaz, M. Khalid, Z. Ullah, N. Aslam, M. Raza, and M. K. Priyan, “A bonded channel in cognitive wireless body area network based on IEEE 802.15.6 and internet of things,” Comput. Commun., vol. 150, pp. 131–143, Jan. 2020, doi: 10.1016/j.comcom.2019.11.016.

[6] T. Waheed, Aqeel-ur-Rehman, F. Karim, and S. Ghani, “QoS Enhancement of AODV Routing for MBANs,” Wirel. Pers. Commun., vol. 116, no. 2, pp. 1379–1406, Jan. 2021, doi: 10.1007/s11277-020-07558-x.

[7] Y. Han, H. Hu, and M. Yao, “Trust-Aware Secure Routing Protocol for

Wireless Sensor Networks," Jisuanji Gongcheng/Computer Eng., vol. 47, no. 9, pp. 145–152, 2021, doi: 10.19678/j.issn.1000-3428.0058217.

[8] G. Valecce, S. Strazzella, A. Radesca, and L. A. Grieco, "Solarfertigation: Internet of things architecture for smart agriculture," in 2019 IEEE International Conference on Communications Workshops, ICC Workshops 2019 - Proceedings, 2019. doi: 10.1109/ICCW.2019.8756735.

[9] L. Guezouli, K. Barka, S. Bouam, and A. Zidani, "A variant of random way point mobility model to improve routing in wireless sensor networks," Int. J. Inf. Commun. Technol., vol. 13, no. 4, pp. 407–423, 2018, doi: 10.1504/IJICT.2018.095031.

[10] L. Mani, S. Arumugam, and R. Jaganathan, "Performance Enhancement of Wireless Sensor Network Using Feisty Particle Swarm Optimization Protocol," ACM Int. Conf. Proceeding Ser., pp. 1–5, Dec. 2022, doi: 10.1145/3590837.3590907.

[11] D. Jayaraj, J. Ramkumar, M. Lingaraj, and B. Sureshkumar, "AFSORP: Adaptive Fish Swarm Optimization-Based Routing Protocol for Mobility Enabled Wireless Sensor Network," Int. J. Comput. Networks Appl., vol. 10, no. 1, pp. 119–129, Jan. 2023, doi: 10.22247/ijcna/2023/218516.

[12] J. Ramkumar and R. Vadivel, "Multi-Adaptive Routing Protocol for Internet of Things based Ad Hoc Networks," Wirel. Pers. Commun., vol. 120, no. 2, pp. 887–909, Apr. 2021, doi: 10.1007/s11277-021-08495-z.

[13] B. Kang, C. Park, and H. Choo, "A Location Aware Fast PMIPv6 for Low Latency Wireless Sensor Networks," IEEE Sens. J., vol. 19, no. 20, pp. 9456–9467, 2019, doi: 10.1109/JSEN.2019.2925637.

[14] M. A. Uddin, A. Mansour, D. Le Jeune, and E. H. M. Aggoune, "Agriculture internet of things: AG-IoT," in 2017 27th International Telecommunication Networks and Applications Conference, ITNAC 2017, 2017, vol. 2017-Janua, pp. 1–6. doi: 10.1109/ATNAC.2017.8215399.

[15] P. K. Dalela et al., "Constraint-Driven IoT-Based Smart Agriculture for Better e-Governance," Advances in Intelligent Systems and Computing, vol. 1077. pp. 177–186, 2020. doi: 10.1007/978-981-15-0936-0_18.

[16] X. Liu, J. Yu, W. Zhang, and H. Tian, "Low-energy dynamic clustering scheme for multi-layer wireless sensor networks," Comput. Electr. Eng., vol. 91, p. 107093, 2021, doi: 10.1016/j.compeleceng.2021.107093.

[17] M. Boushaba, A. Hafid, and M. Gendreau, "Node stability-based routing in Wireless Mesh Networks," J. Netw. Comput. Appl., vol. 93, pp. 1–12, 2017, doi: 10.1016/j.jnca.2017.02.010.

[18] A. Chowdhury and D. De, "MSLG-RGSO: Movement score based limited grid-mobility approach using reverse Glowworm Swarm Optimization algorithm for mobile wireless sensor networks," Ad Hoc Networks, vol. 106, p. 102191, 2020, doi: 10.1016/j.adhoc.2020.102191.

[19] P. Maheshwari, A. K. Sharma, and K. Verma, "Energy efficient cluster based routing protocol for WSN using butterfly optimization algorithm and ant colony optimization," Ad Hoc Networks, vol. 110, p. 102317, 2021, doi: 10.1016/j.adhoc.2020.102317.

[20] J. Aranda, D. Mendez, H. Carrillo, and M. Schölzel, "A framework for multimodal wireless sensor networks," Ad Hoc Networks, vol. 106, p. 102201, 2020, doi: 10.1016/j.adhoc.2020.102201.

[21] K. Patil, M. Jafri, D. Fiems, and A. Marin, "Stochastic modeling of depth based routing in underwater sensor networks," Ad Hoc Networks, vol. 89, pp. 132–141, 2019, doi: 10.1016/j.adhoc.2019.03.009.

