SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V13I3P110

Volume 13 Issue 3, 125-139, March 2026

Original Article

A Novel SCH-VSCH Selection-Enabled Energy Efficient Optimal Path Selection in WSN using LA-FLS and BM-SCSO

Ab Wahid Bhat[1*], Abhiruchi Passi[1]

[1]*Department of ECE, FET MRIIRS, Faridabad, Haryana, India.*

[*1]*Corresponding Author : abwahid1994@gmail.com*

Received: 09 December 2025 Revised: 10 January 2026 Accepted: 15 February 2026 Published: 31 March 2026s

Abstract *- In Wireless Sensor Networks (WSNs), Energy optimization focuses on lessening energy consumption to lengthen the network's lifetime. Nevertheless, the existing studies did not perform a Vice Super Cluster Head (VSCH) with secure handover when the Super Cluster Head (SCH) energy is dropped. Thus, this paper proposes Super Cluster Head Vice Super Cluster Head (SCH-VSCH) selection-enabled Energy-Efficient (EE) optimal path selection in WSN using Log Adjustable Fuzzy Logic System (LA-FLS) and Baker Map Sand Cat Swarm Optimization (BM-SCSO). Initially, the WSN nodes are initialized randomly. Then, by using Clipped Voronoi Diagram-Sinusoidal Sigma Representation Yule's K-Means clustering (CVD-2SRY-KMeans), the redundant Sensor Nodes (SNs) are reduced. Next, based on LS-FLS, the SCH-VSCH is selected. Afterward, between the source and destination, the hop count is estimated. Then, by using the Log Hop First Scheduling Algorithm (LHFSA), the nodes are scheduled according to the hop count. Later, by using the Ad hoc On-demand Distance Vector (AODV) protocol, the possible routes for the packet transmission are generated. Currently, from the SNs, the data is sensed. The congestion of the node is estimated by using LA-FLS according to the packet arrival rate. If congestion is present, then it is minimized by the backpressure method. Lastly, by employing BM-SCSO, the optimal path is selected. SCH's position is securely handed over to VSCH by the Doubling-Doche-Digital-Oriented-Icart-Kohel Signature Algorithm (D3-OIKSA) if the SCH energy drops. Next, for reliable packet transmission, the remaining steps are processed. As per the results, the proposed framework achieved a high throughput of 56825.74916 bytes, thus outperforming prevailing techniques.*

Keywords *- Wireless Sensor Networks (WSN), Super Cluster Head (SCH), Vice SCH (VSCH), Optimal path selection, Secure handover, Congestion reduction, Energy optimization, and Ad hoc On-demand Distance Vector (AODV).*

1. Introduction

WSNs are wielded in several applications, including disaster management, drone applications, and medical applications, owing to their rapid growth [1]. Usually, WSNs are self-organizing networks that consist of a number of nodes with reduced cost. WSN has the ability to capture diverse types of environmental and physical conditions. However, it has a lower battery, lower data processing capacity, and a small size [2]. Nevertheless, these limited characteristics highlight the need for efficient management of routing tasks to improve the lifespan of WSNs [3]. In WSN, when energy usage is employed for data sensing, data collection, and data transmission, it is continuous. The packet payload size is increased owing to the mismanagement of WSNs, thus resulting in data packets dropping [4]. Thus, efficient data transfer is affected, and the retransmission of data packets consumes more energy, thus impacting the energy efficiency of WSN. Lately, to augment the energy efficiency in WSNs, many approaches have been developed, thus improving the network's lifespan and increasing overall performance [5]. For data transmission, enhancing the energy efficiency in WSN, optimal Cluster Head (CH) selection, along with efficient path selection, is done. The network is split into clusters by cluster-centric protocols; here, each cluster consists of CHs [6]. For balancing the energy load among nodes, optimal CH selection considers factors like distance to neighboring nodes, Residual Energy (RE), and network density. Similarly, by minimizing the transmission distance and eliminating energy-depleted nodes, optimal path selection for data transmission reduces energy consumption [7]. For EE routing in WSN, the Exponentially-Ant Lion Whale Optimization (E-ALWO) was introduced in existing studies [8]. Similarly, for performing EE cluster-based routing in WSN, some prevailing works employed Orphan-Low Energy Adaptive Clustering Hierarchy (O-LEACH), Simulated Annealing with Lightning Search Algorithm (SA-LSA), along with Particle Swarm Optimization with LSA (PSO-LSA) [9]. Likewise, for optimal data transmission and pathfinding in

WSN, Incremental Grey Wolf Optimization (I-GWO), along with Expanded GWO (Ex-GWO), was utilized by conventional studies [10].

Likewise, for CH selection in WSN, the Improved GWO and PSO were used by certain prevailing studies [11, 12]. Similarly, for lifetime along with energy optimization in WSN, the Genetic Algorithm (GA) and Least-Squares Policy Iteration (LSPI) were established [13]. Also, by using a fuzzy-centric sleep scheduling algorithm, the energy efficiency of WSNs is enhanced [14].

Similarly, for energy-efficient optimization in WSN, the Gaussian elimination merged with Distributed Energy Efficient Clustering (DEEC-Gauss) was utilized [15]. Nevertheless, the prevailing studies did not perform VSCH with secure handover if the SCH energy is dropped, thus affecting the functionality and efficiency of WSN. This article presents SCH-VSCH selection-based EE optimal path selection in WSN using LA-FLS and BM-SCSO. This remaining part is arranged as follows: the existing research works are conveyed in Section 2, the proposed methodology is illustrated in Section 3, the results and discussion are presented in Section 4, and finally, Section 5 concludes the proposed method with future scope.

2. Literature Survey

Huang et al. [16] offered a framework named a hybrid Intrusion Detection System (IDS) deployment for multi-hop clustered WSNs. Here, by using the IDS architecture, the security of multi-hop clustered WSNs was enhanced. Likewise, for allocating the resources optimally among IDS agents, the Optimal Resource Allocation (ORA) algorithm was introduced, thus maximizing the network's security metrics. The model improved the computational efficiency. However, the packets originating from nodes far from the CH required huge network resources to reach their destination, thereby consuming more energy and bandwidth. Umbreen et al. [17] recommended EE mobility-centric CH selection in WSN for lifetime enhancement. Here, by using an EE Mobility-centric Clustering Strategy (EEMCS), the network lifetime was extended. Similarly, centered on the mobility level of the node, RE, density of neighbors, and distance to sink, the weightage of each node was computed. Consequently, the model provided promising outcomes regarding throughput, energy depletion, network stability, and load balancing. Nevertheless, the research deployed the WSN nodes randomly, thus increasing the redundant data in the network and leading to energy wastage.

