SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V11I7P102
Volume 11 Issue 7, 10-19, July 2024

Original Article

Optimized Multiple Mobile-Sink Energy Efficient Clustering Algorithm in Wireless Sensor Networks

Naseem Y. Siddiqui[1], Sachin Vasant Chaudhari[2]

[1,2]*Sanjivani College of Engineering, Savitribai Phule Pune University, Maharashtra, India.*

[1]*Corresponding Author : siddiquinaseemetc@sanjivani.org.in*

Received: 02 May 2024 Revised: 03 June 2024 Accepted: 01 July 2024 Published: 27 July 2024

Abstract - *Due to their versatility, Wireless Sensor Networks (WSNs) have garnered substantial attention. One critical aspect of WSNs is energy efficiency, particularly in scenarios involving multiple mobile sinks. This research proposes an Optimized Multiple Mobile-Sink Energy Efficient Clustering Algorithm (O-MMECA) scheme to enhance energy utilization and protract network lifetime. The O-MMECA scheme integrates an improvised Artificial Bee Colony (ABC) algorithm. Firstly, the clustering approach is employed to define a Cluster Head (CH), which is elected by each cluster responsible for data aggregation as well as transmission to the mobile sinks. Secondly, multiple mobile sinks are optimally deployed to collect data from CHs, reducing the communication distance and energy consumption. Thirdly, an energy-aware routing protocol is utilized to establish efficient paths from CHs to sinks, considering node energy levels and network conditions. The effectiveness of the O-MMECA scheme in comparison with the existing approaches is demonstrated by the simulations. It achieves significant improvisations in energy efficiency, network lifetime, and data delivery rates, making it suitable for applications requiring reliable data collection in WSNs with multiple mobile sinks.*

Keywords - *Energy efficiency, Wireless sensor networks, Clustering approach, Multiple mobile sinks, Routing protocol.*

1. Introduction

WSNs are defined by decentralized nodes, or sensor nodes, that can detect their immediate surroundings and transmit that data to a main hub or sink node [1]. To monitor and transmit data on the target phenomena wirelessly in multi-hop communication, sensor nodes that run on batteries are dispersed randomly across the study area [2]. To construct a WSN, a vast array of interconnected sensor nodes interacts with one another. Since the sensor nodes are spread out over a large geographical region, the network can be partially or completely linked depending on the situation [3, 4]. A field can be equipped with a vast array of wireless sensor nodes, collectively known as a WSN. The sensor devices that come with these nodes are not very powerful, and they can only communicate wirelessly. WSNs are superior to other options for environmental monitoring, security, and monitoring [5, 6]. As far as WSN advancements are concerned, sensor nodes still rely on low-power batteries to power themselves. The usage of WSNs in remote areas also makes it hard to repair or recharge the sensor nodes' batteries. One of the most pressing issues in WSNs is achieving effective energy management, considering these limitations. Network lifespan enhancement needs should be addressed while developing a new protocol for these types of networks [7].

It is common practice to use multi-hop communication to send data produced by sensor nodes to the sink [8, 9]. The relaying of packets from other sensor nodes might drain a node's battery life and eventually cause it to die in this kind of network [10, 11]. Because of this issue, it will become disconnected from the network, which will cause coverage and connection issues throughout the network. Using mobility in sensor networks, such as a mobile sink or agent, to gather data from sensor nodes is one way to tackle this issue [12, 13]. Methods based on movable sinks [14, 15] include the mobile sink traveling to several meeting places around the network. One of the difficulties with WSNs is choosing the best meeting places for the mobile sink. The likelihood of choosing an ideal sensor node as a meeting place is quite low [16, 17] since, similar to hierarchical techniques, these sites are only chosen using local knowledge.

The rest of the paper is organized in the following manner. In the beginning, the motivation for the research is provided followed by a background study leading to the problem definition. In the next section, the various methods and algorithms to improve the efficiency of the WSNs are discussed. In the following section, the results are presented, and the analysis is discussed, leading to the conclusion section.

1.1. Motivation

The motivation lies in addressing the critical need for energy efficiency in WSNs, especially in scenarios involving multiple mobile sinks. It introduces the O-MMECA scheme, which integrates clustering, optimal deployment of mobile sinks, and energy-aware routing. These mechanisms aim to enhance energy utilization, prolong network lifetime, and improve data delivery rates.

1.2. Background Study

A. M. Alabdali et al. [1] have endeavored to construct a system for wireless energy balancer-based clustering that was energy-efficient in this research. The proposed architecture reduces and balances the energy consumption of CHs in two parts.

The first part of the given framework introduces n-level clustering, whereas the second portion also makes use of an energy balancer, whose job is to equalize energy consumption, therefore decreasing wasted energy and bringing the leftover energy of CHs to zero.

M. Aydin et al. [3] The method of clustering sensor nodes was one approach to achieving energy efficiency and power balancing in WSNs. Using this method, one of the sensor nodes was chosen to serve as the cluster's CH. This node's job was to gather information from the CMs in its cluster and send it to the sink. An essential step for any cluster was picking the suitable node to act as a CH between other nodes and changing it at the necessary intervals.

N. Ghorpade and P. Vijaykarthik [6] For the real-time processing of contemporary BigData applications, the ideal and desired paradigm was an energy-efficient WSN with minimal latency.

He, X. et al. [7] these authors research introduce Energy-Efficient Trajectory Planning (EETP), a MOPSO-based energy-efficient trajectory planning technique.

