

Research Article

An Intelligent Energy-Efficient Data Routing Scheme for Wireless Sensor Networks Utilizing Mobile Sink

Hassan Al-Mahdi ¹, Mohamed Elshrkawey ², Shymaa Saad ³ and Safa Abdelaziz ¹

¹Faculty of Computers and Informatics, Computer Science Department, Suez Canal University, Ismailia, Egypt

²Faculty of Computers and Informatics, Information System Department, Suez Canal University, Ismailia, Egypt

³Department of Computer Science, El Arish University, Arish, Egypt

Correspondence should be addressed to Hassan Al-Mahdi; hassanwesf@yahoo.com

Received 9 August 2023; Revised 18 February 2024; Accepted 19 February 2024; Published 18 March 2024

Academic Editor: Kechen Zheng

Copyright © 2024 Hassan Al-Mahdi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Data collection and energy consumption are critical concerns in Wireless sensor networks (WSNs). To address these issues, both clustering and routing algorithms are utilized. Therefore, this paper proposes an intelligent energy-efficient data routing scheme for WSNs utilizing a mobile sink (MS) to save energy and prolong network lifetime. The proposed scheme operates in two major modes: configure and operational modes. During the configure mode, a novel clustering mechanism is applied once, and a prescheduling cluster head (CH) selection is introduced to ensure uniform energy expenditure among sensor nodes (SNs). The scheduling technique selects successive CHs for each cluster throughout the WSNs' lifetime rounds, managed at the base station (BS) to minimize SN energy consumption. In the operational mode, two main objectives are achieved: sensing and gathering data by each CH with minimal message overhead, and establishing an optimal path for the MS using the genetic algorithm. Finally, the MS uploads the gathered data to the BS. Extensive simulations are conducted to verify the efficiency of the proposed scheme in terms of stability period, network lifetime, average energy consumption, data transmission latency, message overhead, and throughput. The results demonstrate that the proposed scheme outperforms the most recent state-of-the-art methods significantly. The results are substantiated through statistical validation via hypothesis testing utilizing ANOVA, as well as post hoc analysis.

1. Introduction

Wireless sensor networks (WSNs) are made up of several resource-constrained sensor nodes (SNs). The SNs are used to sense the physical environment around them and transmit their sensory data to the base station (BS) [1]. Unlike SNs, the BS has substantial resources. So, it gathers data, analyze it, and then sends the useful information via the internet to the cloud server or end user [2]. Nowadays, WSNs technology has become smart and can be exploited for different functionalities: intercommunication, decision-making ability, military surveillance networks, tracking the objects' movements and their speeds, and monitor critical circumstances, such as temperature, humidity, and pressure [3–6]. In Jain et al.'s [7] study, the authors survey hierarchical routing protocols in WSNs with mobile sink (MS), focusing on event-driven and query-driven scenarios. They discuss the challenges of sink mobility and the need for tailored routing protocols based on application

requirements. The paper provides a comparative analysis of these protocols, highlighting their functionalities, advantages, and performance parameters.

One of the main protective aspects in the WSN architecture of SNs is to avoid the probable breach of cyber criminals. Attackers may try to amend the behavioral pattern of normal SNs through different types of attacks: eavesdropping, the node capture, and spoofing attacks. So, securing the routed aggregated data from SNs to the sink is one of the most imperative issues. Generally, the sensors of wireless network may expose to an adversary breaches by sensing the wireless channel and capturing the data being transferred in an unauthorized manner. In Okine et al.'s [8] study, the authors present a novel approach for routing in tactical wireless sensor networks (T-WSNs) used in military operations. These networks face unique challenges like jamming attacks, which disrupt data communication and complicate packet routing. The proposed solution utilizes distributed multiagent deep

reinforcement learning to overcome these challenges and find reliable routes while meeting strict delay and energy requirements. It considers factors such as hop count, one-hop delay, packet loss rate, and energy cost in action reward estimation. Comparative analysis shows that the proposed scheme outperforms existing algorithms in terms of packet delivery ratio, packet delivery time, and energy efficiency, making it a promising solution for routing in T-WSNs under jamming attacks.

However, WSNs are often established in an inaccessible environment, and the SNs suffer from the batteries limitation which cannot be charged or replaced [9, 10]. According to these limitations, energy preservation is a vital aspect in the creation of an effective routing protocol. In addition, the extending of lifetime of the WSNs are precisely associated with the SN's battery life [11]. For the design of an effective routing protocol, several goals should be achieved: minimize the consumed energy, maximize the packets delivery ratio, enhance throughput, extending network lifetime, and decline computational overhead. It is recognized that creating routing protocol is based on two aspects: an efficient clustering method and using MS to communicate to the BS. The achievement of the aforementioned goals is significantly impacted by these factors [12]. So, several research has been performed on the energy saving of the SNs [13, 14].

The present energy-saving methods partition the entire network area into distinct clusters. Each cluster has a cluster head (CH) [15]. The major role of a CH is to gather the sensory data from its cluster members (CMs) and send it to the BS, which considerably expands the lifetime of the networks. However, the main drawback of the applied clustering techniques is caused by the SNs nearer to the CH. These SNs send more packets than the distant SNs. So, they are exposed to the premature death. This problem is resolved by two strategies: first repartition the observing area into several separated network segments that may not be able to communicate with the BS causing poor network performance [16]. Second, some clustering-methods turn the CH role on all sensors to distribute the energy consumption among the CMs. But the current approaches need massive overhead messages which causes transmission delays and affects the networks performance. The presented approaches have been performed using heavyweight work to resolve routing problems (e.g., heuristic and meta-heuristic-based routing algorithms).

These approaches are designed without considering the mentioned limitations [17]. Moreover, some of the approaches may not be scalable with respect to the network size. So, the presented approaches suffer when applied in real-environment WSNs [18]. Recently, several approaches have emerged employing MS for data gathering from the deployed SNs [17]. Next, MS delivers the gathered data to the sink. This leads to decreases the energy consumption of SNs and expands the lifetime of the WSNs [19]. MS-based approaches can be classified into two types [20]. In the first type, MS passes to each SN, gathering its data, and finally send the gathered data of the whole SNs to BS. This strategy reduces the energy consumption of SNs and balances the energy utilization among them. But the data

collected from each SN causes great data gathering latency and leads to buffer overflow within each SN. The second type employs rendezvous points (RVPs) strategy to overcome the first type of problem. The RVPs are positioned within the WSNs such that the MS visits these positions to acquire the data from SNs/CHs. However, this strategy suffers from the massive message overhead (MO) to find RVPs within the WSNs. Moreover, it does not consider energy balancing among the SNs, which causes the premature death of the WSNs.

The proposed work's significant contribution can be summed up as follows:

- (1) This paper presents an intelligent energy-efficient data routing scheme for WSNs utilizing MS which aims to balance the consumed energy through the anticipated operations in the WSNs topology.
- (2) The proposed work further proposes a minimum data gathering tour for MS that considerably decreases the data collecting time and enhances the overall performance of the networks.
- (3) Furthermore, the proposed scheme constructs a pre-scheduling map to make the spent energy by each SN approximately the same as the remaining SNs or as near as to be the same as other SNs.
- (4) The proposed work introduces a clustering mechanism to divide the monitoring area into equal-size clusters in the configure mode and before enabling the topology operations.
- (5) In addition, a modified time division multiple access (TDMA) is presented to assign a constant and ordered slot time for each sensor during its sensing operation. Moreover, the CHs operations are distributed among sensors in prescheduled order through configure mode.
- (6) Additionally, the proposed scheme adds a better mechanism that enables each SN to self-activate as a CH in its order without the use of extra communication messages.
- (7) The genetic algorithm (GA) is utilized for MS trajectory optimization.
- (8) Statistical validation of the comparative results is further conducted utilizing ANOVA and subsequent post hoc analysis.

The proposed scheme offers versatile applications across diverse real-world scenarios. In precision agriculture, the scheme optimizes data gathering tours, enabling efficient monitoring of soil conditions and crop health. This aids farmers in making informed decisions regarding irrigation and fertilization. Additionally, in wildlife monitoring applications, the MS navigates through wildlife habitats, collecting data on animal behavior and environmental conditions without causing disruptions. In industrial automation, the scheme enhances efficiency by optimizing data collection in manufacturing processes. It minimizes the time required for gathering critical data, improving decision-making in control systems. For smart cities infrastructure, the MS strategically collects data from SNs in urban environments,

optimizing city management systems, and reducing response times. Healthcare monitoring benefits from the scheme's efficient data collection, ensuring timely and accurate health monitoring in both hospital and remote patient care settings. Furthermore, in environmental monitoring, the scheme's adaptability allows it to navigate challenging terrains, optimizing data collection for climate and ecological research. Overall, these scenarios highlight the scheme's broad applicability, showcasing its potential impact in addressing energy efficiency and performance challenges in WSNs across various real-world.

The original source of this paper under different title is available at SSRN [21]. The outstanding paper is organized as follows: Section 2 describes the related work, Section 4 illustrates the proposed scheme in details. Section 5 depicts the performance evaluation and the experimental results. Finally, the paper conclusion is presented in Section 6.

2. Related Work

Several approaches are focused on clustering technique and energy-aware routing protocol for MS in WSN [22]. However, most of these approaches have heavyweight processes in resolving WSN clustering and dealing with MS problems. This section offers a brief literature review of the presenting attempts for both clustering and MS-based data gathering. In Huang and Savkin's [23] study, an unequal-sized cluster-based routing protocol is presented to perform data gathering in WSNs. The proposed protocol tries to balance energy consumption across the network to extend the lifetime of all nodes as much as possible. It allows MS to go along a fixed mobility model to collect the data from the CH. Moreover, this protocol chooses relay CH for the optimum data transmission. However, this protocol also has some issues when the hot spot/energy-hole issue is close to the trajectory of the sink. In Mehto et al.'s [24] study, the authors introduce TARA, an efficient trajectory planning and route adjustment approach designed for WSN-assisted Internet of Things (IoT) environments. It addresses the challenge of efficient data transmission and collection in WSNs, where IoT devices face resource constraints such as limited energy, computing capabilities, and storage availability. TARA divides the deployment region into a uniform grid, identifying optimal rendezvous grid-cells for MS data collection. It reduces energy consumption by 10% – 18% and delay by 15% – 24% compared to state-of-the-art techniques.

Wen et al. [25] suggested an energy-aware path construction (EAPC) algorithm for WSNs. The algorithm selects a few RVPs within the network and builds a route between them. At those points, MS gathers huge amounts of data. EAPC is planned to extend the network lifetime and compute the traveling cost from one point to another. However, this algorithm suffers from excess transmission delays and network partition problems. In Wang and Chen's [26] study, an efficient path planning scheme for MS data gathering in WSNs is introduced. It aims to reliably gather data from sensors of diverse sensing rates. This scheme utilizes both hop distance and the amount of data collected by the SNs to

select RVPs within the network. This scheme defines many RVPs that considerably maximize the data gathering time and lead to a buffer overflow.

Fu and He [27] presented a balanced inter-cluster and inner-cluster energy (BIIE) algorithm for WSNs. In this method, a local reclustering mechanism is used to balance the energy consumption within each cluster based on the residual energy of its sensors. In addition, the method provides a mechanism to select a very few RVPs to serve several CHs. These preferred RVPs are used to build the MS's trip path for gathering data. However, the use of very few RVPs leads to maximizing the data gathering time and causing the buffer overflow problem. In Mehto et al.'s [28] study, squirrels search algorithm based rendezvous points selection (SSA-RVPS) is introduced for the MS to reliably acquire data from WSNs. The SSA-RVPS aims to extend the network lifetime by reducing the trajectory length of MS to gather SNs data generation with different rates. However, the SSA-RPS may implement the reselection of RVPs to ensure a balanced energy distribution among SNs. The main drawback of the SSA-RPS is the loss of data due to the buffer overflow. This occurred when RVPs received more data packets than their available buffer space.

Gutam et al. [20] offered an optimal RVPs selection method to construct MS trajectory for data collection in WSNs (ORPSTC). Initially, authors implement the minimal cost spanning tree (MST) to construct the intended clusters. Next, each CH is identified, and the RVP is also selected for each cluster. ORPSTC created an efficient trajectory for the MS using a low-computation geometry algorithm called MS trajectory construction (MSTC). Through the MS tour, virtual-RVPs (VRVPs) are defined for the SNs that have adequate communication range to connect directly to the MS. However, the VRPS communicates their data to the closest RVPs if the MS becomes out of their communication range. Despite the deduced path considering the sequence of the RVPs locations to improve the data collection, the intended path may not be the shortest one.