Xu et al. [18] offered an EE connected-coverage scheme in WSNs. Primarily, to discover the minimal number of SNs, the Genetic Algorithm (GA)-based node scheduling scheme was employed for checking all target points. Then, the dead nodes were replaced by choosing redundant sleep nodes to prolong the Network Lifetime (NL). Next, by utilizing Improved-Distributed EE Clustering (I-DEEC), routing was performed. Therefore, the model improved data transmission's energy efficiency and extended the NL. Nonetheless, the SCH was chosen based on the RE only, which diminished the security level.

Dattatraya & Rao [19] discovered a model for enhancing the NL along with energy efficiency in WSN centered on CH selection. Here, a hybrid glowworm swarm optimization algorithm with a fruitfly optimization was employed for selecting the best CH in WSN. As per the outcome, regarding alive node analysis, energy analysis, and the cost function, the explored model provided better outcomes. However, owing to its reliance on centralized or semi-centralized processing, the model struggled to handle very large-scale networks.

Kathiroli & Selvadurai [20] offered EE CH selection in WSN. Here, to select the best possible CHs in WSN, the sparrow search algorithm's higher-level search effectiveness, along with the differential evolution algorithm's lively potential, was used. The model efficiently solved the energy efficiency issue and enhanced the lifetime of nodes and the number of alive nodes. The performance of the model could degrade as the size of the network increases in dynamic environments with frequent topology changes.

Divya & Sudhakar [21] advanced a model for route optimization along with CH selection in WSN. Here, for choosing non-CHs and CHs, the adaptive multi-path routing protocol, which was selected employing the Circle-Inspired Optimization Algorithm (CIOA), was employed. Likewise, by using the CIOA, the optimal shortest path was identified. When analogized to the prevailing techniques, this research consumed less energy. Nevertheless, slower convergence and sub-optimal outcomes were caused by poor initial settings of CIOA.

Nagaraju et al. [22] applied secure routing-enabled energy optimization for the Internet of Things (IoT) with heterogeneous WSNs. Here, for secure routing in WSN, the multi-path link routing protocol was utilized. Then, by using the Hierarchical Threshold sensitive Energy Efficient Network (H-TEEN) protocol, the energy and NL were enhanced. Similarly, to improve data storage capacity, a ubiquitous data storage protocol was established. Therefore, the model improved the throughput and energy efficiency while reducing the end-to-end delay. Nevertheless, the model quickly depleted the low-energy nodes, thus degrading the efficiency.

Jagan & Jesu Jayarin [23] presented a model for CH selection and short routing in WSN. Here, for shortest path routing from SNs and CH selection, the Electro-Static Discharge Algorithm (ESDA) was employed. The NL was improved by obtaining EE full connectivity between SNs with the help of ESDA. Likewise, the dead node count was

minimized, thus increasing network longevity. However, frequent re-election of CHs introduced extra energy consumption, thereby reducing network efficiency. Singh et al. [24] deployed EE Multi-path Routing (EEMR) for WSNs. Here, centered on multi-objective lion optimization, the optimal path was computed by EEMR with reduced energy consumption. Here, the EEMR proficiently selected the efficient path amongst multiple paths and satisfied the quality requirements, including throughput, delivery ratio, loss ratio, delay, and latency. Nevertheless, the model did not scale well in networks with a very high density of nodes, leading to higher convergence times.

Hu et al. [25] presented trust-enabled secure and EE routing in WSNs. Here, by calculating the comprehensive trust value based on adaptive direct, indirect, and energy trust values, the Trust-Based Secure and Energy Efficient Routing protocol (TBSEER) resisted the black hole, sinkhole, and hello flood attacks. Therefore, the research minimized the network energy consumption while quickly identifying the malicious nodes. Nevertheless, biased routing decisions were caused by misconfigured initial trust values. The summary of a few prominent algorithms is mentioned in Table 1.

2.1. Problem Statement

The conventional EE optimal path selection in WSN frameworks has some limitations as follows:

- The nodes were deployed randomly in a conventional manner. This increases the redundant data in the network, thus leading to energy wastage.
- In most of the existing works, the network management was less efficient owing to the variations in network topology or changes in node connectivity that necessitate frequent re-clustering, thus consuming more energy.
- The prevailing algorithms choose the SCH regarding the RE only. Therefore, it could rigorously affect the functionality and efficiency of WSN.
- The packets originating from nodes far from the CH needed more network resources to reach their destination, consuming more energy and bandwidth.
- None of the existing works concentrated on the vice or secondary super CH with secure handover when the SCH energy is dropped, thus diminishing the security level. The attackers might compromise them to obtain control over significant parts of the network.

2.2. Objective

The proposed model's key objectives are listed below:

- CVD-2SRY-KMeans is employed to minimize the amount of redundant data in the network to avoid energy wastage.
- 2SRY-K-Means is used for frequent re-clustering, thus making the network management efficient and reducing energy consumption.
- LA-FLS is utilized to select the SCH and VSCH regarding RE, degree of centrality, degree of node trust, and minimum distance from the gateway for improving the security level.
- LHFSA is implemented to minimize energy consumption and bandwidth.
- D3-OIKSA is introduced for performing secure handover to VSCH when the SCH energy is dropped, thus improving the functionality and efficiency of WSN.

3. Proposed SCH-VSCH Selection-Enabled Energy-Efficient Optimal Path Selection In The WSN Framework

Here, to select the SCH-VSCH, the proposed LA-FLS is introduced. Likewise, by employing the proposed CVD-2SRY-KMeans, the redundant SNs are reduced. Similarly, the proposed D3-OIKSA is established for secure handover to VSCH selection. Likewise, by utilizing the proposed LHFSA, the nodes are scheduled. Likewise, to select the optimal paths, the BM-SCSO is introduced. In Figure 1, the proposed model's structural layout is displayed.

3.1. Node Initialization

Initially, the WSN nodes are randomly distributed over the network for data transmission. Hence, the randomly deployed WSN nodes (χ_a^{WSN}) are specified as,

$$\chi_a^{WSN} \rightarrow \left(\chi_1^{WSN} + \chi_2^{WSN} + \chi_3^{WSN} + \cdots \ldots + \chi_\varsigma^{WSN}\right) \quad (1)$$

Here, $a = (1\ to\ \varsigma)$ specifies the number of randomly deployed WSN nodes.

Table 1. Summary of various prominent algorithms

Author's name	Objective	Techniques used	Findings	Advantages	Limitations
(Huang et al., 2022)	Efficient hybrid Intrusion Detection System (IDS) deployment for multi-hop clustered WSN.	Optimal Resource Allocation (ORA) approach	Time cost: 41.26	The model effectively maximized the network security and improved the network performance.	Yet, the packets that were far from the CH needed more network resources to reach their destination. Also, they may pass through multiple intermediate nodes, which consume more energy and bandwidth.