O. Ogundile et al. [12] A clustered routing system with mobility-specific selective-path priority tables is suggested in this work. Using certain simple but effective criteria, the PT was constructed by giving precedence to the two shortest pathways from the source SDs to the CH nodes or the sink. The MSPT routing protocol has demonstrated promise for a long time in reducing the energy consumption of the network, hence extending its lifespan and improving its overall performance.

Y. M. Raghavendra and U. B. Mahadevaswamy [14] A stochastic hill climbing–guided mobile sink solution was suggested in this paper. The paper suggested a mobile sink that uses stochastic hill climbing to direct its motions to optimize the number of relay hops. The mobile sink is reached by taking geographic routing steps from the node. The author compared the proposed approach to current methods and ran simulations for various network configurations.

2. Problem Definition

The existing methodologies, including MMECA, LEACH, PEGASIS, MMSR, and E-MMECA, have certain drawbacks that need to be addressed. PEGASIS, originally designed for static sink-based WSNs, faces scalability issues in scenarios with multiple mobile sinks. LEACH, while energy-efficient for static sink scenarios, cannot be optimized for multiple mobile sinks.

MMSR can suffer from suboptimal routing paths and energy consumption in large mobile sink setups. MMECA, although designed for multiple mobile sinks, might still lack full optimization for energy efficiency and network longevity. E-MMECA, while an improvement over MMECA, cannot fully address all energy consumption challenges in dynamic WSN environments.

3. Methods

The proposed methodology aimed at enhancing energy efficiency in WSNs with multiple mobile sinks is mentioned in detail in this section. The methodology, named O-MMECA, incorporates a comprehensive approach to address the challenges of energy utilization and network longevity in dynamic WSN environments.

3.1. Enhance Sink Placement

Enhancing sink placement involves optimizing the positioning of data collection points, storage locations, or endpoints within a network or system. This process aims to improve data flow efficiency, reduce latency, enhance scalability, and optimize resource utilization by strategically placing sinks based on factors such as traffic patterns, proximity to data sources, and network topology.

Analyzing the nodes sequentially is the basic premise of total flow analysis. Figure 2 shows that cross traffic must also be considered in the many sink situations. This ensures that the nodes with cross-traffic have precisely computed arrival and service curves. The arrival and service curves are computed for those nodes following the steps outlined in Algorithm 2.

Starting at the sink, the first approach iteratively calculates the output boundaries of every node along the route from the source to the sink. After that, in order to get the effective service curve of the specified node β_i^{eff}, we subtract β_i from a_{excl}. After the calculation of the output bound is completed, for every node, the effective service curve and the min-plus deconvolution of the overall traffic towards the sink of the flow of interest (limited by a_i^{pred}) are calculated and saved.