Agarwal et al. [29] proposed an intelligent data routing technique for WSNs based on MS for data collection. They employ particle swarm optimization (PSO) for the optimal cluster formation. Next, the RVPs are evaluated based on the average of all the X - Y coordinates of the SNs of each region. Finally, they utilize these RVPs to draw the MS path for data-gathering tour. The main disadvantage of this approach is the evaluation of the RVPs without any consideration for the actual position of the CHs. In addition, the proposed scheme is not appropriate when employing disconnected networks.

Singh et al. [30] suggested a genetic algorithm for sink mobility technique (GA-SMT). This approach partitions the entire network area into various small-sized regions and picks out RVP for each region using the GA process. Moreover, it employs a vast number of messages to handle and manage small-sized regions which maximizes the consumed energy and decreases the network lifetime. In Sahoo et al.'s [31] study, both GA and PSO are merged in a hybrid algorithm known as GAPSO-H. This algorithm is employed for routing on SM and CH selection. SM has been performed by the PSO algorithm but fails to apply the fitness parameters for routing.

In Wang et al.'s [32] study, another hybrid approach based on the PSO, and the GA is utilized to construct a path scheduling technique. This approach is based on the coverage rate of multiple MSs (TSCR-M). In TSCR-M, the RVPs are primarily established by an enhanced PSO algorithm that reflects sensor coverage and overlapped coverage rates. Next, GA is applied to select the most reasonable route for MMSs. However, the GA fails in addressing the permanence period of the network. Gowda and Jayasree [33] offered a group teaching algorithm by using the Bald Eagle (GTA-BE) routing scheme of WSNs. In this methodology, clusters are created through the mean shift clustering method. The new Bald Eagle Search mechanism is employed to select the CHs while RVPs are determined based on the number of transferred data packets and hop distance. In the end, a hybrid neural network is engaged with group teaching algorithm to select the optimal path between SNs and RVPs. The main shortcoming of this algorithm is that it endures massive MO and transmission delay.

Kumar et al. [34] offered ant colony optimization-based MS path determination (ACO-MSPD) scheme for WSNs. This scheme seeks to select the optimal CHs to meet the delay requirements and balance the energy consumption of the SNs. Moreover, It restricted the maximum touring distance of the MS and chosen the number of RVPs that did not surpass the threshold value of the MS tour. However, this scheme endures high computational complexity. Donta et al. [35] presented an extended ACO to construct MS path for event-driven WSNs. In this attempt, the maximum distance of the MS tour is fixed, and the RVPs are selected according to the SNs data generation rate. However, the RVPs selection is performed due to threshold value. In addition, each RVPs selection remaining used and changes only when the SNs data generation rate is changed. The main disadvantage of this approach is the huge amount of time consumed to select RVPs.

Gupta and Saha [36] offered a hybrid meta-heuristic algorithm-based data routing method for WSNs. Both artificial bee colony and differential evolution (ABC-DE) mechanism are employed to balance the energy spending among the CHs. Besides, a MS-based data collection was performed to gather the data from CHs. However, this algorithm suffers from a minimal convergence rate and decline of the network lifetime. Furthermore, the SNs consume very high energy, which reduces the overall performance of the networks. In Raj et al.'s [37] study, the drawback of constructing a consistent and intelligent route for MS is addressed utilizing game theory and improved ACO-based MS route choice and data gathering (GTAC-DG) approach. The MS route is structured applying an ACO-based algorithm employing the selected (RVPs). Though the GTAC-DG algorithm creates a convincing route for MS and constrains the use of multihop data transfer. But the main drawback is the ignoring of the CH selection process. Tables 1 and 2 provide an overview of the related work.

3. Network Model

In the proposed scheme, WSN topology is divided into a set of clusters. Each cluster comprises a number of SNs. As

shown in Figure 1, each SN has many features: sensing unit, processing system, and communication system. Sensing unit include sensor device and global positioning system (GPS). Sensor device is employed to sense physical event or phenomenon and then send the gathered information to the CH node. GPS is used to get the required knowledge of location with high accuracy. The processing unit embraces microcontroller unit (MCU) and memory and the required operating system. The MCU executes self-coding process and the necessary computation on the collected data when employed as a CH. Transceiver is a broadband-radio communication system to transmit the gathered data among nodes and their CH, and then between CH and MS. Finally, power unit is a battery to provide energy for SN and optional components. However, the architecture of the SNs suffers from the small size and low power, appropriate locations.

In the following subsections, assumptions related to SNs and MS are presented. Moreover, the energy consumption of the radio model applied in the intended WSN is also declared.

3.1. Network Assumptions. In the proposed work, as shown in Figure 2, N SNs are deployed randomly in a given geographical area A with width X and height Y . This area is managed by a MS, that is responsible to collect the sensory data from SNs. The following assumptions are made about the WSN under consideration:

- (1) $\mathcal{S} = \{s_i | 1 \leq i \leq N\}$ represents a set of homogeneous SNs deployed over A , where battery recharging or replacement for each SN is not probable.
- (2) Each SN $s_i \in \mathcal{S}$ has a 2D position (x_i, y_i) in the region A , which is determined after deployment using its built in GPS.
- (3) Each SN $s_i \in \mathcal{S}$ has a transmission range r and can communicate with another SN s_j if the Euclidean distance $d_{ij} \leq r$, where:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \quad (1)$$

- (4) A stationary BS with infinite energy supply collects data from the MS every round.
- (5) The area A is divided into a number of equal-sized clusters. Each cluster has a number of SNs known as CMs with each cluster having a head, called the CH, which serves as a local sink for the SNs of the cluster. The CH is used to aggregate the sensory data from the CMs.
- (6) An MS with a sufficient amount of battery life and computational capacity is used to gather data from SNs in subsequent rounds. In fact, the MS hovering over the WSN and visiting a selected set of positions called RVPs.
- (7) The MS is traveling at a constant speed c m/s during each round.

TABLE 1: Summary of related works: sink type, aggregation, metrics, and trajectory.

References	Sink type		Aggregation		Metrics	Trajectory algorithm
	MS	Fixed	RVPs	CHs		
[22]	✓	✗	✓	✓	Network lifetime, ergodic residual energy, throughput, and delay.	Threshold-based secret sharing scheme with the assistance of various criteria and QoS requirements.
[23]	✓	✗	✗	✓	Number of alive nodes, network life time, and average energy consumption.	Fixed path where MS makes three trips on its fixed path to get each node's information of its local density and its distance to MS' trajectory.
[25]	✓	✗	✗	✗	Network lifetime, fairness index, and energy consumption.	An energy-aware path construction algorithm based on minimal spanning tree.
[26]	✓	✗	✓	✓	Path length, effect of SN, effect of buffer capacity.	An efficient path planning algorithm that implemented based on Dijkstra's algorithm.
[27]	✓	✗	✓	✗	Network lifetime, network energy consumption, path length.	An efficient mechanism that applying PSO with a path encoding method to select a RVP for each cluster.
[20]	✓	✗	✓	✓	Average energy consumption, network life time, and throughput.	A minimum spanning tree-based clustering approach for RVPs selection and then use a low-computation geometry algorithm called MS trajectory construction (MSTC).
[28]	✓	✗	✓	✗	Energy consumption, network life time, number of dropped packets, buffer capacity, and data gathering ratio.	A squirrel search algorithm-based RVPs selection (SSA-RVPs) method is used to minimize the trajectory of MS while visiting a set of optimal RVPs.
[29]	✓	✗	✓	✓	Stability period, network lifetime, throughput, average energy consumption, data transmission latency, and message overhead.	Path planning mechanism consists of two phases; constructing boundary of RVPs by connecting the outer RVPs and then connecting the leftover RVPs.
[30]	✓	✗	✗	✓	Stability interval, network lifetime, throughput, and end-to-end delay.	A genetic algorithm defines the optimal RVPs on the trajectory for each cluster.
[31]	✓	✗	✗	✓	Stability period, network longevity, number of dead nodes, throughput, and remaining energy.	A hybrid of genetic and particle swarm optimization algorithms are used for CH selection and sink mobility-based data transmission.
[32]	✓	✗	✓	✗	Total energy consumption, network lifetime, network delay. Coverage and overlapped coverage rate.	An improved (PSO) combined with mutation operator is presented to search the parking positions with optimal coverage rate. Then the genetic algorithm (GA) is adopted to schedule the moving trajectory for multiple mobile sinks.
[33]	✓	✗	✓	✓	Energy consumption, network lifetime, throughput, end-to-end delay, packet delivery ratio, packet loss ratio, latency, and the jitter performance.	RVPs are chosen based on weighted evaluation of transmitted data packets and hop distance. A hybrid neural network with group teaching optimization algorithm then identifies the best path through these selected RVPs.
[34]	✓	✗	✓	✓	Energy consumption, network lifetime, and average tour length.	A novel ACO-based mobile sink algorithm seeks to determine an almost optimal set of RVPs and tour length for the mobile sink, considering nonuniform data constraints.
[35]	✓	✗	✓	✗	Energy consumption, network lifetime, buffer utilization, and average tour length.	Extended ACO-MSPD optimizes mobile sink paths in event-driven WSNs with a reselection mechanism for RVPs to balance energy and virtual RVPs to minimize data transmissions.
[36]	✓	✗	✗	✓	Energy consumption, residual energy, and network lifetime.	Proposing an artificial bee colony (ABC)-based metaheuristic for mobile sink (MS) relocation, it optimizes MS paths to balance energy consumption during communication with CHs.
[37]	✓	✗	✓	✗	Energy consumption, network lifetime, total travel time, and overlapping area.	A game theory and enhanced ant colony based MS route selection and data gathering (GTAC-DG) technique.

TABLE 2: Summary of related works: benefits and limitations.

References	Scheme/algorithm	Benefits	Limitations
[22]	An enhanced, high-performance cluster-based secure routing protocol to improve the data management quality.	Maximize network lifetime, throughput, and decrease the delay.	The real-world scalability remains unspecified.
[23]	An energy-efficient routing algorithm extends network lifespan by managing cluster sizes and energy use.	Balancing the energy consumption and decreases the overhead.	Issues arise when the energy-hole problem aligns with the sink's trajectory.
[25]	An energy-aware algorithm to minimize energy use in data collection by efficiently selecting points and distributing the workload.	Increase network lifetime, minimize energy consumption.	Suffers from excess transmission delays and network partition problem.
[26]	An efficient path planning for reliable data gathering algorithm.	Less computation overhead, reduced energy consumption, and better utilize of buffers.	Defines numerous RPs that significantly extend data gathering time and risk buffer overflow.
[27]	An energy-efficient data collection algorithm balances intercluster and intracluster energy to prolong network life.	Maximize the network lifetime by balancing energy and reduce communication costs.	Using few RVPs leads to maximizing the data gathering time and causing the buffer overflow problem.
[20]	Introducing ORPSTC for MS data collection base on a minimum spanning tree-based clustering approach for RVP selection.	Extend the network lifetime and ensure a balanced energy distribution among SNs.	The loss of data due to the buffer overflow.
[28]	An optimal rendezvous points selection method to construct MS trajectory for data collection in WSNs.	Balance the energy among the SNs and prolongs the network lifetime.	The intended path may not be the shortest one.
[29]	An intelligent data routing technique for WSNs based on MS for data collection.	Maximize stability, lifetime, and throughput; minimize energy consumption, latency, and overhead.	Evaluating rendezvous points without considering CHs' actual positions is unsuitable for disconnected networks.
[30]	A genetic algorithm (GA) is used to obtain an optimal number of clusters and a sink movement trajectory.	Minimizes the average communication distance by determining the best location for each cluster head.	Maximizes the consumed energy and decreases the network lifetime.
[31]	A hybrid approach of genetic and particle swarm optimization algorithms are used for help in optimized the CH selection and the route for sink mobility.	Obtains the best possible network performance.	Fails to apply the fitness parameters for routing.
[32]	A trajectory scheduling method based on PSO and GA is introduced to search the parking positions with optimal coverage rate.	Decrease energy and increased network lifetime.	GA fails in addressing the permanence period of the network.
[33]	A hybrid of the mean shift and Bald Eagle Search algorithms are used for clustering and selecting the cluster head.	Increase network lifetime and decrease energy consumption and delay.	Endures massive message overhead and transmission delay.
[34]	Ant colony optimization-based MS is introduced to maximize the network lifetime and minimize the delay in collecting data.	Balance the energy consumption and maximize the network lifetime.	High computational complexity.
[35]	An extended ant colony optimization (ACO)-based MS path construction for selecting the best set of the RVPs and determining the efficient MS path.	Decreasing data loss and increasing network lifetime.	Increasing the RVPs selection time.
[36]	A hybrid of artificial Bee Colony and Differential Evolution is used for routing method in WSNs.	Balance the energy among the CHs.	Minimal convergence rate and decline of the network lifetime.
[37]	A game theory and enhanced ant colony-based MS route selection and data gathering technique to manage data transfer, management, energy consumption.	Reduced energy consumption and data delivery delay.	Ignoring of the CH selection process.