(Xu et al., 2020)	Energy-efficient connected-coverage scheme in WSN	Minimum Coverage Control Algorithm (MCCA) and Improved-Distributed Energy Efficient Clustering (I-DEEC) protocol	Network energy consumption: 75.54	The research successfully optimized the energy efficiency of data transmission and improved the lifetime of the network.	Moreover, in this work, the chosen supercluster head considered only the residual energy, which mainly degraded the security level.
(Umbreen et al., 2020)	Energy-efficient mobility-based cluster head selection for lifetime enhancement in WSN	Energy-efficient hierarchical clustering mechanism	Energy consumption: 0.1 Joules for 50 nodes	The presented research proficiently improved the network stability, throughput, and load balancing.	Furthermore, in this research, the nodes were deployed randomly, which increased the redundant data in the network. Therefore, energy was wasted aggregating, processing, and transmitting redundant data.
(Rami Reddy et al., 2023)	Energy-efficient cluster head selection in WSN	Improved Grey Wolf Optimization Algorithm (IGWOA)	Network stability: 333.51%	The research effectively solved the premature convergence and enhanced the lifetime of the network.	However, the model had scalability issues, and it was harder to implement.
(Rathore et al., 2021)	Energy-optimized CH selection for WSN	Mutually exclusive distributive clustering protocol and Shortest Path Selection for Relay Node (SPSRN) approach	Throughput: 70000	The model effectively reduced energy consumption.	But the model took more time to discover the cluster and cluster head owing to the mutual exclusion principle.

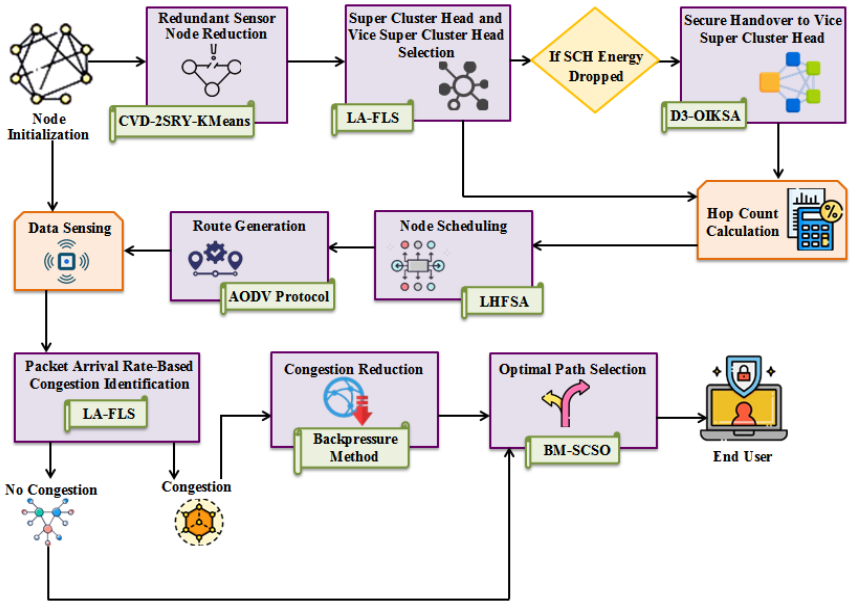


Fig. 1 Structural layout of the proposed model

The proposed framework's step-by-step process is elucidated briefly below.

3.2. Redundant Sensor Node Reduction

Redundant data may be sensed owing to the SN's random deployment because of the inappropriate clusters, thus causing energy wastage. Hence, the redundant SNs are reduced (χ_a^{WSN}) by using CVD-2SRY-KMeans to avoid energy wastage. Usually, K-Means has the ability to handle large datasets with high dimensionality, and it is computationally efficient. K-means can converge to diverse solutions due to the initial placement of centroids. Nevertheless, poor initialization of centroids may lead to suboptimal results. Likewise, the determination of an optimal number of clusters is difficult. To overcome these issues, Sinusoidal Representation initialization is employed instead of random centroid initialization, and Yule's sigma distance is used instead of Euclidean distance.

The steps involved in the CVD-2SRY-KMeans are explained below, Where, by utilizing a clipped Voronoi diagram, they (χ_a^{WSN}) are divided. Usually, a clipped Voronoi diagram is a variation of the standard Voronoi diagram; here, the diagram is clipped to a particular region or domain. A Voronoi diagram divides a space into regions according to the proximity to a provided set of points. Therefore, the division (χ_a^{WSN}) by the clipped Voronoi diagram is indicated as β_h. Next, they β_h are clustered based on the energy as follows,

Step 1: Initially, based on the Sinusoidal Representation initialization($\delta\partial$), the centroids are initialized. It is written as,

$$\delta\partial \sim U\left(-\sqrt{\frac{6}{\beta_h}}, \sqrt{\frac{6}{\beta_h}}\right) \tag{2}$$

Here, U implies the uniform distribution. The initialized centroids (Ω_c) are signified as,

$$\Omega_c \xrightarrow{\delta\partial} (\Omega_1, \Omega_2, \Omega_3, \dots\dots, \Omega_\kappa) \; where \; c = (1,2, \dots, \kappa) \tag{3}$$

Where κ indicates the number of initialized centroids.

Step 2: Then, by employing Yule's sigma distance, the distance between the (Ω_c) and β_h is estimated. It is expressed as,

$$\partial is(\Omega_c, \beta_h) = \frac{\sqrt{\Omega_c} - \sqrt{\beta_h}}{\sqrt{\Omega_c} + \sqrt{\beta_h}} \tag{4}$$

Here, ∂is specifies the Yule's sigma distance.

Step 3: Afterwards, for each data point that belongs to each cluster, the average is discovered. Next, the new centroid is estimated for all clusters as follows,

$$N = \frac{\sum_{c=1,h=1}^{\kappa,n} Avg_{ch}\beta_h}{\sum_{c=1,h=1}^{\kappa,n} \beta_h} \tag{5}$$

Here, N implies the new centroid, n depicts the number of β_h, and Avg_{ch} signifies the average of the data. Finally, all nodes are reassigned to the new closest centroid of each cluster. Until the best clusters are found, the above-described steps are continued. After reducing the redundant SNs, the obtained essential SNs are signified as Ξ_i. The pseudocode for CVD-2SRY-KMeans is depicted below,

Pseudocode for CVD-2SRY-KMeans

Input: Randomly Deployed WSN Nodes (χ_a^{WSN})
Output: Essential Sensor Nodes (Ξ_i)

Begin
  Initialize (χ_a^{WSN})
  For each (χ_a^{WSN})
    Divide (χ_a^{WSN}) using a clipped Voronoi diagram
      β_h
    Cluster β_h based on energy.
    Estimate Sinusoidal Representation initialization.
      $$\delta\partial \sim U\left(-\sqrt{\frac{6}{\beta_h}}, \sqrt{\frac{6}{\beta_h}}\right)$$
    Initialize centroids(Ω_c) according to ($\delta\partial$)
    Calculate Yule's sigma distance.
      $$\partial is(\Omega_c, \beta_h) = \frac{\sqrt{\Omega_c} - \sqrt{\beta_h}}{\sqrt{\Omega_c} + \sqrt{\beta_h}}$$
    Find the average for each data point (Avg_{ch})
    Compute the new centroid
      $$N = \frac{\sum_{c=1,h=1}^{\kappa,n} Avg_{ch}\beta_h}{\sum_{c=1,h=1}^{\kappa,n} \beta_h}$$
    Reassign each node
  End For
  Obtain Essential Sensor Nodes (Ξ_i)
End

Then, the SCH and VSCH are selected and are explained in the upcoming section.