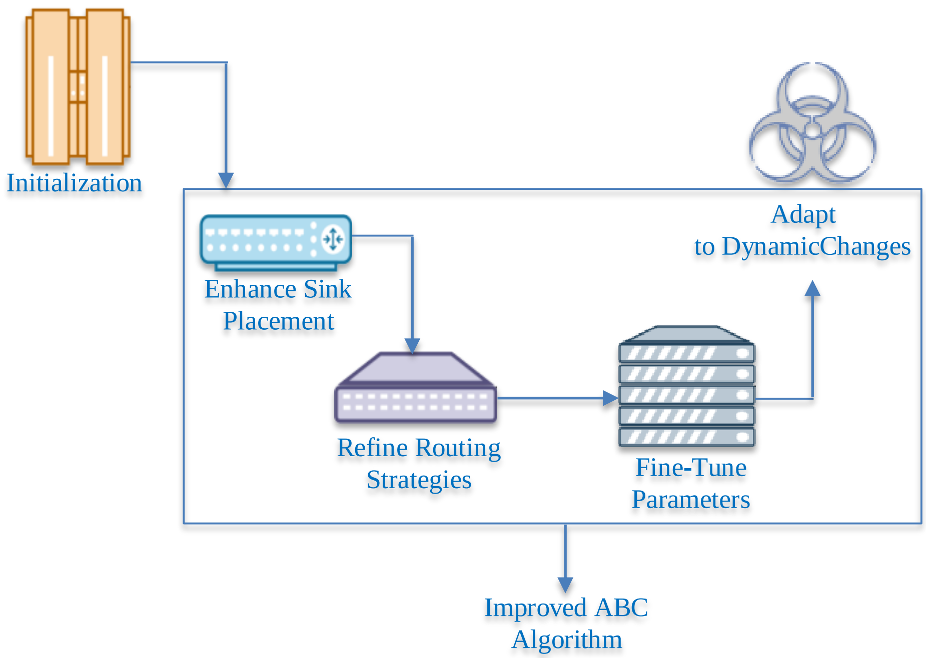

Fig. 1 Overall architecture

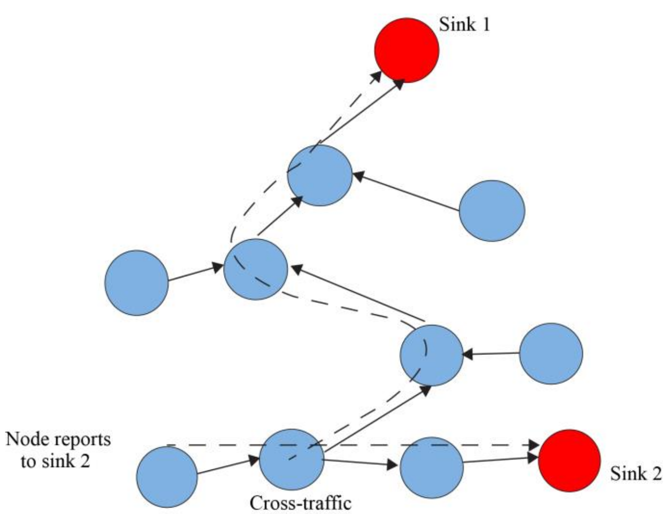

Fig. 2 Illustration of cross traffic in scenarios with multiple sinks

3.2. Refine Routing Strategies

Refining routing strategies involves optimizing the selection and management of paths for data, resources, or vehicles within a network or system. This process aims to improve efficiency, reduce congestion, enhance reliability, and minimize costs by utilizing advanced algorithms, real-time data, and predictive analytics.

Within dense networks, we take into account a two-dimensional wireless network comprised of n S-D pairings that are evenly and independently dispersed across a square of unit area. For example, it is shown that the whole region is partitioned into u square cells, with one single-antenna BS occupying the center of each cell. The assumption is that there are no nodes physically situated within the BSs. There is a relationship between the parameters n and m.

$$m = n^{\beta} \tag{1}$$

Also, these BSs are not considered sources or destinations, and it is presumed that the capacity of the BS-to-BS linkages is limitless. It is assumed that there is a constant average transmit power limitation P for each node and BS. While it can be present at the receivers, it is not present at the transmitters. Assumption: Depending on its position in the network, each node can transfer data packets at varying rates. For the entire network throughput, we use $T\ (n)$. We simplify implementation by not assuming any complex multi-user detection algorithms at each receiver.

What follows is a description of the uplink signal model. The set of wireless nodes that can transmit data at the same time, $J \subset \{1,\cdots n\}$, is a subset of the total number of transmitters in the network. Next, for a certain time instance, the received signal y_k at BS $k \in \{1,\cdots m\}$ is provided by

$$y_k = \sum_{i \in I} h_{ki} x_i + n_k \tag{2}$$

We can also think of an extended network with a unit node density as a basic network model. Even though the specifics are not shown in this research, it is feasible to achieve a logarithmic gain, also known as power gain, in extended networks under certain circumstances by permitting a total transmit power restriction ZP for our BS-based transmission.

We think that a power gain can be achieved even for networks with arbitrary sizes with n, and our study can be extended to the general scenario where the network area ranges from 1 to n.

3.3. Fine-Tune Parameters

To improve performance or accomplish desired results, it is common practice to fine-tune parameters inside a system, model, algorithm, or process. This involves modifying particular settings or variables. Software development, optimization, engineering, and machine learning are just a few of the many areas that often use this method. Hyperparameter tuning is a common way to fine-tune parameters in data science and machine learning. Model parameters, including learning rate, regularization strength, batch size, and model complexity, are controlled via hyperparameters, which are external variables to the model. To get the optimal combination of hyperparameters for example, more accuracy, lower error rates, or better generalization it is necessary to change their values and assess the model's performance systematically.

The policy network receives the pre-trained embedding x S from the source domains and produces a binary vector. $P_e^{(i)}$ As an output for the embedding layer. In Equation (3), the i^th element of this vector, $P_e^{(i)}(x^s)$, determines whether the i-th feature field should use the fine-tuned embedding parameters or the pre-trained ones. The feature embeddings V_i^T are regulated by the policy network, and each field is chosen from either pre-trained or fine-tuned parameters.

$$x_i^T = P_e^{(i)}(x^s)V_i^S X_{Hot,i} + \left(1 - P_e^{(i)}(x^S)\right) V_i^T X_{hot,i} \tag{3}$$

Parameter tuning, as used in optimization and engineering, is the process of fine-tuning existing systems or process settings to get the best possible outcome. Manufacturing process control parameter adjustments to maximize output or minimize waste, numerical optimization algorithm parameter adjustments to converge faster to the global optimum, and feedback control system parameter adjustments to achieve desired stability and response characteristics are all examples of what is meant by this.

3.4. Adapt to Dynamic Changes

Systems processes can adapt to dynamic changes when they can fundamentally alter in response to changes in their environment, input data, needs, or circumstances. It shows up in software and hardware systems as auto-scaling web apps, adaptive algorithms in machine learning, and dynamic resource allocation in cloud computing.

$$\left(1 + a1z^{-1} + a2z^{-2} + \cdots + a_{n_a}z^{na}\right)y(k) \tag{4}$$

$y(k)$: Represents the output of the system at time k.

$z^{-1}, z^{-2}, \ldots, z^{na}$: These terms represent the time-delayed input signals to the system. They indicate how past input values affect the current output, allowing the system to capture temporal dynamics and adapt accordingly.

Quick reaction to changing requirements is made possible by agile approaches in business and management, while organizations can adjust to growing client wants via market adaption strategies. In order to stay alive, biological systems adapt to their environments by maintaining internal stability. When faced with disruptions, engineering and control systems use resilience tactics and feedback mechanisms to keep performance at a desirable level.

$$\left(b_0 + b_1z^{-1} + b_2z^{-2} + \cdots + b_{n_b}z^{-nb}\right)u\,(k) + \varepsilon(k) \tag{5}$$

3.5. Optimization Using an Improved ABC Algorithm

Optimization using the improved ABC algorithm involves enhancing the traditional ABC algorithm to achieve better convergence speed, accuracy, and solution quality in solving optimization problems. This improvement typically includes modifications such as adaptive parameter tuning, dynamic search strategies, and enhanced exploration-exploitation balance, resulting in more efficient and effective optimization processes across various domains.