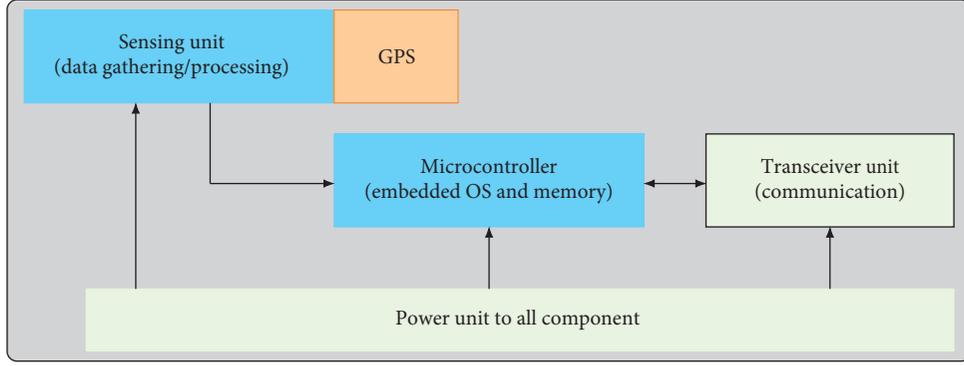


FIGURE 1: SN architecture.

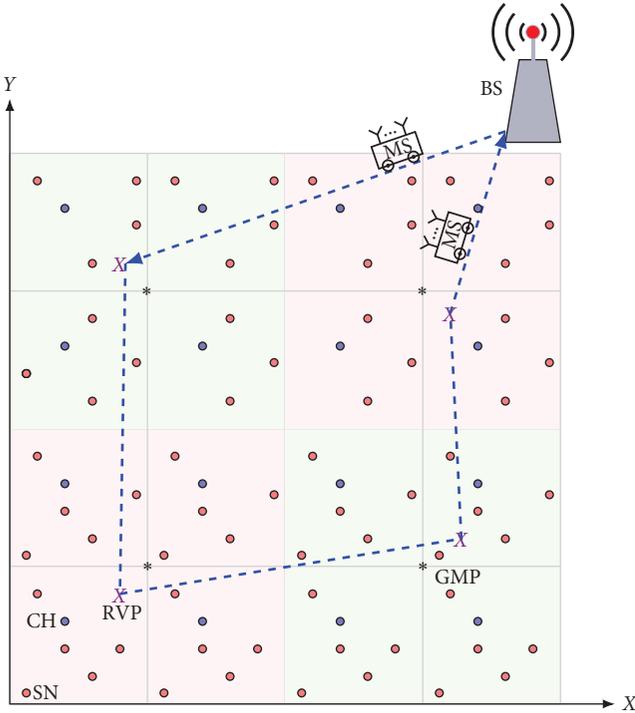


FIGURE 2: The conceptual structure of the considered network topology.

- (8) Based on the MS's position, the CHs modify their transmission range to fit within the MS's range.
- (9) All SNs have the same initial energy E_i joule. If there is not enough power remaining to transmit a packet to the CH, the SN will be deemed dead.

3.2. Energy Model. In this paper, the energy consumption of SNs is assessed using the first-order radio energy model [38]. Normally, the SN's energy is dissipated during data sensing, processing, transmission, and reception, as well as analog to digital converters. At receivers, the SNs dissipate energy for radio electronics, whereas at transmitters, they dissipate energy for radio electronics and power amplifiers. Let the transmitter or receiver dissipates E_e energy per bit in its circuit. Typically, channel coding, modulation, filtering,

and spreading have an impact on E_e . Let E_f and E_m represent, respectively, the energy needed to send a bit over a given distance in free space and a multipath fading channel. The transmission distance threshold d_0 is given as follows:

$$d_0 = \sqrt{\frac{E_f}{E_m}}. \quad (2)$$

The dissipated energy $E_{TX}(n, d_{ij})$ for transmitting n -bit over distance d_{ij} between SNs s_i and s_j is expressed as follows:

$$E_{TX}(n, d_{ij}) = \begin{cases} n \times (E_e + E_f \times (d_{ij})^2), & d_{ij} < d_0 \\ n \times (E_e + E_m \times (d_{ij})^4), & d_{ij} \geq d_0 \end{cases}. \quad (3)$$

The dissipated energy to receive a n -bit at a SN is given as follows:

$$E_{RX}(n) = nE_e. \quad (4)$$

The dissipated energy at a given CH due to aggregating n -bit from k SNs is given as follows:

$$E_{agg}(k, n) = kE_{RX}(n). \quad (5)$$

The dissipated energy at a given CH consists of three components: receiving, aggregating, and transmitting. Based on Equations (3)–(5), the total consumed energy of a given CH in each round is given as follows:

$$E_{CH} = kE_{RX}(n) + E_{agg}(k, n) + E_{TX}(p, d_{HS}), \quad (6)$$

where p and d_{HS} denoting the data payload and the Euclidean distance from a CH to the MS, respectively.

4. The Proposed Scheme

In the envisaged scheme, a significant portion of computations and communications is alleviated from both SNs and CHs, being instead transferred to the MS and BS. This scheme operates in dual modes: *configure* and *operational* modes. The *configure* mode is specifically designed to carry

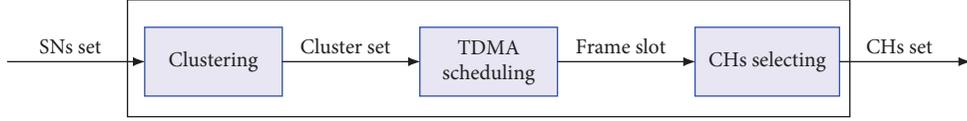


FIGURE 3: The three configure mode processes.

out the necessary calculations operations to reduce the consumed energy when the topology is running. Therefore, the necessary information are available to both BS and MS. On the other hand, the *operational* mode is employed to achieve two main objectives: performing the minimum mandatory operations for topology running and establishment of an optimal route for the MS to upload the gathered data from the CHs.

4.1. Configure Mode. The *configure* mode is a pivotal aspect of our proposed scheme, where the computational burden is centralized at the BS. This approach ensures that SNs and the MS are not overwhelmed with continuous computational tasks, thereby minimizing the impact on their resources and energy consumption. As illustrated in Figure 3, the *configure* mode comprises three consecutive processes: cluster construction, TDMA scheduling, and the selection of CHs.

4.1.1. Cluster Construction. This process presents a novel method of portioning the monitoring geographical area A into an *even* number of equal-size clusters, where w and h denoting width and height of any generated cluster. The BS organizes the generated clusters in rows and columns, where the number of clusters in each row and column must be even. As shown in Figure 4, initially, the BS assumes that the initial height h_I and width w_I of each cluster is calculated based on the transmission range R as follows:

$$w_I = \sqrt{0.5R}, \quad (7)$$

and

$$h_I = \sqrt{R^2 - w_I^2}. \quad (8)$$

To ensure that the number of generated clusters is even in each row and column directions, the BS calculates h and w for each cluster based on Equations (7) and (8). Let n_1 be an integer number given as follows:

$$n_1 = \left\lfloor \frac{X}{w_I} \right\rfloor, \quad (9)$$

where $\lfloor x \rfloor$ denotes the floor of x . Let B_1 be a binary variable defined as follows:

$$B_1 = \begin{cases} 1, & \text{if } \left(\frac{X}{w_I} - n_1 \right) = 0 \\ 0, & \text{otherwise} \end{cases}. \quad (10)$$

Using Equations (7)–(10), the value of w is recalculated as follows:

$$w = \begin{cases} w_I + \left\lfloor \frac{[X - (n_1 \times w_I)]}{n_1} \right\rfloor, & n_1 \text{ even} \\ w_I - \left(\frac{B_1 \times w_I}{n_1} \right) - \left\lfloor \frac{((1 - B_1)(w_I - [X - n_1 w_I]))}{(n_1 + 1)} \right\rfloor, & n_1 \text{ odd} \end{cases}. \quad (11)$$

The term $\left(\frac{B_1 \times w_I}{n_1} \right)$ reduces the width of each cluster for making room to add another cluster. Equation (11) guarantees an even number of clusters per row. In similar way, the number of clusters in each column can be adjusted to even number as follows. Let n_2 be an integer number given as follows:

$$n_2 = \left\lfloor \frac{Y}{h_I} \right\rfloor. \quad (12)$$

Let B_2 be a binary variable defined as follows:

$$B_2 = \begin{cases} 1, & \text{if } \left(\frac{Y}{h_I} - n_2 \right) = 0 \\ 0, & \text{otherwise} \end{cases}. \quad (13)$$

Using Equations (8), (12), and (13), the value of h is recalculated as follows:

$$h = \begin{cases} h_I + \left\lfloor \frac{[Y - (n_2 \times h_I)]}{n_2} \right\rfloor, & n_2 \text{ even} \\ h_I - \left(\frac{B_2 \times h_I}{n_2} \right) - \left\lfloor \frac{((1 - B_2)(h_I - [Y - n_2 h_I]))}{(n_2 + 1)} \right\rfloor, & n_2 \text{ odd} \end{cases}. \quad (14)$$

This leads to an even number within each column for constructed clusters. The total number of generated clusters M within the area A is given as follows:

$$M = \begin{cases} n_1 \times n_2, & n_1 \text{ even}, n_2 \text{ even} \\ n_1 \times (n_2 + 1), & n_1 \text{ even}, n_2 \text{ odd} \\ (n_1 + 1) \times n_2, & n_1 \text{ odd}, n_2 \text{ even} \\ (n_1 + 1) \times (n_2 + 1), & n_1 \text{ odd}, n_2 \text{ odd} \end{cases}. \quad (15)$$

Its clear that $\mathcal{A} = M \times h \times w$. Let $\mathcal{C} = \{C_i | 1 \leq i \leq M\}$ be the set of generated clusters from the cluster construction process. After clusters are formed, start from left to right and up to bottom of the clusters topology, each four adjacent clusters forms a group, as shown in Figure 4. Let $\mathcal{G} = \{G_i | 1 \leq i \leq \frac{M}{4}\}$ be the set of the generated groups, where

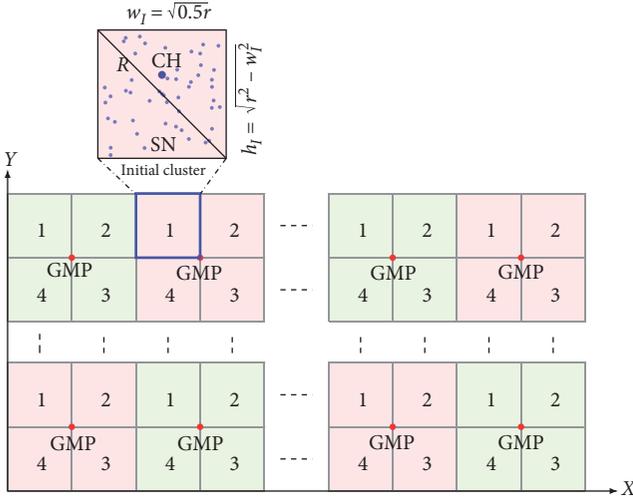


FIGURE 4: Square-shaped clusters arranged into $\frac{M}{4}$ groups. Each group is made up of four neighboring clusters with numbers from 1...4.

$G_i = \{C_{ik} | 1 \leq k \leq 4\}$ and C_{ik} denoting the cluster number k within group G_i . Each $C_{ik} \in G_i$ is with N_{ik} SNs. It is clear that:

$$N = \sum_{i=1}^{\frac{M}{4}} \sum_{k=1}^4 N_{ik}. \quad (16)$$

The vertices meeting point of the 4-clusters within each group represent the group meeting positions (GMPs), where m_i represents the GMP of the group $G_i \in \mathcal{G}$.