[22] X. Hao, N. Yao, L. Wang, and J. Wang, "Joint resource allocation algorithm based on multi-objective optimization for wireless sensor networks," Appl. Soft Comput. J., vol. 94, p. 106470, 2020, doi: 10.1016/j.asoc.2020.106470.

[23] M. R. Rahman, M. M. Islam, A. I. Pritom, and Y. Alsaawy, "ASRPH: Application Specific Routing Protocol for Health care," Comput. Networks, vol. 197, p. 108273, 2021, doi: 10.1016/j.comnet.2021.108273.

[24] P. Ghosh, H. Ren, R. Banirazi, B. Krishnamachari, and E. Jonckheere, "Empirical evaluation of the heat-diffusion collection protocol for wireless sensor networks," Comput. Networks, vol. 127, pp. 217–232, 2017, doi: 10.1016/j.comnet.2017.08.018.

[25] H. Liu and K. Y. Ki, "Application of wireless sensor network based improved immune gene algorithm in airport floating personnel positioning," Comput. Commun., vol. 160, pp. 494–501, 2020, doi: 10.1016/j.comcom.2020.04.036.

[26] B. Chakraborty, S. Verma, and K. P. Singh, "Temporal Differential Privacy in Wireless Sensor Networks," J. Netw. Comput. Appl., vol. 155, p. 102548, 2020, doi: 10.1016/j.jnca.2020.102548.

[27] J. Lu, L. Feng, J. Yang, M. M. Hassan, A. Alelaiwi, and I. Humar, "Artificial agent: The fusion of artificial intelligence and a mobile agent for energy-efficient traffic control in wireless sensor networks," Futur. Gener. Comput. Syst., vol. 95, pp. 45–51, Apr. 2019, doi: 10.1016/j.future.2018.12.024.

[28] N. Khernane, J. F. Couchot, and A. Mostefaoui, "Maximum network lifetime with optimal power/rate and routing trade-off for Wireless Multimedia Sensor Networks," Comput. Commun., vol. 124, pp. 1–16, 2018, doi: 10.1016/j.comcom.2018.04.012.

[29] N. V. S. S. R. Lakshmi, S. Babu, and N. Bhalaji, "Analysis of clustered QoS routing protocol for distributed wireless sensor network," Comput. Electr. Eng., vol. 64, pp. 173–181, Nov. 2017, doi: 10.1016/j.compeleceng.2016.11.019.

[30] F. Ullah, M. Zahid Khan, M. Faisal, H. U. Rehman, S. Abbas, and F. S. Mubarek, "An Energy Efficient and Reliable Routing Scheme to enhance the stability period in Wireless Body Area Networks," Comput. Commun., vol. 165, pp. 20–32, 2021, doi: 10.1016/j.comcom.2020.10.017.

[31] S. Doostali and S. M. Babamir, "An energy efficient cluster head selection approach for performance improvement in network-coding-based wireless sensor networks with multiple sinks," Comput. Commun., vol. 164, pp. 188–200, 2020, doi: 10.1016/j.comcom.2020.10.014.

[32] D. Wang, J. Liu, and D. Yao, "An energy-efficient distributed adaptive cooperative routing based on reinforcement learning in wireless multimedia sensor networks," Comput. Networks, vol. 178, p. 107313, 2020, doi: 10.1016/j.comnet.2020.107313.

[33] A. Rajini, N. Nithya "Hybrid Intrusion Detection System in IOT Network Environments" Compliance Engineering Journal, vol.10, no.11, pp.541-548, 2019.

Authors

Mr. V. Veerakumaran is an Assistant Professor in Department of Computer Science at Nehru Arts and Science College (Autonomous), affiliated with Bharathiar University, Coimbatore. He has twelve years of teaching experience. He obtained M.C.A., degrees from Bharathiar University, Coimbatore, Tamil Nadu, India. He has attended/presented the research papers in various Seminars, Conferences and Workshop at National and International level. He has made significant contributions to research in the field of Networking. Area of focuses on Data Mining, Artificial Intelligence, Data Science, Machine learning and Computational Intelligence.

Dr. Aruchamy Rajini holds a doctorate degree in Computer Science from Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore. Presently, she is an Assistant Professor in the Department of Computer Science (Aided) in NGM College, Pollachi, with 24+ years of teaching experience. She has published several research papers in referred journals and conferences. She has attended many FDP, Workshops, Webinars and Training Programs. She acted as a session chairperson and resource person in various colleges, and also has edited several book chapters. Her areas of research interest include Networking, Data Mining and Image Processing.

RESEARCH ARTICLE

How to cite this article:

V. Veerakumaran, Aruchamy Rajini, "Performance Enhancement of Mobility-Enabled Wireless Sensor Network Using Sophisticated Eagle Search Optimization-Based Gaussian Ad Hoc On-Demand Distance Vector (SESO-GAODV) Routing Protocol", International Journal of Computer Networks and Applications (IJCNA), 10(5), PP: 816-833, 2023, DOI: 10.22247/ijcna/2023/223428.