To determine the optimal path, the improved ABC algorithm employs bee populations; the "colony" is made up of three different kinds of bees: employed, scout, and bystander bees. Every bee signifies a location inside the search field. A bee is classified as a spectator if it waits on the "dance" area to select a food source, a scout if it searches randomly, and an employee if it returns to a food source it has already visited. Whereas food source locations represent prospective solutions, a food source's "nectar" quantity represents the quality (fitness) of a potential solution to the optimization problem. The lower part of the colony is home to observer bees, while worker bees occupy the upper half.

The four main steps of the ABC algorithm are mentioned below. 1. Initialization: Pretend that the original population's food supply is denoted by N and that the population size is SN. In the optimization problem, O is the vector dimension, and $Y_i = \{a_{i1}, a_{i2}, \ldots, a_{iD}\}$ $(i = 1, 2, \ldots, N)$. Afterwards, the initially random population is

$$X_i = X_{min} + rand(0,1).(X_{max} - X_{min}) \tag{6}$$

Here, X_i represents the initial random population $X_{min}\ and\ X_{max}$ minimum and maximum values of the search spa $rand(0,1)$ Random number generator between 0 and 1.

2. Population Updating: Every worker bee starts with a randomly assigned food source; after that, it iteratively finds a new food source that is nearby using (8), calculates the amount of nectar from that new source, and repeats the process. If one

food source's nectar content is greater than another's, the worker bee will transfer to that source; otherwise, it will stick with the old one.

$$V_{ij} = X_{ij} + rand(-1, 1).\left(X_{ij} - X_{k_j}\right) \tag{7}$$

Here, V_{ij} is the new food source location for the worker bee i in dimension j , X_{ij} and X_{kj} are current and other randomly selected food source locations, and $rand(-1, 1)$ function generates a random number generator between -1 and 1.

The range of the generating neighborhood of Y_{ib} is controlled by the numerical value rand (−1, 1), where k and a are both elements of the set$\{1, 2, 3, \ldots, S_N\}$, and a is also an element of the set $\{1, 2, 3, \ldots, D\}$. When searching for the best answer, the local scope becomes smaller with time. Bee Source Selection: At this point, the worker bees' actions are dictated by the income rate (obtained from their sources using fitness value)

$$P_i = \frac{fit(X_i)}{\sum_{n=1}^{SN} fit(X_n)} \tag{8}$$

Here, P_i Probability of choosing the food source X_i based on its fitness, $fit(X_i)$ is the fitness value of food source X_i, and SN is the number of solutions in the population.

The fitness value of the solution $n(n)$ multiplied by the quantity of nectar from the food source $n \in \{1, 2, 3, \ldots, S_N\}$ is called fit(X_n). Here is how fitness is determined:

$$Fit(X_n) = \begin{cases} \frac{1}{f(X_n)} & f(x_n) \geq 0 \\ 1 + abs\left(f(x_n)\right) & f(X_n) < 0 \end{cases} \tag{9}$$

Here an represents the value of the objective function for the bee source an the algorithm's local exploitative ability is enhanced when the following bees hunt in the area of the sources.

Population Elimination: In the event that a particular solution does not show any discernible progress after continual limit cycle updates, it is therefore considered to have entered a local optimum and is discarded. In this case, the bees that were watching transform into scouts and, at random, create a new solution to take its place.

$$X_{ij} = X_{minj} + rand(0, 1)\left(X_{maxJ} - X_{min\ j}\right) \tag{10}$$

X_{ij} New food source location for scout bee i in dimension j.
X_{minj} and X_{maxJ} Minimum and maximum values in dimension j.
$rand(0, 1)$ Random number generator between 0 and 1.

Algorithm 1: O-MMECA

Initialization:
Initialize a few parameters, such as the sensing range of nodes, the number of mobile sinks, etc.
Steps:
• **Population Updating**:
  • Each sensor node acts as a "bee" and randomly chooses a cluster to join based on proximity to the cluster head.
  $V_{ij} = X_{ij} + rand(-1, 1).\left(X_{ij} - X_{k_j}\right)$
  • Calculate the energy consumption for each node based on its distance to the cluster head and other factors.
  $P_i = \frac{fit(X_i)}{\sum_{n=1}^{SN} fit(X_n)}$
  • Update the cluster heads' positions (mobile sinks) based on the energy consumption and cluster stability criteria.
• **Bee Source Selection**:
  • Based on its energy consumption, stability, and network coverage, calculate the fitness value for each cluster.
  • Compute the probability of each cluster being selected as a source for further optimization based on its fitness value.
• **Population Elimination**:
  • Evaluate the performance of each cluster and eliminate poorly performing clusters.
  $X_{ij} = X_{minj} + rand(0, 1)\left(X_{maxJ} - X_{min\ j}\right)$
  • Reassign nodes from eliminated clusters to other nearby clusters or create new clusters if necessary.
• **Optimized Multiple Mobile-Sink Energy Efficient Clustering**:
  • Repeat the population updating, bee source selection and population elimination steps for multiple iterations or until convergence criteria are met.
  $Fit(X_n) = \begin{cases} \frac{1}{f(X_n)} & f(x_n) \geq 0 \\ 1 + abs\left(f(x_n)\right) & f(X_n) < 0 \end{cases}$
  • Optimize the clustering structure to achieve network coverage, energy efficiency and stability while considering the mobility of sinks and the dynamic nature of wireless sensor networks.
Output:
Output the optimized clustering structure, including the positions of mobile sinks (cluster heads) and the assignment of sensor nodes to clusters.