4.1.2. TDMA Schedule. Originally, WSNs employ broadcast communication in which multiple devices may emit signals simultaneously, leading to collisions and signal destruction. To address this issue, TDMA is employed as a scheduling algorithm to coordinate a group of SNs in transmitting their data within a predetermined frame. This frame is partitioned into equal time slots, allocating one slot to each SN for transmission [15].

The proposed scheme intended to reduce the sources of energy loss and reducing the gap of residual energy among all SNs in each cluster. As aforementioned, each $C_{ik} \in G_i$ has N_{ik} SNs deployed in its area, where one of them is selected as a CH as illustrated in the next section. At the BS, a TDMA scheduling is modified to define a frame f_{ik} of equal time slots for each cluster $C_{ik} \in G_i$, where $i = 1, 2, 3, \dots, \frac{M}{4}$ and $k = 1, 2, 3, 4$. The number of time slots in each f_{ik} is equivalent to the number of SNs N_{ik} , where each slot is associated with only one SN $s_j \in C_{ik}$ ($j = 1, 2, 3, \dots, N_{ik}$). This means the frame's length of each cluster is different according to its number of SNs. On the operational mode and for each round, the SN $s_j \in C_i$ has one chance to transmit its sensed data to the CH according to its order in the frame f_i . In some cases, the frame length is greater than the number of active SNs in the cluster. This case has happened when some of the SNs within its cluster are dead. In the traditional TDMA scheduling, when the frame length is greater than the number of SNs, some of them take more than one chance to transmit their data in the

same round. In such a case, the power of SN is drained quickly. To prevent the SN from sending data more than once, the SN $s_j \in C_{ik}$ switches to sleep mode once it sent its data within its associated slot and then switches to wakeup mode in the next round. Once the TDMA scheduling frame f_{ik} is formed at the BS, the BS broadcasting f_{ik} to all SNs within C_{ik} at the first round only.

4.1.3. Cluster Head Selection. Up to best literature review, the computation processing to select the optimal CH is generally performed among the SNs of each cluster. So, the SNs are exposed to lose some of a significant energy due to the high overhead communication between them. The proposed scheme introduces a novel approach that breaks the CH selection process for each cluster into two stages. The first stage is implemented initially at the BS and contains the most computation processing of the CHs selection, The second stage is implemented during the operational mode.

In the first stage, the BS executes the CH scheduling Algorithm 1 for all clusters in parallel processing. In this algorithm, for each group $G_i \in \mathcal{G}$ and for each cluster $C_{ik} \in G_i$, the distance d_{ij} between SN $s_j \in C_{ik}$ ($j = 1, 2, 3, \dots, N_{ik}$) and the GMP m_i is computed as follows:

$$d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}, \quad (17)$$

where (x_j, y_j) and (x_i, y_i) denote the positions of the SN $s_j \in C_{ik}$ and the GMP m_i . The intradistances among the SN $s_j \in C_{ik}$ and remaining SNs within the cluster C_{ik} is given as follows:

$$d_{jw} = \sqrt{(x_j - x_w)^2 + (y_j - y_w)^2}, \quad (18)$$

where $1 \leq w \leq N_{ik}$ and $w \neq j$. For SN $s_j \in C_{ik}$, let d_j^T the total calculated distances for SN $s_j \in C_{ik}$ which given based on Equations (17) and (18) as follows:

$$d_j^T = d_{ij} + \sum_{w=1, w \neq j}^{N_{ik}} d_{jw}. \quad (19)$$

Construct the cluster descending order vector \mathbf{E}_{ik} cosponsoring to the SNs within the cluster $C_{ik} \in G_i$:

$$\mathbf{E}_{ik} = ((ID_1, e_2), (ID_2, e_2), (ID_3, e_3), \dots, (ID_{N_{ik}}, e_{N_{ik}})), \quad (20)$$

which orders the SNs within its cluster based on their total distances d_j^T given in Equation (19). In particular, $e_j = 1$ means the SN $s_j \in C_{ik}$ with identification ID_j has first smallest total distance d_j^T . $e_j = 2$ means the SN $s_j \in C_{ik}$ with identification ID_j has the second smallest total distance d_j^T . The order variable e_j within the vector \mathbf{E}_{ik} represents the order of SN $s_j \in C_{ik}$ to work as a CH. Algorithm 1 illustrates the formulation of the CH selection order vector \mathbf{E}_{ik} in the first stage. Finally, the BS constructs a message containing the

```

1: Execute the cluster construction process
2: for Each group  $G_i \in \mathcal{G}$  do
3:   for Each cluster  $C_{ik} \in G_i$  do
4:     for Each SN  $s_j \in C_{ik}$  do
5:       Calculate  $d_{ij}$  using 15
6:     for Each  $s_w \in C_{ik}$  and  $w \neq j$  do
7:        $sum = sum + d_{ij}$  using 16
8:     end for
9:     Set  $d_j^T = sum + d_{ij}$ 
10:    push the tuple  $(ID_j, d_j^T)$  into vector  $\mathbf{E}_{ik}$ .
11:    Sort the vector  $\mathbf{E}_{ik}$  in descending order.
12:    for  $j = 1$  to  $N_{ik}$  do
13:      labelling the second item of the pair  $j$  in  $\mathbf{E}_{ik}$  as  $e_j$ .
14:    end for
15:  end for
16:  Send the vector  $\mathbf{E}_{ik}$  to cluster  $C_{ik} \in G_i$ .
17: end for
18: end for

```

ALGORITHM 1: The BS scheduled CH process.

vector \mathbf{E}_{ik} and sends it to all SNs within the cluster $C_{ik} \in G_i$. For $C_{ik} \in G_i$, the SN $s_j \in C_{ik}$ is selected to be as a CH in the first round if $e_j = 1$, where $i = 1, 2, 3, \dots, \frac{M}{4}$, $k = 1, 2, 3, 4$, and $j = 1, 2, 3, \dots, N_{ik}$. Actually, the scheduled CH Algorithm 1 gives each SN a chance to be a CH at specific round, where a SN is selected as a CH based on the mapping between vector \mathbf{E}_{ik} and round number. This relation will be clarified and explained through the sensor's self-encoding module (SEM) in the *operational* mode which represents the second stage of the CH selection process.

4.2. Operational Mode. This mode is executed in successive *global* round $R = 1, 2, 3, 4, \dots$. When activated, the network components are waking up to perform the required operations for the data routing trip from each SNs up to BS. This trip consists of CH activation process and the MS trajectory planning process.

4.2.1. CH Activation Process. Initially and only once, for each $C_{ik} \in G_i$, the BS send a set of wakeup messages to all SNs within C_{ik} . This is the first communication message that the SNs have gotten. The wakeup message composed of two items: the TDMA scheduling frame f_{ik} and the CH selection order vector \mathbf{E}_{ik} which are formed in the *configure* mode. The values of f_{ik} and \mathbf{E}_{ik} are employed by the SEM of all SNs within C_{ik} to perform the routing operation. In addition, each SEM fortified by a dedicated *local* round variable R_{ik} that defined as the current round which used to count the number of executed rounds within C_{ik} . Initially, the *local* round $R_{ik} = 0$ for all SN within the cluster $C_{ik} \in G_i$. When the *global* round R begins (i.e., $R = 1$), R_{ik} is set to 1 by the SEM of all SNs in C_{ik} .

Based on the vector \mathbf{E}_{ik} , the SN $s_j \in C_{ik}$ elects itself as a CH if $e_j = R_{ik}$. The election process is executed without any communications between the SNs themselves or the SNs and

```

1: Execute the cluster construction process
2: for each group  $G_i \in \mathcal{G}$  do
3:   for each cluster  $C_{ik} \in G_i$  do
4:     for  $R_{ik} = 1$  to  $N_{ik}$  do
5:       for  $j = 1$  to  $N_{ik}$  do
6:         if  $e_j = R_{ik}$  then
7:           Select  $s_j$  as CH
8:           break
9:         end if
10:      end for
11:      Aggregates data at  $s_j$ 
12:      Send data to the MS.
13:      Switch  $s_j$  to member state.
14:      if  $E_j \leq T_e$  then
15:        Set  $s_j$  to sleep mode
16:        Set the slot of  $s_j$  to unused mode
17:        for  $r = e_j + 1$  to  $N_{ik}$  do
18:          Set  $e_r = e_r - 1$ 
19:        end for
20:      end if
21:    end for
22:  end for
23: end for

```

ALGORITHM 2: CH activation algorithm.

their CHs. As a result, the energy consumption of the network is significantly reduced. As shown in Algorithm 2, the CH selection is carried out during the *operational* mode as follows:

- (1) Step 1: If the value of $e_j \in \mathbf{E}_{ik}$ of a given SN $s_j \in C_{ik}$ is equal to the current round R_{ik} then it employs itself as a CH while remaining SNs within C_{ik} employ themselves as CMs.
- (2) Step 2: After the current CH of C_{ik} collects the sensory data from its CMs and sends it to the MS, it switches itself to be a CM in the next round.
- (3) Step 3: If the current CH is the last SN within the vector \mathbf{E}_{ik} then all SNs within C_{ik} including the CH itself will reset the value of R_{ik} as follows:

$$R_{ik} = \begin{cases} |f_{ik}|, & \text{if } (R \bmod |f_{ik}|) = 0 \\ R \bmod |f_{ik}|, & \text{if } (R \bmod |f_{ik}|) \neq 0 \end{cases}, \quad (21)$$

where $|x|$ is the cardinality of x . All SNs are then follow Step 1 again.

- (4) Step 4: If the current CH is not the last SN within the vector \mathbf{E}_{ik} then all SNs within C_{ik} including the CH itself will set increase the value of R_{ik} by 1 and follow Step 1 again.

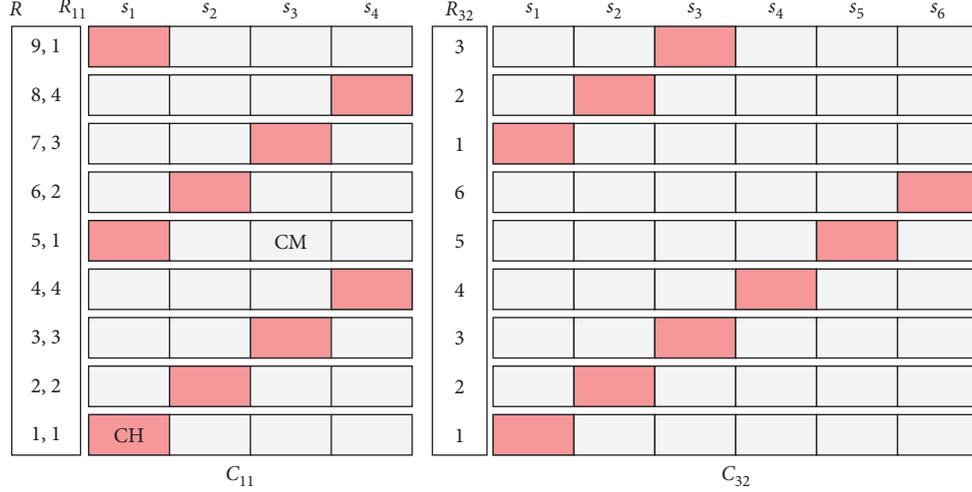


FIGURE 5: Example of two clusters from different groups in operational mode. $C_{11} \in G_1$ with four SNs and frame with length four slots. $C_{32} \in G_3$ with six SNs and frame with length six slots.

After reparations of numerous rounds, one or more of SNs may lose the majority amount of its energy. Let T_e be threshold energy that is enough to enable a SN to work as a CH. The remaining energy of the SN $s_j \in C_{ik}$ at the start of a given round is denoted as E_j . If E_j is less than T_e then the SN $s_j \in C_{ik}$ sends an announce message to all SNs to tell them about its energy status. Based on this message, the following actions are executed:

- (1) The SN $s_j \in C_{ik}$ switches to the sleep mode and its assigned slot changes to unused slot mode. Any slot in unused mode can not be used by the other SNs.
- (2) All SNs which coming after s_j in the vector \mathbf{E}_{ik} will decrease its order e_j CH by 1.