3.3. Super Cluster Head and Vice Super Cluster Head Selection

The SCH and VSCH are selected by using an LA-FLS (Ξ_i) according to the RE, degree of centrality, degree of node trust, and minimum distance from the gateway. Commonly, the Fuzzy Logic System (FLS) is highly flexible, and it can easily adapt to changing environments. Likewise, FLS provides effective solutions to complex issues. The number of rules needed for the FLS can grow exponentially as the complexity of the system increases, thus making the system challenging to understand and manage. To address this problem, the log-adjustable membership function is utilized in FLS. The step-by-step process of LA-FLS is derived as follows,

3.3.1. Rule Formation

Usually, to generate the fuzzy rules (F), If-Then rules are used. It is expressed as,

$$F \xrightarrow{\Xi_i} \begin{cases} if(R, D, T, G' == high\ trust\ value) & selectas\ SCH \\ if(R, D, T, G' == Second\ highest\ trust\ value) & select\ as\ VSCH \\ otherwise & none \end{cases} \quad (6)$$

The SCH and VSCH are selected based on (F). Here, R designates the RE, D indicates the degree of centrality, T represents the degree of node trust, and G' exhibits minimum distance from the gateway.

3.3.2. Log Adjustable Membership Function

Then, to easily manage and understand the system, the log-adjustable membership function is estimated. Essentially, the logarithmic adjustable function simplifies the calculations involved in estimating the membership function, thus leading to quicker execution times. It is defined as,

$$L = \begin{cases} \frac{log(\Xi_i - w + 1)}{log(v - w + 1)} & if\ w \leq \Xi_i \leq v \\ 0 & otherwise \end{cases} \quad (7)$$

Here, L implies the log adjustable membership function, w depicts the interval's lower bound, and v signifies the interval's upper bound.

3.3.3. Rule Generation Unit

Then, the rule generation unit executes interference operations, such as fuzzification and defuzzification.

3.3.4. Fuzzification Interference Operation

The fuzzy data (F) are converted into crisp data (ϖ) in fuzzification (Φ) and are mathematically equated as,

$$\Phi = (F \rightarrow \varpi) \quad (8)$$

The membership function is plotted according to the crisp data (ϖ).

3.3.5. Defuzzification Interference Operation

Next, the defuzzification $(\Im)$ is performed. It is the inverse operation of fuzzification. Here, the crisp data (ϖ) are converted into fuzzy data (F) and are determined as,

$$\Im = (\varpi \rightarrow F) \quad (9)$$

Finally, the selected SCH and VSCH (δ) are given as,

$$\delta \rightarrow (S_x, V_y) \quad (10)$$

Here, S_x indicates the SCH and V_y states the VSCH. They send a hello packet and broadcast from time to time to alert the adjacent nodes after selecting the SCH and VSCH nodes.

3.4. Hop Count

Then, between the source and destination SCH nodes (S_x), the hop count is computed. Hop count is the number of nodes a data packet traverses to attain its destination. The estimated hop count is signified as H_{sch}. Usually, the node that transmits the packet from the farthest distance has huge hop counts to reach the destination, thus resulting in more energy consumption. Thus, the node scheduling is performed and is explained in the following section.

3.5. Node Scheduling

The SCH nodes (S_x) are scheduled according to the hop count (H_{sch}) to avoid more energy consumption. Here, for node scheduling, the LHFSA is used. The First Scheduling Algorithm (FSA) scales well with the number of nodes. The scheduling complexity grows logarithmically as the network size increases. To overcome this problem, the logarithmic function is included in FSA. The step-by-step process of LHFSA is derived below,

Here, the nodes, which are closer to the destination, are preferred. The weights (Y_{S_d,S_e}) are assigned as,

$$Y_{S_d,S_e} \xrightarrow{S_x} \frac{1}{log(H_{sch}(S_d,S_e)+1)} \quad (11)$$

Here, $H_{sch}(S_d, S_e)$ designates the hop count from node S_d to destination node S_e. Then, regarding the weight (Y_{S_d,S_e}) and RE of the transmitting node $Res\,(S_d)$, the priority (Pr_{S_d,S_e}) of the link (S_d, S_e) is computed. Here, to diminish the scheduling complexities, the logarithmic function is included.

$$Pr_{S_d,S_e} = log\left(Y_{S_d,S_e} \bullet Res\,(S_d)\right) \quad (12)$$

Next, for each transmitting node S_d, the highest priority (H_p) link is selected.

$$H_p = max(Pr_{S_d,S_e}) \quad (13)$$

Then, the RE of the transmitting node is updated, and the scheduling process is continued for all transmissions. The scheduled nodes (ζ_v) are indicated as,

$$\zeta_v = [\zeta_1, \zeta_2, \zeta_3, \dots\dots, \zeta_m] \quad (14)$$

Here, $v = (1,2, \dots, m)$ indicates the number of (ζ_v).

3.6. Route Generation

Next, the possible routes for the transmission of the packets are generated (ζ_v) by the AODV protocol. Generally, the AODV protocol provides loop-free routes, and it can quickly adapt to topological changes. The AODV protocol's working is described as follows,

Initially, the route discovery process is initiated; here, the Route Request (RREQ) message is disseminated by the source node to its neighbors. In the RREQ, each intermediate node increments the Hop Count (HC) before forwarding it.

$$HC = HC + 1 \tag{15}$$

Next, when the RREQ attains the destination, the source node sends the Route Reply (RREP). Here, the lifetime (Z) of the route is computed by the destination, and the lifetime is updated.

$$Z = max(Z_{old}, Current\ Time + Active\ Route\ Timeout) \tag{16}$$

Here, Z_{old} implies the old lifetime. AODV maintains them while the routes are active. AODV sends a Route Error (RERR) message to the affected nodes if any links are broken. Whenever the RREQ, RREP, and RERR are received by the nodes, the routing table is updated. The generated routes are denoted as G_r.