The previous answer is substituted with the new one that is produced via computation, and the best solution is then outputted as a result. The numerical value between the intervals of -1 and 1 is represented by rand (0, 1), and the maximum and lowest values are Xmax and Xmin, respectively, for$b \in \{1, 2, 3, \ldots, D\}$.

One novel approach to intelligent population optimization, the ABC algorithm, has several benefits. Among its many advantages are the following: (1) its convergence to the whole is fast and relatively smooth; (2) its applicability is broad; (3) its parameter setup is simple in comparison to other optimal algorithms; and (4) its foundation in population makes it easy to implement and handle.

4. Results and Discussion

In this section, the results obtained from implementing the proposed O-MMECA are presented, and their implications and significance in enhancing energy efficiency in WSNs with multiple mobile sinks are discussed.

$$\text{Throughput} = \frac{Number\ of\ Packet\ Size}{Time\ duration * Successful\ average\ Packet\ size} \quad (11)$$

Table 1. Throughput comparison table

	Throughput					
Packet Size	**PEGASIS**	**LEACH**	**MMSR**	**MMECA**	**E-MMECA**	**O-MMECA**
50	0.172	0.212	0.243	0.263	0.303	0.344
100	0.344	0.425	0.487	0.526	0.606	0.689
150	0.517	0.638	0.731	0.789	0.909	1.034
200	0.689	0.851	0.975	1.052	1.212	1.379
250	0.862	1.063	1.219	1.315	1.515	1.724

Table 2. Energy comparison table

	Energy in joules					
Number of Nodes	**PEGASIS**	**LEACH**	**MMSR**	**MMECA**	**E-MMECA**	**O-MMECA**
10	0.833	0.769	0.666	0.588	0.526	0.476
20	1.666	1.538	1.333	1.176	1.052	0.952
40	3.333	3.076	2.667	2.352	2.105	1.904
60	5.000	4.615	4.001	3.529	3.157	2.857
80	6.666	6.153	5.334	4.705	4.210	3.809
100	8.333	7.692	6.667	5.882	5.263	4.761

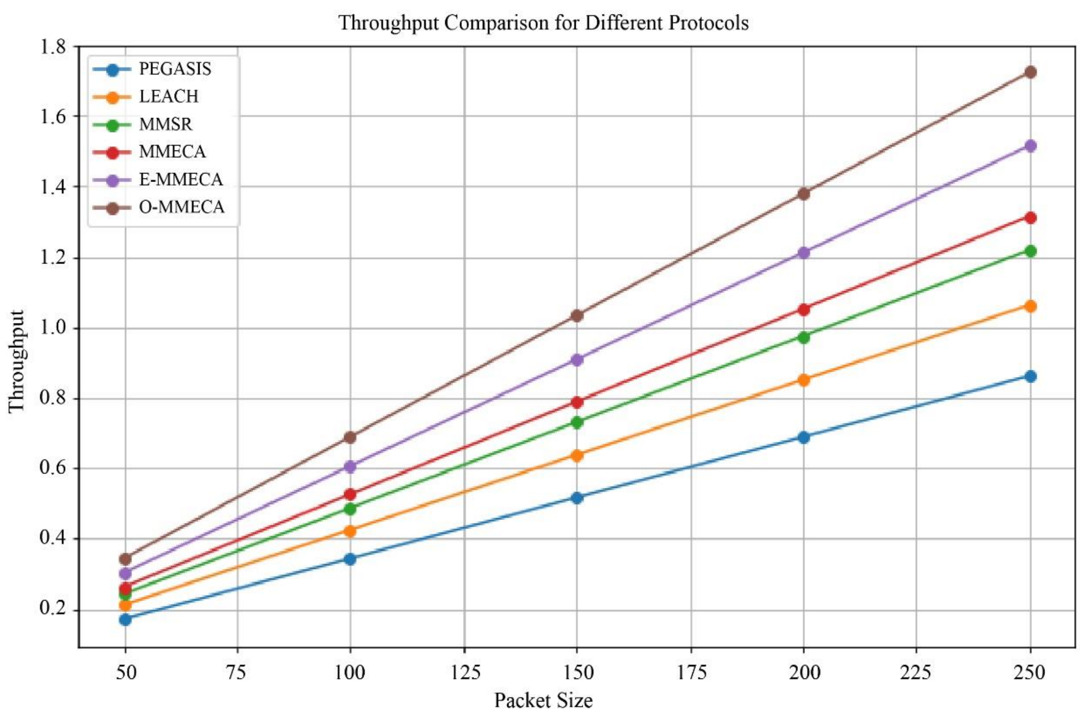

Fig. 3 Comparison chart of the throughput

Table 1 and Figure 3 show throughput results that indicate the data transfer rates achieved in different protocols across varying packet sizes in WSNs with multiple mobile sinks. Overall, the Optimized Multiple Mobile-Sink Energy Efficient Clustering Algorithm (O-MMECA) consistently outperforms other protocols such as MMECA, LEACH, PEGASIS, MMSR, and E-MMECA in terms of throughput. Specifically, at smaller packet sizes (50 and 100), O-MMECA demonstrates substantial improvements over other protocols, with throughput values ranging from 0.344 to 1.724. As packet size increases, the gap between O-MMECA and other protocols widens, highlighting O-MMECA's efficiency in handling larger data payloads and maintaining high throughput rates, making it a promising solution for data-intensive applications in WSNs with multiple mobile sinks by

$$\text{Energy} = \frac{\textit{Number of Sensor nodes}}{\textit{Energy consumption for sending packets at a times}} \tag{12}$$

Table 2 and Figure 4 show energy consumption results reveal the energy usage in joules by different protocols as the number of nodes in Wireless Sensor Networks (WSNs) with multiple mobile sinks increases.