To make the CH activation process more understandable, we provide the following illustrative example as shown in Figure 5. In this example, we focus on cluster $C_{11} \in G_1$ and cluster $C_{32} \in G_3$. Four SNs are deployed in C_{11} (i.e., $N_{11} = 4$) and six SNs are deployed C_{32} (i.e. $N_{32} = 6$). The TDMA scheduling frames $f_{11} = 4$ slots and $f_{32} = 6$ slots. Initially, global round $R = 1$, the local round $R_{11} = 1$ and local round $R_{32} = 1$. The vectors \mathbf{E}_{11} and \mathbf{E}_{32} are given as $\mathbf{E}_{11} = [s_1, s_2, s_3, s_4]$ and $\mathbf{E}_{32} = [s_1, s_2, s_3, s_4, s_5, s_6]$. For the first round, the SNs $s_1 \in C_{11}$ and $s_1 \in C_{32}$ are selected as CHs. For the second round (i.e., $R = 2$, $R_{11} = 2$, and $R_{32} = 2$), the SNs $s_2 \in C_{11}$ and $s_2 \in C_{32}$ are selected as CHs. This process continues until round 5, $R_{11} = [5 \bmod 4] + 1 = 1$ and $R_{32} = 5$. In such cases, the SNs $s_1 \in C_{11}$ and $s_5 \in C_{32}$ are selected as CHs. At $R = 7$, $R_{11} = 3$, and $[7 \bmod 4] + 1 = 1$. In such case, the SNs $s_3 \in C_{11}$ and $s_1 \in C_{32}$ are selected as CHs and so on.

4.2.2. Path Planning for MS. In the planned topology, the GMPs are defined at the vertices meeting point of every four adjacent clusters. The GMPs are positioned to accomplish three purposes: limiting the transmission range of any cluster's sensors from exceeding GMP position, constructing the trajectory of MS, and identifying initial locations of the RVPs which represent the MS data collection positions. Originally,

an efficient plan of the MS path should minimize total energy utilization. This can be achieved if the transmission ranges of the CHs to MS are also reduced. However, the employment of some GMPs as RVPs positions may not be the appropriate choice for all successive rounds due to the change of the CHs positions. So, some extra points may be needed to replace some of not appropriate GMPs. In the following, these extra points are created according to the need of them.

(1) *The RVPs Points Formulation.* As mentioned before, MS has infinite energy supply and has processing power like BS. In addition, MS is aware of the necessary information about the formed clusters, their included SNs, and the initial sorting of CHs for all clusters. So, MS employs this information to test the suitability of the presented GMPs to continue as RVPs locations or replaced by more appropriate new locations. Let $\mathcal{R} = \{m_1, m_2, \dots, m_i\}$ be the initial RVP locations sequence that the MS follows to collect data from CHs, where $i = 1, 2, 3, \dots, \frac{M}{4}$. At the start of each round R executes the following procedure to update the sequence \mathcal{R} as follows. For each group, $G_i \in \mathcal{G}$ do the following steps.

- (1) Determine the centroid $c_i^H = (x_i^H, y_i^H)$ of the CHs within the group G_i :

$$x_i^H = \frac{x_{i1}^H + x_{i2}^H + x_{i3}^H + x_{i4}^H}{4}, \quad (22a)$$

and

$$y_i^H = \frac{y_{i1}^H + y_{i2}^H + y_{i3}^H + y_{i4}^H}{4}, \quad (22b)$$

where (x_{ik}^H, y_{ik}^H) is the CH position of the cluster $C_{ik} \in G_i$ ($k = 1, 2, 3, 4$).

- (2) Compute the sum of distances D_i among the four CHs and the centroid c_i^H as follows:

- 1: **Input:** The set \mathcal{R} , the number of clusters K .
- 2: **Output:** The MS path trajectory \mathbf{P} .
- 3: Divide \mathcal{R} into K clusters using constrained- K -means algorithm.
- 4: Initialize $\mathbf{P} = \{ \}$.
- 5: **for** $i = 1$ to K **do**
- 6: Apply GA algorithm to generate the optimal sub path p_i of cluster i .
- 7: Add p_i to \mathbf{P} .
- 8: **end for**
- 9: **for** $i = 1$ to $K - 1$ **do**
- 10: Connect the last RVP of the path p_i to the start of RVP of the path p_{i+1} .
- 11: **end for**

ALGORITHM 3: The MS path trajectory algorithm.

$$D_{1_i} = \sum_{k=1}^4 \sqrt{(x_i^H - x_{ik}^H)^2 + (y_i^H - y_{ik}^H)^2}. \quad (23)$$

- (3) Compute the sum of distances D_{2_i} among CHs and the GMP m_i as follows:

$$D_{2_i} = \sum_{k=1}^4 \sqrt{(x_i - x_{ik}^H)^2 + (y_i - y_{ik}^H)^2}, \quad (24)$$

where (x_i, y_i) denoting the position of the GMP m_i .

- (4) If $D_{1_i} < D_{2_i}$ then replace m_i within the sequence R with the location c_i^H .
- (5) If $D_{2_i} \leq D_{1_i}$ then keep m_i within the sequence R without any change.

Since the sequence \mathcal{R} is performed by the MS at every round, its value may be changed from round to another.

(2) *The MS Path Optimization.* The path trajectory starts from the BS, passes through all RVPs \mathcal{R} , and finally backs to the BS. For this purpose, the optimization GA is employed to select the shortest path among all available path traversing all RVPs. The MS path trajectory algorithm takes the sequence \mathcal{R} of all RVP as input and gives a path trajectory \mathbf{P} that traverses all RVPs in a shortest path manner as output. As shown in Algorithm 3, the process of constructing the MS path trajectory proceeds as follows:

- (1) Divided the set \mathcal{R} into an arbitrary number $K < \frac{M}{4}$ of clusters, using the constrained- K -means algorithm.
- (2) Apply the GA algorithm to obtain the shortest path p_i within each cluster i , where $i = 1, 1, 3, \dots, K$.
- (3) Connect the resulting subpaths $p_1, p_2, p_3, \dots, p_K$ to obtain the MS path trajectory \mathbf{P} .

4.3. Computational Complexity Analysis. For N SNs, M clusters, $G_i (1 \leq i \leq \frac{M}{4})$ groups, four cluster per group and N_{ik} SNs for cluster k within group i , the time complexity of Algorithm 1 is given as follows. In Step 2, the time complexity is $\mathcal{O}(\frac{M}{4})$, considering that the number of groups is $\frac{M}{4}$. In Step 3, the time complexity is $\mathcal{O}(\frac{M}{4} \times 4) = \mathcal{O}(M)$, as each group contains only

four clusters. The time complexity in Steps 4, 5, 9, 10, and 11 is given as $\mathcal{O}(\frac{M}{4} \times 4 \times N_{ik}) = \mathcal{O}(M \times N_{ik})$. For Steps 6 and 7, the time complexity is given as $\mathcal{O}(M \times N_{ik} \times (N_{ik} - 1))$. The time complexity of steps from 12 and 13 is given as $\mathcal{O}(M \times N_{ik} \times N_{ik})$. Therefore, the time complexity of Algorithm 1 is expressed as $\mathcal{O}(M \times N_{ik} \times N_{ik})$. The time complexity of the CH activation in Algorithm 2 is the same as in Algorithm 1.

The time complexity of the MS path trajectory in Algorithm 3 can be determined through the following calculation. The set of RVPs, denoted as \mathcal{R} , is partitioned into K clusters, where $K \leq \frac{M}{4}$, utilizing the constrained- K -means clustering algorithm in Step 2. Let n_i be the number of RVPs in cluster i and $i = 1, 2, 3, \dots, K$. It is clear that $L = \sum_{i=1}^K n_i$, where L denotes the number of RVPs in the set \mathcal{R} , $n_i \leq \frac{M}{4}$ and $M \ll N$. The loop (Steps 6–8) in Algorithm 3 iterates through each cluster i and employs the GA to determine the shortest path among the RVPs within that specific cluster. Given the input size n_i RVPs, the population size P and the number of generations G , the time complexity is given in Chatterjee et al.'s [39] and Srinivas and Patnaik's [40] studies, as $\sum_{i=1}^K \mathcal{O}(P \times G \times n_i)$. For Steps 9 and 10, the time complexity is given as $\mathcal{O}(K)$.

In our proposed scheme, we emphasize the significance of the configure mode, a pivotal aspect that plays a crucial role in the overall system operation. The configure mode is designed to centralize the computational burden at the BS. By adopting this approach, we ensure that SNs and the MS are not overwhelmed with continuous computational tasks. This strategic distribution of computational load minimizes the impact on the resources and energy consumption of SNs and the MS, thereby enhancing the overall efficiency and sustainability of our proposed solution.

5. Experimental Results

In this section, various effective metrics are employed through simulation experiments to assess the performance of the proposed scheme. So, the experiments result of the presented scheme compared with the prevailing state-of-the-art algorithms, such as EEMSR [29], EAPC [25], BIIE [27], GA-SMT [30], and GTA-BE [33]. Both the proposed scheme and the state-of-art algorithms implemented using Python

TABLE 3: Control parameters and their values.

Parameter	Value
Network type	WSN
Number of sensors	100–500 SNs
Number of MS	1
Physical medium wireless	MAC layer 802.11
MAC protocol	Modified TDMA
Transmission range	25 m
MS speed, v	2 m/s
Propagation radio model	Two way ground
Monitoring area	$100 \times 100 \text{ m}^2$
Initial energy of a sensor node	0.5 J
E_c	50 nJ/bit
E_f	10 pJ/bit/m ²
E_m	0.0013pH/bit/m ⁴
E_{agg}	5 nJ/bit
Data packet size	512 bytes
Control packet size	64 bytes
BS coordinates	(100,100)

3.11.0 mounted on Microsoft Windows 10 Pro with Intel Core i7 CPU of 4.7 GHz and 16 GB RAM. The parametric values of the state-of-the-art algorithms used as published in their papers and validating the results by comparing them with their outcome. In addition, the comparison among these algorithms and the offered scheme performed based on a variety of performance metrics such as stability period, network lifetime, average energy consumption, communication MO, data transmission latency (DTL), and throughput. Moreover, the comparison was performed with various node densities. All the SNs randomly distributed in the observing area $A = 100 \times 100 \text{ m}^2$. The BS placed at location (100, 100). The experiments run 100 times for each metric. The average results of each metric are plotted in a specific graph as in the following subsection. Table 3 reveals the additional simulation parameters considered in the experiments.

5.1. Stability Period. The stability period is measured based on the total number of data gathering rounds performed before any SN's residual energy reaches zero. According to the implemented methodology, the power consumption by each SN within each cluster is approximately equivalent. As a result, all clusters' SNs will remain doing their role together for extended period of time. This means the whole network continues functional for longer duration. As a result, the network performance is enhanced as the stability period is prolonged.

Figure 6 presents a comparison of the stability periods versus different numbers of SNs. When utilizing 100 sensors, the stability period of the proposed scheme outperforms EEMSR by 3.89%, GTA-BE by 43.9%, BIIE by 51.94%, GA-SMT by 58.62%, and EAPC by 67.69%. However, as the number of employed SNs increases, the results of the proposed scheme exhibit significantly greater improvements compared to the outcomes of the compared algorithms. Specifically, when employing 500 SNs, the stability period of the proposed scheme increases to 42.75%, 68.18%, 76.67%,

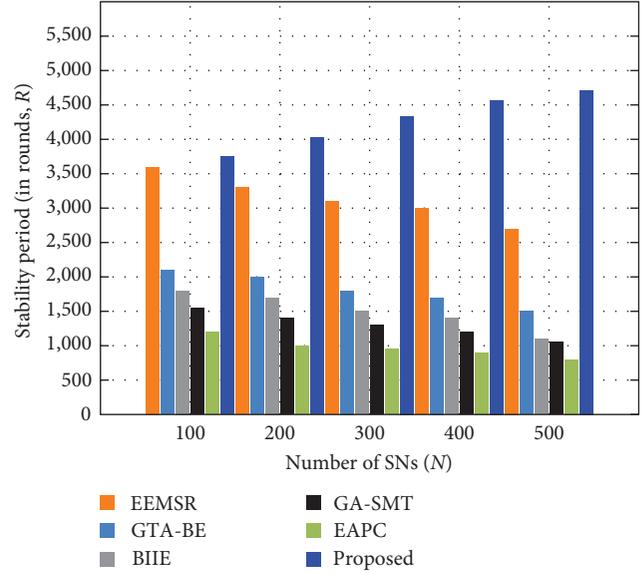


FIGURE 6: Stability period when first SN is death.