3.7. Data Sensing

Now, the data are sensed from the SNs; then, to reduce energy consumption, the sensed data is transmitted to the end user through the optimal path. The sensed data (γ_j) are specified as,

$$\gamma_j \rightarrow \langle \gamma_1, \gamma_2, \gamma_3, \dots\dots, \gamma_\tau \rangle \tag{17}$$

Here, γ_τ specifies the number of sensed data.

3.8. Packet Arrival Rate-based Congestion Identification

Then, centered on the packet arrival rate, the congestion of the nodes (ϑ) that exist in the generated route (G_r) is evaluated. Firstly, the packet arrival rate (P_{arr}) is estimated as,

$$P_{arr} = \frac{\lambda}{\ddot{\varpi}} \tag{18}$$

Here, λ implies the number of packets received and $\ddot{\varpi}$ depicts the time interval. Then, the nodes' congestion is identified $(P_{arr})$by employing LA-FLS. In section 3.3, the process of LA-FLS is derived. Here, the fuzzy rules are constructed as,

$$\aleph \xrightarrow{\vartheta} \begin{cases} if\ 70 \le P_{arr} < 75 & M \\ if\ 75 \le P_{arr} < 85 & C \end{cases} \tag{19}$$

The congestion identification outcomes (CE) are given as,

$$CE = \langle C, M \rangle \tag{20}$$

Here, C designates the presence of congestion and M indicates no congestion. The congestion is minimized C; otherwise, the optimal path is selected for data transmission.

3.9. Congestion Reduction

The congestion in the nodes (ϑ) is minimized by the Backpressure method if the congestion is present (C). This is because the congested node stops receiving packets from the upstream node. The backpressure method proficiently adjusts the data flow rates to prevent congestion before it becomes challenging. The backpressure method superiorly balances the load across the network by dynamically routing traffic according to queue backlogs.

Where each node ϑ_l (G_r) has a queue $Q_{\vartheta_l}(ty)$ that indicates the backlog of data at a given time ty. Then, the evolution of the queue at the node ϑ_l is computed as,

$$Q_{\vartheta_l}(ty+1) = Q_{\vartheta_l}(ty) + \sum_{\vartheta_\phi} \rho_{\vartheta_\phi \vartheta_l}(ty) - \sum_{\vartheta_\omega} \rho_{\vartheta_l \vartheta_\omega}(ty) \tag{21}$$

Here, $\rho_{\vartheta_\phi \vartheta_l}$ specifies the data received by node ϑ_l from node ϑ_ϕ and $\rho_{\vartheta_l \vartheta_\omega}$ indicates the data received by node ϑ_l from node ϑ_ω. Next, the backlog differential between nodes ϑ_ϕ and ϑ_ϕ for the commodity (cm) is evaluated.

Then, for each commodity (cm), the transmission rates are allocated regarding backlog differentials. Then, a decision rule is made. Finally, to ensure stability, the Lyapunov drift analysis is used.

Lastly, the congestion-free nodes (α_k) are specified as,

$$\alpha_k \rightarrow [\alpha_1 + \alpha_2 + \alpha_3 + \cdots \dots + \alpha_{v'}] \quad where \quad k = (1,2, \dots, v') \tag{22}$$

Here, $\alpha_{v'}$ designates the number of (α_k).

3.10. Optimal Path Selection

Finally, by using BM-SCSO, the optimal path is selected (α_k) for transmitting the data to the end user with energy efficiency. Generally, Sand Cat Swarm Optimization (SCSO) excellently balances exploration and exploitation, leading to a robust search process.

Likewise, it superiorly simulates the sand cat's low-frequency hearing, thus providing effective solutions in the search space. However, for more complicated issues, SCSO may exhibit slow convergence. To avoid this problem, the

Baker map function is used in SCSO. The process of BM-SCSO is explained below,

3.10.1. Population Initialization

Initially, the sand cat's population is initialized; here, the congestion-free nodes are considered as the initialized population. The population matrix (J) is given as,

$$J \xrightarrow{\alpha_k} \begin{bmatrix} J_{1,1} & J_{1,2} & \vdots & J_{1,dm} \\ J_{2,1} & J_{2,2} & \vdots & J_{2,dm} \\ \vdots & \vdots & \vdots & \vdots \\ J_{K,1} & J_{K,2} & \vdots & J_{K,dm} \end{bmatrix} \quad (23)$$

Here, $J_{K,dm}$ specifies the position of K^{th} the sand cat with dm dimensions. Each (J) must lie between lower and upper boundaries.

3.10.2. Fitness Function

Next, the fitness function is computed; here, the minimum distance $(min(B))$ is considered as the fitness value, which is expressed as,

$$\lambda = min(B) * \lambda(J) \quad (24)$$

$$J_{best} = Best(\lambda) \quad (25)$$

Therefore, the best fitness value $(Best(\lambda))$ is indicated as the best candidate solution (J_{best}).

3.10.3. Exploration (Search for Prey)

Each sand cat senses the low frequency below 2 kHz in the exploration phase. The sensitivity (I_τ) can be computed as,

$$I_\tau = E_{sen} - \left(\frac{E_{sen} \times iter}{X}\right) \quad (26)$$

Here, E_{sen} represents the maximum sensitivity, $iter$ states the current iteration, and X indicates the maximum iteration. A novel location in the sensitivity range is discovered while searching for prey by each sand cat based on the baker map function (βm). Each sand cat searches for the position of prey based on the optimal candidate solution (J_{best}), current position $(J_{cut}(iter))$, and sensitivity range (g). It is mathematically formulated as,

$$g = I_\tau \times \beta m \quad (27)$$

$$\beta m(\alpha_1, \alpha_2) = \begin{cases} \left(2\alpha_1, \frac{\alpha_2}{2}\right) & for\ 0 \le \alpha_1 < \frac{1}{2} \\ \left(2 - 2\alpha_1, 1 - \frac{\alpha_2}{2}\right) & for\ \frac{1}{2} \le \alpha_1 < 1 \end{cases} \quad (28)$$

$$J^{new1}(iter+1) = g \times (J_{best}(iter) - \beta m \times J_{cut}(iter)) \quad (29)$$

Here, $J^{new1}(iter+1)$ signifies the updated position in the exploration phase.