Notably, the Optimized Multiple Mobile-Sink Energy Efficient Clustering Algorithm (O-MMECA) consistently demonstrates lower energy consumption compared to other protocols like MMECA, LEACH, PEGASIS, MMSR, and E-MMECA across varying node numbers. Specifically, at higher node counts (80 and 100 nodes), O-MMECA achieves significant energy savings, with values ranging from 3.809 to 8.333 joules. This underscores O-MMECA's effectiveness in optimizing energy utilization and prolonging network lifetime, making it a promising solution for energy-constrained WSNs with multiple mobile sinks.

$$\text{Time Delay} = \frac{\textit{Number of Sensor nodes}}{\textit{energy consumption for sending packets at a times x forwarding time in ms}} \tag{13}$$

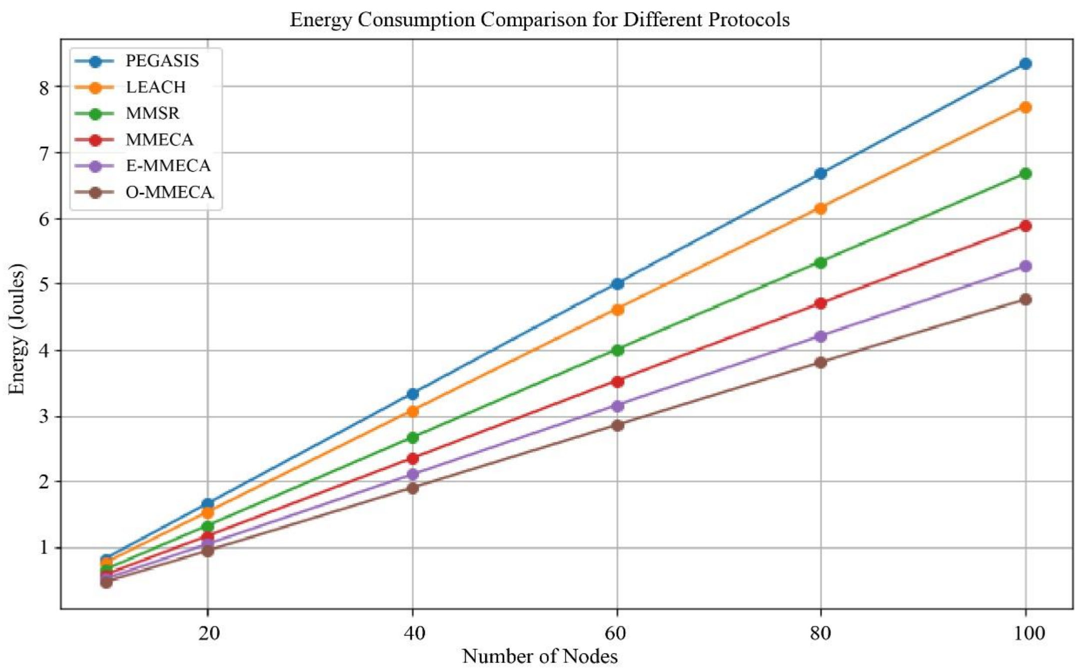

Fig. 4 Comparison chart of the energy

Table 3. Time delay comparison table

	Time (End-to-End Delay)					
Number of Nodes	**PEGASIS**	**LEACH**	**MMSR**	**MMECA**	**E-MMECA**	**O-MMECA**
10	0.066	0.066	0.063	0.062	0.057	0.056
20	0.133	0.132	0.127	0.124	0.114	0.112
40	0.267	0.265	0.255	0.248	0.229	0.224
60	0.401	0.398	0.383	0.372	0.344	0.337
80	0.535	0.531	0.511	0.496	0.459	0.449
100	0.669	0.664	0.639	0.621	0.574	0.562

Table 4. Packet delivery ratio comparison table

	Packet Delivery Ratio					
Number of Packets	**PEGASIS**	**LEACH**	**MMSR**	**MMECA**	**E-MMECA**	**O-MMECA**
50	96.2	96.4	96.6	97.6	98.6	99.2
100	98.1	98.2	98.3	98.8	99.3	99.6
150	98.7	98.8	98.86	99.2	99.53	99.7
200	99.05	99.1	99.15	99.4	99.65	99.8
250	99.24	99.28	99.32	99.52	99.72	99.84