80.19%, and 3.03% when compared with EEMSR, GTA-BE, BIIE, GA-SMT, and EAPC, respectively. These results are deemed valid because an increase in the number of SNs in the presented scheme leads to a rise in the number of SNs within each cluster. Consequently, each SN will assume the role of a CH after an excessive number of rounds, allowing them to retain residual energy for more rounds. Additionally, CH selection is a self-coding process performed without the need for overhead messages or mutual processing among the cluster's members. As a result, the proposed scheme is more efficient in avoiding the premature death of SNs and provides extended stability compared to other state-of-the-art algorithms.

5.2. Network Lifetime. Figure 7 illustrates the achieved network lifetime for the proposed scheme compared with EEMSR, GTA-BE, BIIE, GA-SMT, and EAPC, respectively. The X-axis symbolizes the different number of SNs that participate in the execution of each experiment. The Y-axis specifies the overall network lifetime steadiness of individual schemes. For 100 sensors, the simulation results show that the presented scheme enhances the network lifetime up to 0.1, 49.3, 73.18, 74.81, 81.68, and 84.46 when compared with EEMSR, GTA-BE, BIIE, GA-SMT, and EAPC, respectively. When 500 sensors are deployed, the mentioned results are raised to 49.3%, 73.18%, 74.81%, 81.68%, and 84.46% for EEMSR, GTA-BE, BIIE, GA-SMT, and EAPC, respectively. The massive results differences are achieved due to an increase in the amount of energy assigned to the excessive sensors. The recorded results for employing 500 sensors show massive differences as compared to the use of 100 sensors. This improvement is achieved due to an increase in the amount of energy offered by the 500 sensors. In addition, the offered scheme performs three significant operations to enhance the network lifetime: the modified TDMA scheduling, self-coding technique of each SN, and the optimal routing algorithms using a MS. For each round, the modified TDMA limits the time of sensing

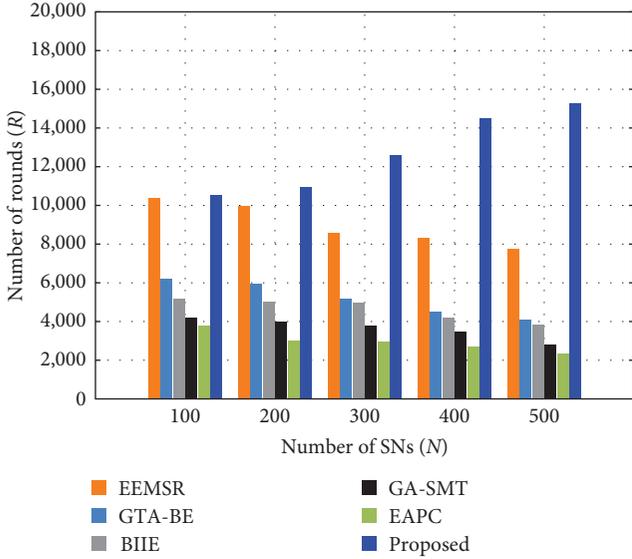


FIGURE 7: Network lifetime (i.e., the number of rounds until residual energy become zero).

operation of each SN to one definite slot. This leads to save the energy consumption due to the repeating use of some sensors as a CH more than once in the same round as in the original TDMA. Moreover, the employment of each sensor to be a CH at a specific round is prescheduled in the configure mode. At the start of each round, each sensor executes CH activation algorithm to know its role in the current round, CH or CM. Accordingly, a lot of energy consumed in the compared schemes due to overhead communication among CMs to select CH is avoided. Consequently, both modified TDMA and self-coding algorithm are participated in saving a lot of energies lost in the compared algorithms. Finally, the optimum selection of the RVPs leads to the minimization of the CHs' distances when transmitting their aggregated data to MS. As a result of the implemented procedures in the mentioned operations, the consumed power is reduced, and hence network lifetime is expanded. So, the proposed scheme is more effective in extending network lifetime than the use of the remaining state-of-the-art algorithms.

5.3. Average Energy Consumption. The average energy consumption of the network E_N evaluated based on the energy spending E_j of each SN $s_j \in C_{ik}$ every round R . To reduce the entire energy consumption, the energy consumption of each SN should be balanced during the network lifetime. The E_N can be deduced as follows:

$$E_N = \frac{\sum_{R=1}^{R_T} \sum_{i=1}^{|\mathcal{G}|} \sum_{k=1}^4 \sum_{j=1}^{N_{ik}} E_{jk}}{N}. \quad (25)$$

Figures 8 and 9 depict the comparison of average energy consumption among the proposed scheme with other state-of-the-art algorithms. For 100 SNs, Figure 8 reveals that the suggested scheme reduces the average energy consumption by 32.56%, 56.42%, 66.53%, 72.93%, and 75.3% compared to

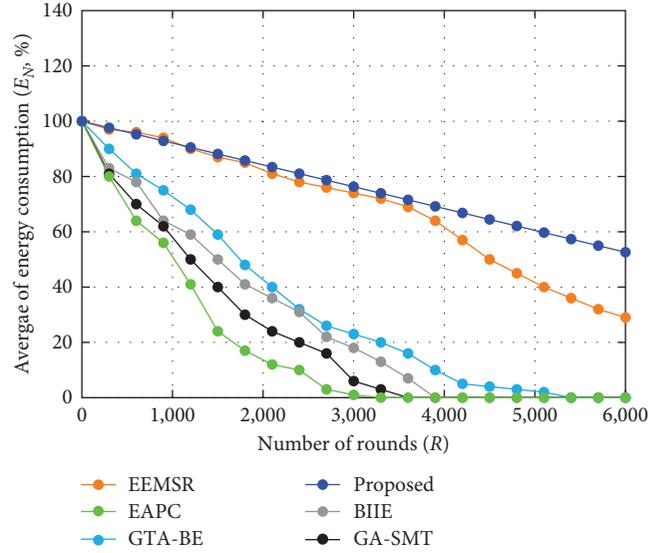


FIGURE 8: Average of network energy consumption, E_N , for $N = 100$ SNs.

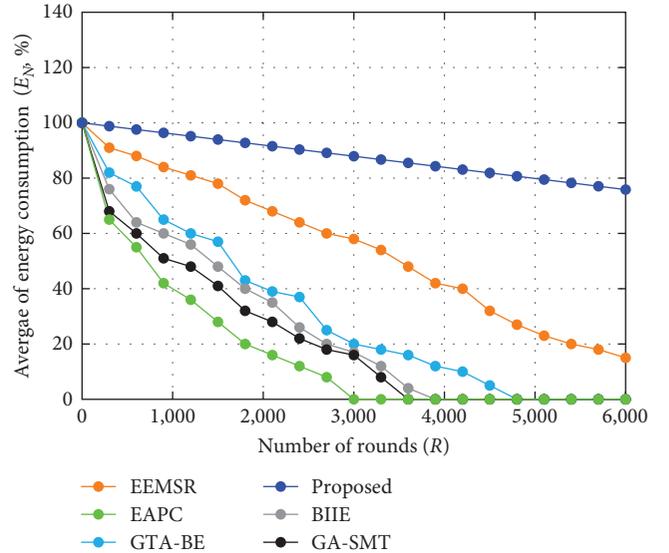


FIGURE 9: Average of network energy consumption, E_N , for $N = 500$ SNs.

EEMSR, GTA-BE, BIIE, GA-SMT, and EAPC, respectively. For 500 SNs, Figure 9 indicates that the suggested scheme reduces the average energy consumption by 58.55%, 71.79%, 78.84%, 83.45%, and 88.93% compared to EEMSR, GTA-BE, BIIE, GA-SMT, and EAPC, respectively.

The increased number of SNs leads to an increase in the amount of energy offered due to excess sensors. According to the energy model presented in Section 3.2, the dissipated energy of each SN is maximized when it plays the role of the CH. According to the scheduled CH Algorithm 1 of the proposed scheme, each SN is given a chance to be a CH at specific round. In addition, this chance does not repeat until the all-remaining SNs are given the same chance. Consequently, the energy consumption of each SN will be saved when its role as a CH is delayed. This is achieved when the

numbers of SNs are increased within each cluster. In addition, CH selection is performed based on the self-coding of each SN. In addition, CH selection is performed based on the self-coding of each SN. This means that there are no overhead communication performed due to the CH selection. Avoiding the overhead communications leads to tiny power consumption as compared with the other algorithms. Finally, the optimum selection of the RVPs leads to the minimization of each CH's distances when transmitting its aggregated data to MS. This distance's minimization leads to the minimization of power consumption due to the aggregated data transmission. As a result, the implementation of the proposed method has succeeded in presenting different techniques to avoid a lot of energy consumed when the other compared methods are applied. Hence the proposed scheme is more effective in extending network lifetime than the use of the remaining state-of-the-art algorithms. However, the enhanced energy efficiency is opposed by some increasing in the latency of the routed data. This increase is caused due to increase the time of gathering data from excess sensors to CH then transmitting to MS. However, this drawback can be avoided by exploiting more than MSs to minimize the data routing latency.

5.4. Message Overheads. MO is defined as the number of control messages transmitted among the employed SNs, MS, and BS through the network configuration and operational modes. The amount of this metric should be minimized. The increase of these messages leads to the increase of the collision and power consumption of the deployed SNs. Two communication overhead messages are employed in the proposed scheme. First, in the configure mode, the BS transmits a single message to all SNs contains the TDMA frame and the vector of the CH order for each cluster. Second, in the operational mode, the SN which loses its ability to be a CH sends a single message to all the CMs in its cluster. So, the total number of the overhead messages within the network is reduced as compared by the remaining state-of-the-art algorithms. Figure 10 reveals the comparison between the presented scheme and these state-of-the-art algorithms in terms of the MOs. The MO reduced up to 29% as compared to EEMSR, up to 38.4% as compared to GTA-BE, up to 38.72% as compared to BIIE, up to 44.65% as compared to GA-SMT, and up to 47.75% as compared to EAPC. This decline occurs due to the one-time network topology established process and scheduling algorithm of the CHs selection. This leads to the minimization of the excessive message exchange in the networks.

5.5. Data Transmission Latency. DTL is the time needed to transmit data from SNs to BS through the MS. It is computed by summation of the overall time mandatory to visit each RVP hovering locations and gathering the data from each CH. So, selecting the optimal number of RVPs leads to reducing the DTL. As a result, the reliability and overall performance of the networks are improved.

Let v denoting the MS speed and T_{H_i} denoting the MS hovering time at RVP location $r_i \in \mathcal{R}$. The data transmission latency, DTL, consists of two components: (1) the overall MS

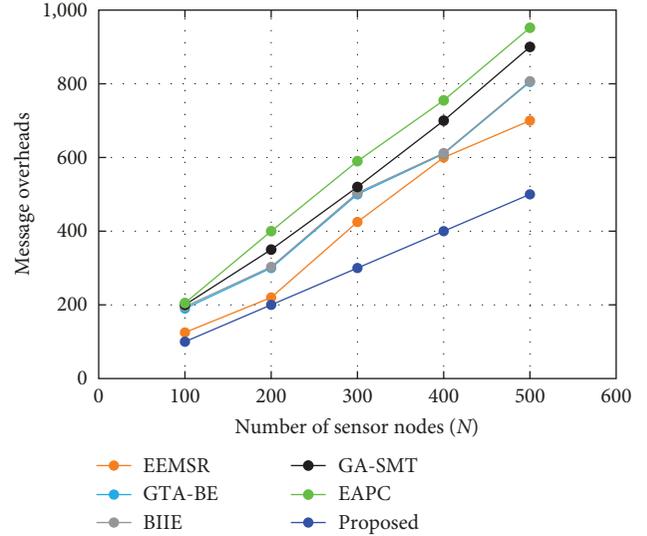


FIGURE 10: Overhead communications versus the number of deployed SNs.