3.10.4. Exploitation (Attack Prey)

For simulating the sand cat's prey-attacking behavior, the distance(J_C) between the sand cat and the prey is computed. Likewise, the prey attacked by the sand cat is equated as,

$$J_C = |rand(0,1) \times J_{best}(iter) - J_{cut}(iter)| \quad (30)$$

$$J^{new2}(iter+1) = J_{best}(iter) - g \times J_{rd} \times cos(\propto) \quad (31)$$

Here, $J^{new2}(iter+1)$ implies the updated position in the exploitation phase, J_{rd} signifies the random position, $rand(0,1)$ designates the random number within the interval of 0 and 1, and $cos(\propto)$ depicts the random angle. The sand cat attacks the prey if the random number Φ is less than or equal to 1; else, the sand cat searches for the prey. The selected optimal paths (O_t) are signified as,

$$O_t \Rightarrow [O_1, O_2, O_3, \dots\dots, O_v] \quad where \quad t = (1,2,\dots,v) \quad (32)$$

Here, O_v denotes the number of selected optimal paths(O_t). The pseudocode for BM-SCSO is displayed as follows,

Pseudocode for BM-SCSO

Input: Congestion-free nodes (α_k)
Output: Selected optimal path (O_t)

Begin
Initialize (α_k)
Set $[\boldsymbol{iter = 1}]$
While $[\boldsymbol{iter \le (iter + 1)}]$
Initialize the population of sand cat (J)
Estimate fitness function $\lambda = min(B) * \lambda(J)$
// Search for prey
Perform exploration
Compute sensitivity $I_\tau = E_{sen} - \left(\frac{E_{sen} \times iter}{X}\right)$
Find the sensitivity range. $g = I_\tau \times \beta m$
Discover the Baker map function

$$\beta m(\alpha_1, \alpha_2) = \begin{cases} \left(2\alpha_1, \frac{\alpha_2}{2}\right) & for\ 0 \le \alpha_1 < \frac{1}{2} \\ \left(2 - 2\alpha_1, 1 - \frac{\alpha_2}{2}\right) & for\ \frac{1}{2} \le \alpha_1 < 1 \end{cases}$$

Estimate new position

$$J^{new1}(iter+1) = g \times (J_{best}(iter) - \beta m \times J_{cut}(iter))$$

// Attack prey

Implement exploitation

Compute the distance between the sand cat and the prey

$$J_C = |rand(0,1) \times J_{best}(iter) - J_{cut}(iter)|$$

Find a new position

$$J^{new2}(iter+1) = J_{best}(iter) - g \times J_{rd} \times cos(\propto)$$

If $(\boldsymbol{\Phi} \leq \mathbf{1})$

Attack prey

Else

Search for prey

End If

End While

Obtain the selected optimal path (O_t)

End

Next, through the selected optimal paths(O_t), the sensed data (γ_j) are transferred.

3.11. Secure Handover to Vice Super Cluster Head Selection

The SCH's position (η) is securely handed over to VSCH (V_y) without disrupting the network operations if the SCH's energy drops. This secure handover was performed by employing the D3-OIKSA. Usually, the Digital Signature Algorithm (DSA) is computationally infeasible for an attacker to forge a signature without accessing the private key. However, when compared to other digital signature schemes, DSA signatures are relatively large, thus increasing the storage and bandwidth requirements, especially when dealing with signed data's large volumes. To solve this problem, the Doubling-oriented-Doche-Icart-Kohel curve is used.

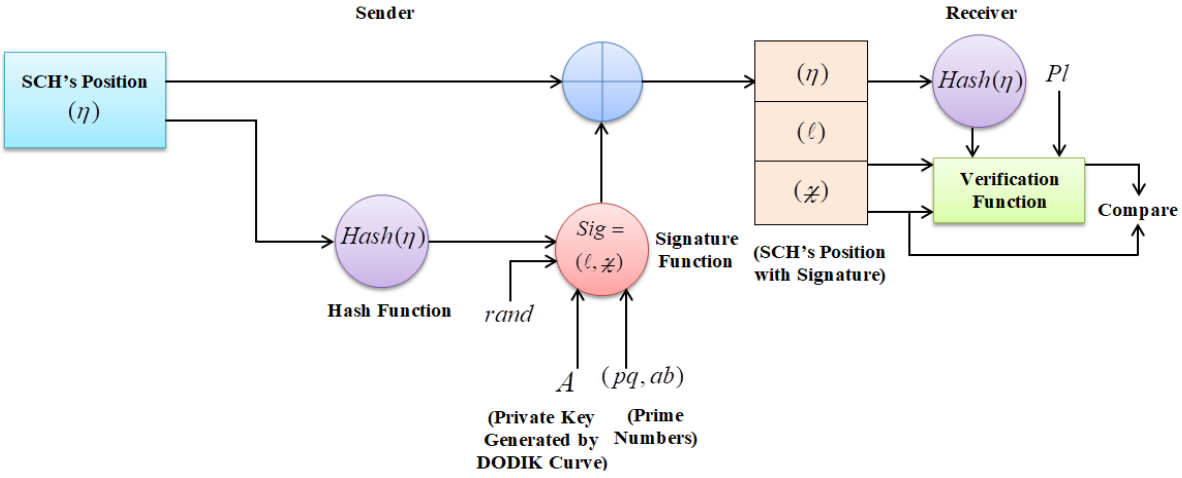


Fig. 2 Structure of D3-OIKSA

3.12. Key Generation

Initially, the key generation is performed on the sender side. Here, by employing the Doubling-oriented-Doche-Icart-Kohel curve, the private key is generated. It supports smaller key sizes compared to traditional elliptic curves. Therefore, the generated public (Pl) and private keys (A) are given as,

$$A = y^3 + \hat{v}y^2 + 16\hat{v}y - z^2 \quad (33)$$

$$Pl = gn^A \mod pq \quad (34)$$

Here, y and z indicates the coordinates on the curve, $\hat{v}$ indicates the parameter, gn indicates the generator, and pq indicates the large prime number.

3.13. Signature Generation

The SCH's position (η) is hashed as follows,

$$\hbar_{send} = Hash(\eta) \quad (35)$$

Here $\hbar_{send}$ is the hashed information on the sender side. Next, the signature(Sig) is computed as,

$$Sig = (\ell, \not{\chi}) \quad (36)$$

$$\ell = (gn^{rand} \mod pq) \mod ab \quad (37)$$

$$\not{\chi} = rand^{-1} \bullet (\hbar + A \cdot \ell) \mod ab \quad (38)$$

Where ℓ indicates the first part of the signature, $\not{\chi}$ exhibits the second part of the signature, ab specifies the smaller prime number, and $rand$ outlines the random integer. Then, the (η) signature $(\ell, \not{\chi})$ is sent to the receiver.