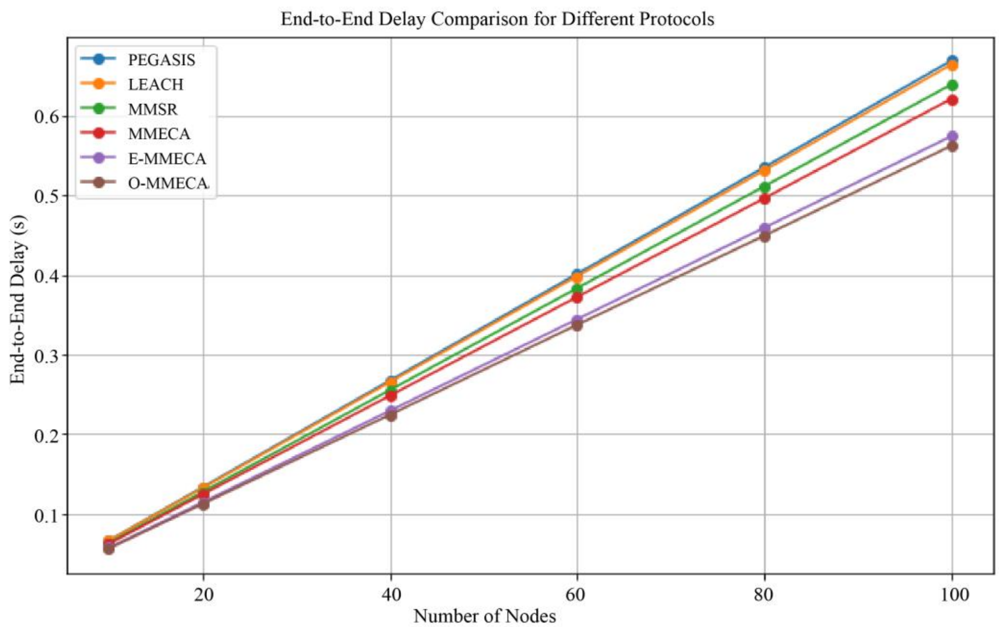

Fig. 5 End-to-end delay comparison chart

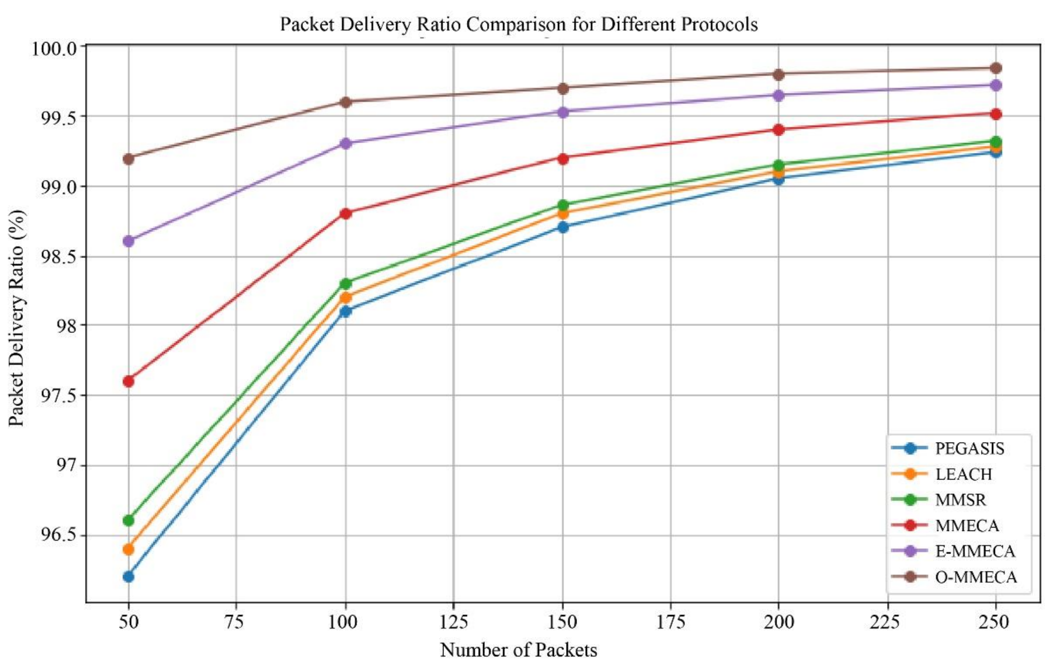

Fig. 6 Packet delivery ratio comparison chart

Table 3 and Figure 5 illustrate the end-to-end delay results and the time taken for data packets to travel from source to destination nodes in WSNs with multiple mobile sinks as the number of nodes increases. The Optimized Multiple Mobile-Sink Energy Efficient Clustering Algorithm (O-MMECA) consistently exhibits lower end-to-end delays compared to other protocols like MMECA, LEACH, PEGASIS, MMSR, and E-MMECA across varying node numbers.

Particularly at higher node counts (80 and 100 nodes), O-MMECA demonstrates significant reductions in end-to-end delay, with values ranging from 0.449 to 0.669 seconds. This underscores O-MMECA's efficiency in establishing efficient communication paths and minimizing latency, making it a suitable choice for time-sensitive applications in WSNs with multiple mobile sinks.

$$\text{PDR} = \frac{\mathit{Number\ of\ Packets\ Receive}}{\mathit{Total\ Packets}} \times 100 \tag{14}$$

Table 4 and Figure 6 show packet delivery ratio results showcase the percentage of successfully delivered packets by different protocols as the number of packets increases in WSNs with multiple mobile sinks. O-MMECA scheme consistently exhibits higher packet delivery ratios compared to other protocols like PEGASIS, MMSR, MMECA, and E-MMECA across varying numbers of packets. Particularly at larger packet counts (200 and 250 packets), O-MMECA achieves significant improvements in packet delivery ratios, with values ranging from 99.05% to 99.84%.

This emphasizes O-MMECA's effectiveness in ensuring reliable and robust data delivery, making it well-suited for applications requiring high packet delivery rates in WSNs with multiple mobile sinks.