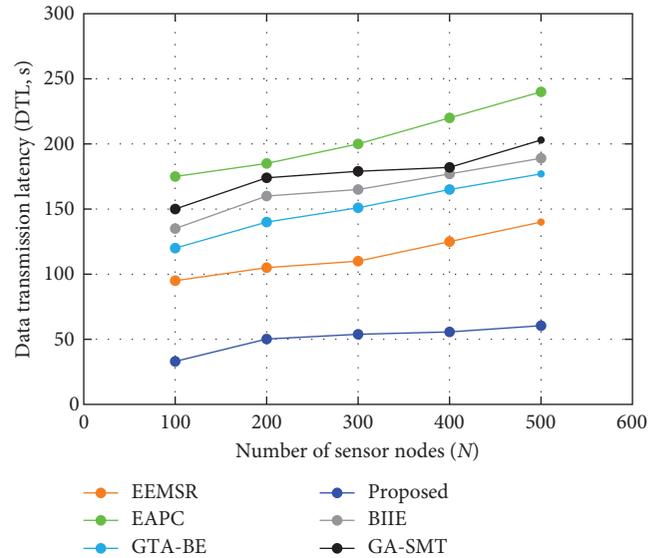


FIGURE 11: Data transmission latency in seconds versus the number of deployed SNs, N .

trajectory time $(p_1 + p_2 + \dots + p_K) \times v$, where $p_j \in \mathbf{P}$ and $j = 1, 2, 3, \dots, K$ and K is the number of clusters generated by the K-mean algorithm. (2) The amount of time due to the MS spent hovering over each RVP location $r_i \in \mathcal{R}$. Consequently, the DTL can be given as follows:

$$\text{DTL} = \left(\sum_{i=1}^K p_i \right) \times v + \sum_{i=1}^{|\mathcal{R}|} T_{H_i}. \quad (26)$$

Figure 11 illustrates the vast noticeable reduction of the DTL among the proposed scheme and the remaining algorithms. The amount of DTL increases as the number of sensors increases. When 100 sensors are used, the proposed scheme

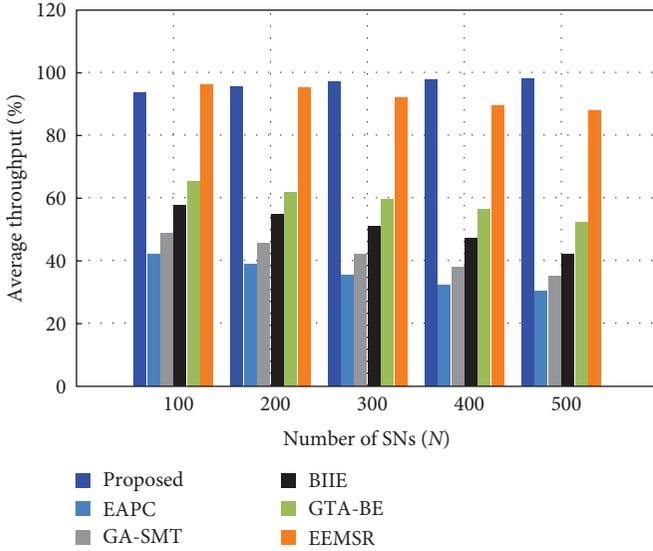


FIGURE 12: Average throughput of the network for 500 rounds.

is reduced up to 15.3% as compared EEMSR, up to 72.5% compared to GTA-BE, up to 75.5% compared to BIIE, up to 87.9% compared to GA-SMT, up to 81.5% compared to EAPC. This reduction is achieved due to two main reasons. First, selecting the optimal number of RVPs. Second, the positions of RVPs are designed to allow MS to serve four CHs at each move. In other algorithms, each RVP is selected to serve one CH. So, the proposed scheme overcame the compared algorithms and its MS become able to accomplish the data gathering process in a more convenient time.

According to the aforementioned network model, the intended geographic area is divided into equal-sized clusters based on the range of these similar sensors. In addition, each cluster's members perform their task independently to other cluster's members. Thus, the scalable of WSN is simply possible by employing additional number of clusters to cover the extended area. Each of additional clusters will supply by an excess number of required SNs. This scalability may cause an increasing in path trajectory of the MS when move to gather the aggregated data from the CHs of different clusters. The lengthen path may cause bad effect on the delay bound of the collected data. However, this problem can be avoided by assigning more than MS with different path. This mean the whole area is portioned into more than section. Each section has its own MS and each MS has its own path. So, the architecture of the proposed scheme can be designed to be flexible and scaled to large numbers of nodes. This allows an easy adaptation to different environments and tolerating an extensive data collection and monitoring in a wide range of applications.

5.6. Throughput Analysis. The network throughput is characterized as the proportion of the aggregate packets received by the BS to the total number of packets aggregated at the CHs and transmitted to the MS at RVPs [23]. Figure 12 illustrates the mean throughput of the proposed scheme in comparison to other state-of-the-art algorithms. The X-axis

represents the total SNs while the Y-axis signifies the overall percentage of the average throughput over 5,000 rounds. From the figure, we note that the proposed scheme exhibits an enhanced average throughput, surpassing EAPC, GA-SMT, BIIE, GTA-BE, and EEMSR by 60.27%, 54.1%, 45.45%, 36.9%, and 4.9%, respectively. The proposed scheme consistently demonstrates commendable average throughput throughout the majority of rounds. This is deduced due to the equilibrium in energy consumption across each SN over the network's lifespan that ensures each SN preserves its residual energy for an extended duration.

5.7. Statistical Analysis. The one-way analysis of variance (ANOVA) is a statistical analysis widely employed across diverse fields such as psychology, biology, sociology, and business [41, 42]. It is utilized to ascertain whether there are any statistically significant differences between the means of one or more independent (unrelated) groups [43, 44]. In this paper, we conduct a one-way ANOVA test on the sample data generated by both the proposed algorithm and state-of-the-art algorithms, including EEMSR, GTA-BE, BIIE, GA-SMT, and EAPC. To conduct the test, we have taken into account performance metrics such as stability period, network lifetime, average energy consumption, communication MO, DTL, and throughput. Additionally, we utilize the same parameters as depicted in Table 3.

The ANOVA test determines whether the null hypothesis H_0 (which suggests that the means of the given algorithms are the same) can be rejected, thereby accepting the alternative hypothesis H_1 (indicating that the means of the algorithms are significantly different), or whether H_0 is accepted and H_1 is rejected [44]. Let the null hypothesis be as follows:

$$H_0: \mu_{\text{proposed}} = \mu_{\text{EEMSR}} = \mu_{\text{GTA-BE}} = \mu_{\text{BIIE}} = \mu_{\text{GA-SMT}} = \mu_{\text{EAPC}}, \quad (27)$$

$$H_1: \mu_{\text{proposed}} \neq \mu_{\text{EEMSR}} \neq \mu_{\text{GTA-BE}} \neq \mu_{\text{BIIE}} \neq \mu_{\text{GA-SMT}} \neq \mu_{\text{EAPC}}, \quad (28)$$

where μ_x represents the mean of algorithm x . The ANOVA table provides a comprehensive summary of various statistical measures including sources of variation, sum of squares SS, degrees of freedom df, mean squares MS, F -statistical value (F_s), p -value (p_v) and F -critical value (F_c). It delineates three primary sources of variation: between-group variation, which accounts for differences between group means; within-group variation, reflecting differences within each group; and total variation, representing the overall variability in the dataset. The value of significance level is considered as, $\alpha = 0.05$.

Typically, if the p_v is less than the chosen significance level α and the F_s exceeds the F_c , then the null hypothesis H_0 in Equation (27) is rejected, and the alternative hypothesis H_1 in Equation (28) is accepted. Otherwise, H_0 is accepted, and H_1 is rejected. Therefore, we encounter the following two conditions:

TABLE 4: Results of ANOVA test.

(a) Stability period						
Source of variation	SS	df	MS	F_s	p_v	F_c
Between groups	4.08E + 07	5	8.16E + 06	106.47	1.42E – 15	2.621
Within groups	1.84E + 06	24	7.66E + 04	—	—	—
Total	4.26E + 07	29	—	—	—	—
(b) Network lifetime						
Source of variation	SS	df	MS	F_s	p_v	F_c
Between groups	3.56E + 08	5	7.13E + 07	57.01	1.56E – 12	2.621
Within groups	13.00E + 07	24	1.25E + 06	—	—	—
Total	3.86E + 08	29	—	—	—	—
(c) Average energy consumption						
Source of variation	SS	df	MS	F_s	p_v	F_c
Between groups	7.41E + 04	5	1.48E + 04	21.18	3.89E – 15	2.621
Within groups	8.40E + 04	120	7.00E + 02	—	—	—
Total	1.58E + 05	125	—	—	—	—
(d) Communication message overhead						
Source of variation	SS	df	MS	F_s	p_v	F_c
Between groups	2.43E + 05	5	4.85E + 04	0.80	0.56	2.621
Within groups	1.46E + 06	24	6.10E + 04	—	—	—
Total	1.71E + 06	29	—	—	—	—
(e) Data transmission latency						
Source of variation	SS	df	MS	F_s	p_v	F_c
Between groups	4.73E + 04	5	9.45E + 03	23.773	1.42264E – 08	2.621
Within groups	9.54E + 03	24	3.98E + 02	—	—	—
Total	5.68E + 04	29	—	—	—	—
(f) Throughput						
Source of variation	SS	df	MS	F_s	p_v	F_c
Between groups	1.66E + 04	5	3.32E + 03	148.590	3.06017E – 17	2.621
Within groups	536.255	24	22.344	—	—	—
Total	1.71E + 04	29	—	—	—	—

TABLE 5: Tukey's HSD test results: stability period.

Treatments pair	Stability period			
	Tukey HSD Q statistic	Tukey HSD p_v	Tukey HSD inference	Null hypothesis
Proposed vs. EAPC	26.731	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GA-SMT	24.065	0.0010053	$p_v < 0.05$	Reject
Proposed vs. BII E	22.4493	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GTA-BE	19.8641	0.0010053	$p_v < 0.05$	Reject
Proposed vs. EEMSR	9.2001	0.0010053	$p_v < 0.05$	Reject

- (1) Condition 1: Reject H_0 and accept H_1 , if $p_v < \alpha$ and $F_s > F_c$.
- (2) Condition 2: Accept H_0 and reject H_1 , if condition 1 is not satisfied.

The ANOVA results, displayed in Table 4, outline the performance of six different algorithms across multiple metrics: stability period, network lifetime, average energy consumption, communication MO, DTL, and throughput. Notably, five of these metrics meet Condition 1, leading to the rejection of the

null hypothesis. However, for communication MO, the null hypothesis is accepted. This suggests that there is a significant difference between at least one of the means of the algorithms. However, without further information, it remains unclear which specific algorithm or algorithms contribute to this distinction, highlighting the necessity for Tukey's honest significant difference (HSD) test [41]. Tables 5–9 present the results obtained from conducting the Tukey's HSD test. Upon thorough examination of these tables, it becomes apparent that the "reject" value in the last column for each scenario signifies the rejection of the

TABLE 6: Tukey's HSD test results: network lifetime.

Network lifetime				
Treatments pair	Tukey HSD Q statistic	Tukey HSD p_v	Tukey HSD inference	Null hypothesis
Proposed vs. EAPC	19.6253	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GA-SMT	18.2253	0.0010053	$p_v < 0.05$	Reject
Proposed vs. BII E	16.2654	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GTA-BE	15.1654	0.0010053	$p_v < 0.05$	Reject
Proposed vs. EEMSR	7.5455	0.0010053	$p_v < 0.05$	Reject

TABLE 7: Tukey's HSD test results: average energy consumption.

Average energy consumption				
Treatments pair	Tukey HSD Q statistic	Tukey HSD p_v	Tukey HSD inference	Null hypothesis
Proposed vs. EAPC	11.7877	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GA-SMT	10.8945	0.0010053	$p_v < 0.05$	Reject
Proposed vs. BII E	10.3586	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GTA-BE	9.4817	0.0010053	$p_v < 0.05$	Reject
Proposed vs. EEMSR	5.4461	0.0025754	$p_v < 0.05$	Reject

TABLE 8: Tukey's HSD test results: data transmission latency.

Data transmission latency				
Treatments pair	Tukey HSD Q statistic	Tukey HSD p_v	Tukey HSD inference	Null hypothesis
Proposed vs. EAPC	13.4264	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GA-SMT	10.4662	0.0010053	$p_v < 0.05$	Reject
Proposed vs. BII E	9.0757	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GTA-BE	7.43867	0.00300533	$p_v < 0.05$	Reject
Proposed vs. EEMSR	3.4467	0.183274	Insignificant	Accept

TABLE 9: Tukey's HSD test results: throughput.