3.14. Signature Verification

The digital signature is verified on the receiver side. Initially, the signature components are validated; then, the hash is generated for the received (η)data.

$$\hbar_{rece} = Hash(\eta) \quad (39)$$

Here, $\hbar_{rece}$ specifies the hashed information on the receiver side. Next, the modular inverse of χ is estimated. Then, the intermediate values (b_1 and b_2) are computed. Then, φ it is evaluated, which is the reconstructed value of ℓ. Lastly, the signature is validated as,

$$SV = \begin{cases} if\ \varphi = \ell & valid\ signature \\ if\ \varphi \neq \ell & Invalid\ signature \end{cases} \quad (40)$$

Where SV denotes the signature verification. The SCH's position is handed over to VSCH (V_y) if the signature is valid; otherwise, the SCH's position is not handed over. Next, for reliable packet transmission, the remaining steps, such as hop count calculation, node scheduling, route generation, data sensing, packet arrival rate-based congestion identification, congestion reduction, and optimal path selection, are processed. Therefore, the proposed framework excellently enhanced the energy efficiency in WSN.

4. Result and Discussion

Here, the proposed model's performance is validated, and a comparative analysis is carried out to highlight its trustworthiness and efficiency. The proposed model is implemented in the working platform of MATLAB/Simulation Platform.

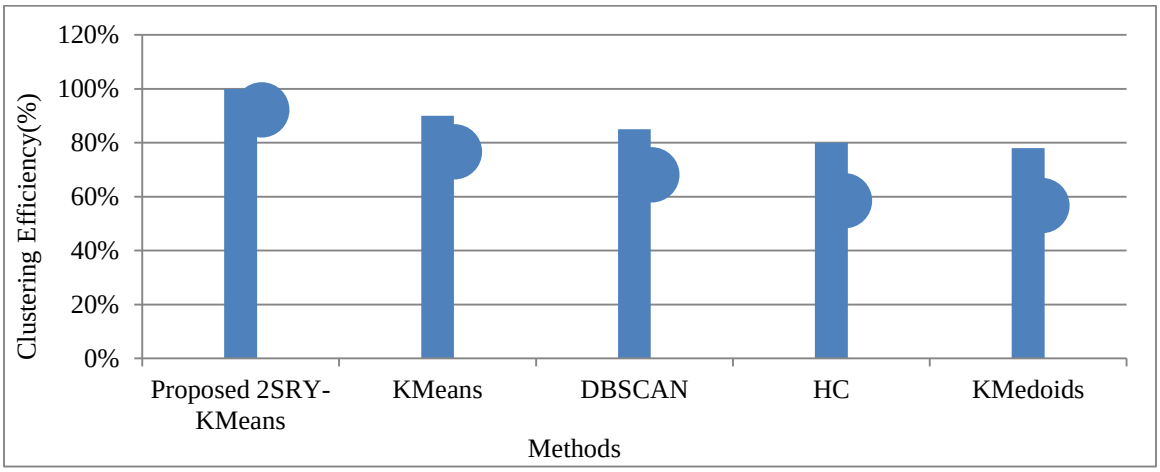
4.1. Performance Evaluation

Here, the performance of the proposed and conventional techniques is evaluated at different phases (I to III) of the algorithm to demonstrate the proposed model's reliability.

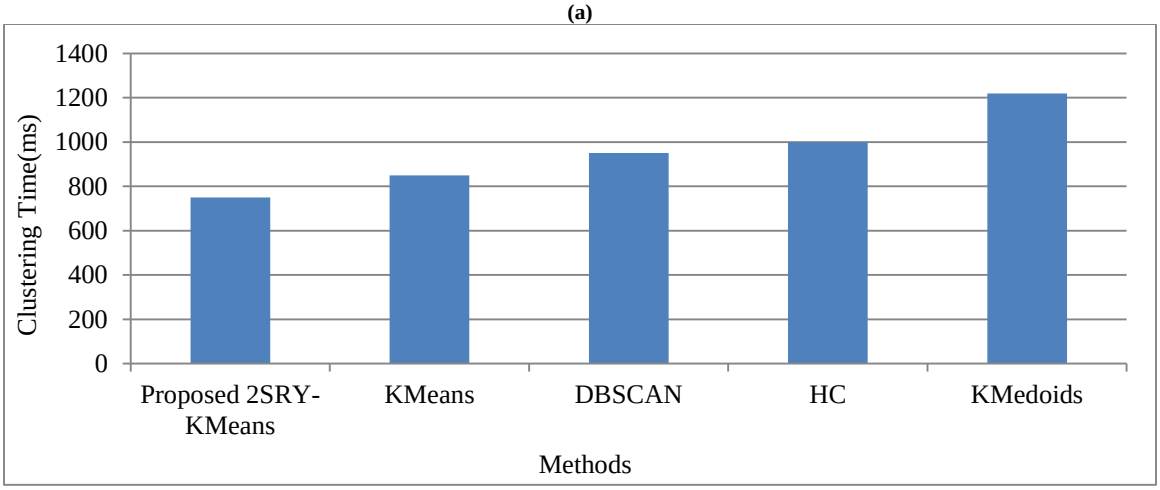
Phase I: Clustering and selection of Cluster Head

The performance assessment of the proposed LA-FLS and conventional techniques is depicted in Figure 3.

For fuzzification, defuzzification, and rule generation, the proposed LA-FLS achieved a low value of 0.0033598s, 0.0097118s, and 0.021741s; while the conventional Trapezoidal-FLS (Trap-FLS), Sigmoid-FLS (Sig-FLS), Triangular-FLS (Tri-FLS), and Decision Rule techniques obtained a high mean value of 0.5677819s, 0.4589238s, and 0.2893157s, respectively. Thus, due to the usage of the log-adjustable membership function, the proposed model efficiently selected the SCH and VSCH.



(a)



(b)

Fig. 3 Performance analysis regarding, (a) Clustering efficiency, and (b) Clustering time.

In Table 2, the total energy consumption analysis of the proposed LA-FLS and existing methods is shown. The proposed LA-FLS consumed less total energy of 0.075771 Joule (J) with the usage of a log adjustable membership function; whereas the existing Trap-FLS, Sig-FLS, Tri-FLS, and Decision Rule techniques consumed a huge total energy of 0.085495J, 0.089506J, 0.094083J, and 0.095803J, respectively. Therefore, the proposed model's reliability is proven.

The performance analysis of the proposed 2SRY-KMeans and existing techniques regarding clustering efficiency and clustering time is depicted in Figure 3. Here, the proposed 2SRY-KMeans achieved a high clustering efficiency of 99% and a low clustering time of 750ms.

Likewise, the existing KMeans, Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Hierarchical Clustering (HC), and KMedoids attained a low clustering efficiency of 90.20%, 85%, 80%, and 78%, respectively.