5. Conclusion

Wireless Sensor Networks (WSNs) that use multiple mobile sinks can improve their energy efficiency and extend their network lifespan with the help of the O-MMECA scheme. As a result of its integration of clustering, energy-aware routing, and optimum mobile sink deployment, O-MMECA successfully solves important problems encountered by WSNs in dynamic situations. The clustering method streamlines data collection and transmission to mobile sinks by effectively organizing sensor nodes into clusters with specified Cluster Heads (CHs). This enhances resource consumption and increases network scalability while also decreasing overhead.

Additionally, by strategically placing several mobile sinks, communication distances can be minimized, resulting in substantial energy savings and improved data transmission speeds. When an energy-aware routing protocol is implemented, it enhances the network's intelligence by optimizing data transmission channels according to node energy levels and network circumstances. In situations when energy restrictions and mobility patterns are variable, this adaptive routing system improves resilience and dependability.

References

[1] Aliaa M. Alabdali, Niayesh Gharaei, and Arwa A. Mashat, “A Framework for Energy-Efficient Clustering with Utilizing Wireless Energy Balancer,” *IEEE Access*, vol. 9, pp. 117823-117831, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Bilal R. Al-Kaseem et al., “Optimized Energy–Efficient Path Planning Strategy in WSN with Multiple Mobile Sinks,” *IEEE Access*, vol. 9, pp. 82833-82847, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] Muhammed Ali Aydin, Baybars Karabekir, and Abdül Halim Zaim, “Energy Efficient Clustering-Based Mobile Routing Algorithm on WSNs,” *IEEE Access*, vol. 9, pp. 89593-89601, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] Khalid A. Darabkh et al., “An Innovative Cluster-Based Power-Aware Protocol for Internet of Things Sensors Utilizing Mobile Sink and Particle Swarm Optimization,” *Neural Computing and Applications*, vol. 35, no. 26, pp. 19365-19408, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Niayesh Gharaei et al., “Collaborative Mobile Sink Sojourn Time Optimization Scheme for Cluster-Based Wireless Sensor Networks,” *IEEE Sensors Journal*, vol. 18, no. 16, pp. 6669-6676, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[6] Naveen Ghorpade, and P. Vijaykarthik, “Efficient Mobile Sink Path Scheduling for Clustered Based Wireless Sensor Network,” *IOP Conference Series: Materials Science and Engineering*, vol. 1022, no. 1, pp. 1-8, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] Xiaolin He, Xiuwen Fu, and Yongsheng Yang, “Energy-Efficient Trajectory Planning Algorithm Based on Multi-Objective PSO for the Mobile Sink in Wireless Sensor Networks,” *IEEE Access*, vol. 7, pp. 176204-176217, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[8] Baybars Karabekir, Muhammed Ali Aydın, and Abdül Haim Zaim, “Energy-Efficient Clustering-Based Mobile Routing Algorithm for Wireless Sensor Networks,” *Electrica*, vol. 21, no. 1, pp. 41-49, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] Ajay Kaushik, S. Indu, and Daya Gupta, “Adaptive Mobile Sink for Energy Efficient WSN Using Biogeography-Based Optimization,” *International Journal of Mobile Computing and Multimedia Communications*, vol. 10, no. 3, pp. 1-22, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[10] Kambiz Koosheshi, and Saeed Ebadi, “Optimization Energy Consumption with Multiple Mobile Sinks Using Fuzzy Logic in Wireless Sensor Networks,” *Wireless Networks*, vol. 25, pp. 1215-1234, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[11] Uttej Kumar Nannapanenia et al., "Cluster-Based Collection Point Energy Efficient Routing Protocol for the Mobile Sink in Wireless Sensor Network," *International Journal of Future Generation Communication and Networking*, vol. 13, no. 2, pp. 787-796, 2020. [Google Scholar] [Publisher Link]

[12] Olayinka O. Ogundile et al., "Energy-Balanced and Energy-Efficient Clustering Routing Protocol for Wireless Sensor Networks," *IET Communications*, vol. 13, no. 10, pp. 1449-1457, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[13] R. Radhika, and G.S Binu, "An Enhanced Energy Efficient Routing Method for Mobile Sink Using Ant Colony Optimisation Based Clustering in WSN," *International Conference on Circuits and Systems in Digital Enterprise Technology*, Kottayam, India, pp. 1-6, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[14] Y.M. Raghavendra, and U.B. Mahadevaswamy, "Energy Efficient Routing in Wireless Sensor Network Based on Mobile Sink Guided By Stochastic Hill Climbing," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 6, pp. 5965-5973, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] V. Saranya, S. Shankar, and G.R. Kanagachidambaresan, "Energy Efficient Clustering Scheme (EECS) for Wireless Sensor Network with Mobile Sink," *Wireless Personal Communications*, vol. 100, pp. 1553-1567, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[16] Shayesteh Tabatabaei, and Amir Mohsen Rigi, "Reliable Routing Algorithm Based on Clustering and Mobile Sink in Wireless Sensor Networks," *Wireless Personal Communications*, vol. 108, no. 4, pp. 2541-2558, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[17] Shamineh Tabibi, and Ali Ghaffari, "Energy-Efficient Routing Mechanism for Mobile Sink in Wireless Sensor Networks Using Particle Swarm Optimization Algorithm," *Wireless Personal Communications*, vol. 104, pp. 199-216, 2019. [CrossRef] [Google Scholar] [Publisher Link]