Throughput				
Treatments pair	Tukey HSD Q statistic	Tukey HSD p_v	Tukey HSD inference	Null hypothesis
Proposed vs. EAPC	28.6979	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GA-SMT	25.7781	0.0010053	$p_v < 0.05$	Reject
Proposed vs. BII E	21.6904	0.0010053	$p_v < 0.05$	Reject
Proposed vs. GTA-BE	17.6557	0.00300533	$p_v < 0.05$	Reject
Proposed vs. EEMSR	2.0364	0.6809685	Insignificant	Accept

null hypothesis. This leads us to confidently affirm that our proposed algorithm exhibits statistical significance, showing notable differences from the other algorithms. However, it is paramount to underscore that while the proposed algorithm yields enhancements in throughput and latency by 4.9% and 15.3%, respectively, in comparison with the EEMSR algorithm as delineated in Sections 5.5 and 5.6, this advancement is deemed statistically insignificant, as evidenced by Tukey's honestly significant difference (HSD) test, as elucidated in Tables 8 and 9.

6. Conclusion

This paper introduces an intelligent energy-efficient data routing scheme for WSNs utilizing MS. The operations of

the proposed scheme are performed in two successive modes: configure and operational modes. The configure mode includes cluster construction, TDMA scheduling, and CHs selection. The self-encoding module in each SN is used to pick the CH in operational mode without any communication between the CMs within each cluster. For the MS-based data routing, the optimal number of RVPs is selected, from which the MS's best path is determined using K-mean clustering and GAs. Simulations and analysis have been implemented to confirm the effectiveness of the proposed scheme. The simulation results guarantee that the offered scheme is more efficiently than the current state-of-the-algorithms such as GTA-BE, BII E, GA-SMT, EAPC, and EEMSR in terms of the stability period, network lifetime, average energy consumption, DTL, MOs, and the

average of throughput. The findings are further supported by statistical validation through hypothesis testing and Tukey's honest significant difference analysis.

The implementation of the proposed scheme may depend on the environment of the intended scenario. For instance, scenarios of monitoring border's activities or traffic monitoring may need to specified knowledge and proficiency to determine the appropriate locations for the WSN. One of the implications affects these scenarios is interference with signals of other wireless sensor devices. This causes a bad influence on network performance and reliability. In future work, we aim to enhance the proposed scheme to handle scalability and obstacles issues. In addition, we investigate the deployment of multiple MSs simultaneously in large scale to cover different areas.

Data Availability

No underlying data were collected or produced in this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] N. R. Patel, S. Kumar, and S. K. Singh, "Energy and collision aware WSN routing protocol for sustainable and intelligent IoT applications," *IEEE Sensors Journal*, vol. 21, no. 22, pp. 25282–25292, 2021.
- [2] J. Amutha, S. Sharma, and J. Nagar, "WSN strategies based on sensors, deployment, sensing models, coverage and energy efficiency: review, approaches and open issues," *Wireless Personal Communications*, vol. 111, no. 2, pp. 1089–1115, 2020.
- [3] B. Bhushan and G. Sahoo, "Requirements, protocols, and security challenges in wireless sensor networks: an industrial perspective," in *Handbook of Computer Networks and Cyber Security*, B. B. Gupta, G. M. Pérez, D. P. Agrawal, and D. Gupta, Eds., pp. 683–713, Springer, Cham, 2020.
- [4] M. G. Ball, B. Qela, and S. Wesolkowski, "A review of the use of computational intelligence in the design of military surveillance networks," in *Recent Advances in Computational Intelligence in Defense and Security*, vol. 621 of *Studies in Computational Intelligence*, pp. 663–693, Springer, Cham, 12 2016.
- [5] R. Chaudhry, S. Tapaswi, and N. Kumar, "A green multicast routing algorithm for smart sensor networks in disaster management," *IEEE Transactions on Green Communications and Networking*, vol. 3, no. 1, pp. 215–226, 2019.
- [6] A. Jindal, V. Agarwal, and P. Chanak, "Emergency evacuation system for clogging-free and shortest-safe path navigation with IoT-enabled WSNs," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 10424–10433, 2022.
- [7] S. Jain, R. K. Verma, K. K. Pattanaik, and A. Shukla, "A survey on event-driven and query-driven hierarchical routing protocols for mobile sink-based wireless sensor networks," *The Journal of Supercomputing*, vol. 78, no. 9, pp. 11492–11538, 2022.
- [8] A. A. Okine, N. Adam, F. Naeem, and G. Kaddoum, "Multi-agent deep reinforcement learning for packet routing in tactical mobile sensor networks," *IEEE Transactions on Network and Service Management*, 2024.
- [9] T. A. Alghamdi, "Energy efficient protocol in wireless sensor network: optimized cluster head selection model," *Telecommunication Systems*, vol. 74, no. 3, pp. 331–345, 2020.
- [10] Z. Liu, Y. Liu, and X. Wang, "Intelligent routing algorithm for wireless sensor networks dynamically guided by distributed neural networks," *Computer Communications*, vol. 207, pp. 100–112, 2023.
- [11] J. Wang, Y. Gao, X. Yin, F. Li, and H.-J. Kim, "An enhanced PEGASIS algorithm with mobile sink support for wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 9472075, 9 pages, 2018.
- [12] I. Ha, M. Djuraev, and B. Ahn, "An optimal data gathering method for mobile sinks in WSNs," *Wireless Personal Communications*, vol. 97, no. 1, pp. 1401–1417, 2017.
- [13] M. K. Roberts and J. Thangavel, "An optimized ticket manager based energy-aware multipath routing protocol design for IoT based wireless sensor networks," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 28, Article ID 10, 2022.
- [14] M. Rathee, S. Kumar, A. H. Gandomi, K. Dilip, B. Balusamy, and R. Patan, "Ant colony optimization based quality of service aware energy balancing secure routing algorithm for wireless sensor networks," *IEEE Transactions on Engineering Management*, vol. 68, no. 1, pp. 170–182, 2021.
- [15] M. Elshrkawey, S. M. Elsharif, and M. E. Wahed, "An enhancement approach for reducing the energy consumption in wireless sensor networks," *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 2, pp. 259–267, 2018.
- [16] L. Xu, R. Collier, and G. M. P. O'Hare, "A survey of clustering techniques in WSNs and consideration of the challenges of applying such to 5G IoT scenarios," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1229–1249, 2017.
- [17] J.-C. Cuevas-Martinez, A.-J. Yuste-Delgado, A.-J. Leon-Sanchez, A.-J. Saez-Castillo, and A. Triviño-Cabrera, "A new centralized clustering algorithm for wireless sensor networks," *Sensors*, vol. 19, no. 20, Article ID 4391, 2019.
- [18] K. Guleria and A. K. Verma, "Comprehensive review for energy efficient hierarchical routing protocols on wireless sensor networks," *Wireless Networks*, vol. 25, no. 3, pp. 1159–1183, 2019.
- [19] V. Agarwal, S. Tapaswi, and P. Chanak, "A survey on path planning techniques for mobile sink in IoT-enabled wireless sensor networks," *Wireless Personal Communications*, vol. 119, pp. 211–238, 2021.
- [20] B. Gutam, P. K. Donta, C. Sekhar, and Y.-C. Hu, "Optimal rendezvous points selection and mobile sink trajectory construction for data collection in WSNS," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, pp. 7147–7158, 2023.
- [21] H. Al-Mahdi, M. sharkawy, S. Saad, and S. Abdul Aziz, "An energy-efficient mobile sink-based intelligent data routing scheme for wireless sensor networks," Available at SSRN, vol. 111, [Online], 2023.
- [22] M. K. Roberts and P. Ramasamy, "An improved high performance clustering based routing protocol for wireless sensor networks in IoT," *Telecommunication Systems*, vol. 82, pp. 45–59, 2023.
- [23] H. Huang and A. V. Savkin, "An energy efficient approach for data collection in wireless sensor networks using public transportation vehicles," *AEU-International Journal of Electronics and Communications*, vol. 75, pp. 108–118, 2017.

- [24] A. Mehto, R. K. Verma, and S. Jain, "Efficient trajectory planning and route adjustment strategy for mobile sink in WSN-assisted IoT," in *2022 IEEE Region 10 Symposium (TENSYMP)*, pp. 1–6, IEEE, Mumbai, India, 2023.
- [25] W. Wen, S. Zhao, C. Shang, and C.-Y. Chang, "EAPC: energy-aware path construction for data collection using mobile sink in wireless sensor networks," *IEEE Sensors Journal*, vol. 18, no. 2, pp. 890–901, 2018.
- [26] Y.-C. Wang and K.-C. Chen, "Efficient path planning for a mobile sink to reliably gather data from sensors with diverse sensing rates and limited buffers," *IEEE Transactions on Mobile Computing*, vol. 18, no. 7, pp. 1527–1540, 2019.
- [27] X. Fu and X. He, "Energy-balanced data collection with path-constrained mobile sink in wireless sensor networks," *AEU-International Journal of Electronics and Communications*, vol. 127, Article ID 153504, 2020.
- [28] A. Mehto, S. Tapaswi, and K. Pattanaik, "Optimal rendezvous points selection to reliably acquire data from wireless sensor networks using mobile sink," *Computing*, vol. 103, pp. 707–733, 2021.
- [29] V. Agarwal, S. Tapaswi, and P. Chanak, "Energy-efficient mobile sink-based intelligent data routing scheme for wireless sensor networks," *IEEE Sensors Journal*, vol. 22, no. 10, pp. 9881–9891, 2022.
- [30] M. K. Singh, S. I. Amin, and A. Choudhary, "Genetic algorithm based sink mobility for energy efficient data routing in wireless sensor networks," *AEU-International Journal of Electronics and Communications*, vol. 131, Article ID 153605, 2021.
- [31] B. M. Sahoo, H. M. Pandey, and T. Amgoth, "GAPSO-H: a hybrid approach towards optimizing the cluster based routing in wireless sensor network," *Swarm and Evolutionary Computation*, vol. 60, Article ID 100772, 2021.
- [32] J. Wang, Y. Gao, C. Zhou, R. S. Sherratt, and L. Wang, "Optimal coverage multi-path scheduling scheme with multiple mobile sinks for WSNs," *Computers, Materials & Continua*, vol. 62, no. 2, pp. 695–711, 2020.
- [33] C. S. Gowda and P. V. Y. Jayasree, "Rendezvous points based energy-aware routing using hybrid neural network for mobile sink in wireless sensor networks," *Wireless Networks*, vol. 27, pp. 2961–2976, 2021.
- [34] D. P. Kumar, A. Tarachand, and A. C. S. Rao, "ACO-based mobile sink path determination for wireless sensor networks under non-uniform data constraints," *Applied Soft Computing*, vol. 69, pp. 528–540, 2018.
- [35] P. K. Donta, T. Amgoth, and C. S. R. Annavarapu, "An extended ACO-based mobile sink path determination in wireless sensor networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 8991–9006, 2021.
- [36] G. Gupta and B. Saha, "Load balanced clustering scheme using hybrid metaheuristic technique for mobile sink based wireless sensor networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, pp. 5283–5294, 2020.
- [37] P. V. P. Raj, Z. Al Aghbari, and A. M. Khedr, "Data gathering via mobile sink in WSNs using game theory and enhanced ant colony optimization," *Wireless Networks*, vol. 26, pp. 2983–2998, 2020.
- [38] J. Amutha, S. Sharma, and S. K. Sharma, "An energy efficient cluster based hybrid optimization algorithm with static sink and mobile sink node for wireless sensor networks," *Expert Systems with Applications*, vol. 203, Article ID 117334, 2022.
- [39] S. Chatterjee, C. Carrera, and L. A. Lynch, "Genetic algorithms and traveling salesman problems," *European Journal of Operational Research*, vol. 93, no. 3, pp. 490–510, 1996.
- [40] M. Srinivas and L. M. Patnaik, "Genetic algorithms: a survey," *Computer*, vol. 27, no. 6, pp. 17–26, 1994.
- [41] A. Kaswan, V. Singh, and P. K. Jana, "A multi-objective and PSO based energy efficient path design for mobile sink in wireless sensor networks," *Pervasive and Mobile Computing*, vol. 46, pp. 122–136, 2018.
- [42] D. K. Sah, K. Cengiz, P. K. Donta, V. N. Inukollu, and T. Amgoth, "EDGF: empirical dataset generation framework for wireless sensor networks," *Computer Communications*, vol. 180, pp. 48–56, 2021.
- [43] C. B. K. Yadav and D. Dash, "An energy efficient periodic data gathering and charging schedule using MVs in wireless rechargeable sensor networks," *Computing*, vol. 105, no. 11, pp. 2563–2593, 2023.
- [44] R. Anwit, A. Tomar, and P. K. Jana, "Tour planning for multiple mobile sinks in wireless sensor networks: a shark smell optimization approach," *Applied Soft Computing*, vol. 97, Part A, Article ID 106802, 2020.