Likewise, the existing KMeans, DBSCAN, HC, and KMedoids obtained a high clustering time of 845ms, 949ms, 1008ms, and 1220ms, respectively. Therefore, the proposed 2SRY-KMeans efficiently reduced the redundant nodes with the usage of Sinusoidal Representation initialization and Yule's sigma distance.

The performance evaluation of the proposed 2SRY-KMeans and conventional techniques is depicted in Table 3. Here, to efficiently reduce the redundant nodes, the Sinusoidal Representation initialization and Yule's sigma distance are modified with KMeans. The proposed 2SRY-K-Means achieved a high silhouette score of 0.896 and a low Davies-Bouldin index of 0.012.

Nevertheless, the conventional KMeans, DBSCAN, HC, and KMedoids attained a low average silhouette score of 0.6973444 and a high average Davies-Bouldin index of 0.04025. Therefore, when analogized to the prevailing techniques, the proposed 2SRY-K-Means achieved better performance.

Phase II: Selection of optimal Path for Data Routing

The comparative assessment of the proposed BM-SCSO and prevailing techniques regarding throughput, optimization time, and NL is displayed in Tables 4 and 5.

Here, for 100, 200, 300, 400, and 500 nodes, the proposed BM-SCSO achieved a high throughput of 56825.74916 bytes, a low optimization time of 1275ms, and a high NL of 1300 rounds, 1259 rounds, 1192 rounds, 1158 rounds, and 1087 rounds, respectively. Likewise, the prevailing SCSOs, Elephant Herding Optimization (EHO), Harris Hawk Optimization (HHO), and Sparrow Search Optimization (SSO) attained a low throughput of 51200 bytes, 46545.45455 bytes, 34133.33333 bytes, and 28444.44444 bytes, respectively.

Likewise, the prevailing techniques obtained high optimization time and low NL. To solve the slower convergence problem, the Baker map function is employed in BM-SCSO; thus, the proposed model proficiently selected the optimal paths.

Table 2. Total energy consumption analysis

Methods	Total Energy Consumption(J)
Proposed LA-FLS	0.075771
Trap-FLS	0.085495
Sig-FLS	0.089506
Tri-FLS	0.094083
Decision Rule	0.095803

Table 3. Performance evaluation based on performance metrics

Techniques	Silhouette Score	Davies-Bouldin Index
Proposed 2SRY-KMeans	0.896	0.012
KMeans	0.876491881	0.033
DBSCAN	0.78	0.041
HC	0.594298845	0.042
KMedoids	0.538586942	0.045

Table 4. Comparative assessment in terms of throughput and optimization time

Techniques	Throughput (bytes)	Optimization Time (ms)
Proposed BM-SCSO	56825.74916	1275
SCSO	51200	3254
EHO	46545.45455	4517
HHO	34133.33333	5124
SSO	28444.44444	6547

Table 5. Network lifetime analysis

Techniques / Number of nodes	100	200	300	400	500
Proposed BM-SCSO	1300	1259	1192	1158	1087
SCSO	1262	1185	1155	1081	1039
EHO	1198	1147	1091	1042	981
HHO	1153	1079	1059	986	949
SSO	1095	1037	990	954	901

To avoid slower convergence issues, the proposed BM-SCSO utilized the Baker map function, thus obtaining reduced latency. In Figure 4, the latency validation of the proposed and existing techniques is shown. For latency, the proposed BM-SCSO achieved a low value of 0.901 seconds (s), whereas the existing SCSO, EHO, HHO, and SSO attained a high value of 1.001s, 1.101s, 1.501s, and 1.801s, respectively.

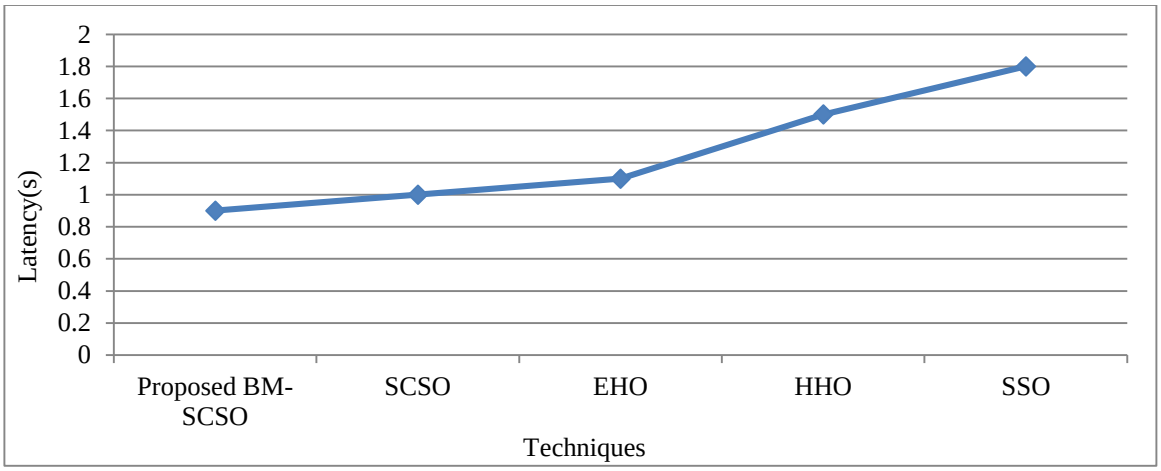


Fig. 4 Latency evaluation

Therefore, when analogized to existing techniques, the proposed BM-SCSO selected the optimal paths with lower latency. The time taken by the algorithm during the calculation of fuzzy fitness functions is also depicted in Figure 5.

Phase III: Secure Handover from SCH to VSCH

The graphical representation of the proposed D3-OIKSA and existing techniques regarding attack level and security level is depicted in Figure 6. The proposed D3-OIKSA achieved a low attack level of 5% and a high-security level of 95% with the inclusion of the Doubling-oriented-Doche-Icart-Kohel curve. Nevertheless, the existing techniques like DSA, Advanced Encryption Standard (AES), Elliptic Curve Cryptography (ECC), along with Rivest-Shamir-Adleman (RSA), obtained a high average attack level of 11.75% and a low mean security level of 88.25%.

Hence, the proposed model effectively transmitted the data by securely handing over the position of SCH to VSCH while the SCH energy was dropped. In Figure 7, performance validation of the proposed D3-OIKSA and prevailing techniques with respect to memory usage is displayed. Here, for key generation, the Doubling-oriented-Doche-Icart-Kohel curve is employed, thus reducing the storage and bandwidth requirements.

For encryption and decryption, the D3-OIKSA used less memory of 965 kilobytes (kB) and 1264kB; while the existing AES and RSA used a huge memory of 2188kB and 2928kB for encryption. Moreover, the prevailing DSA and ECC utilized a huge memory of 1553kB and 2768kB for decryption. Therefore, the proposed model's effectiveness is proven.

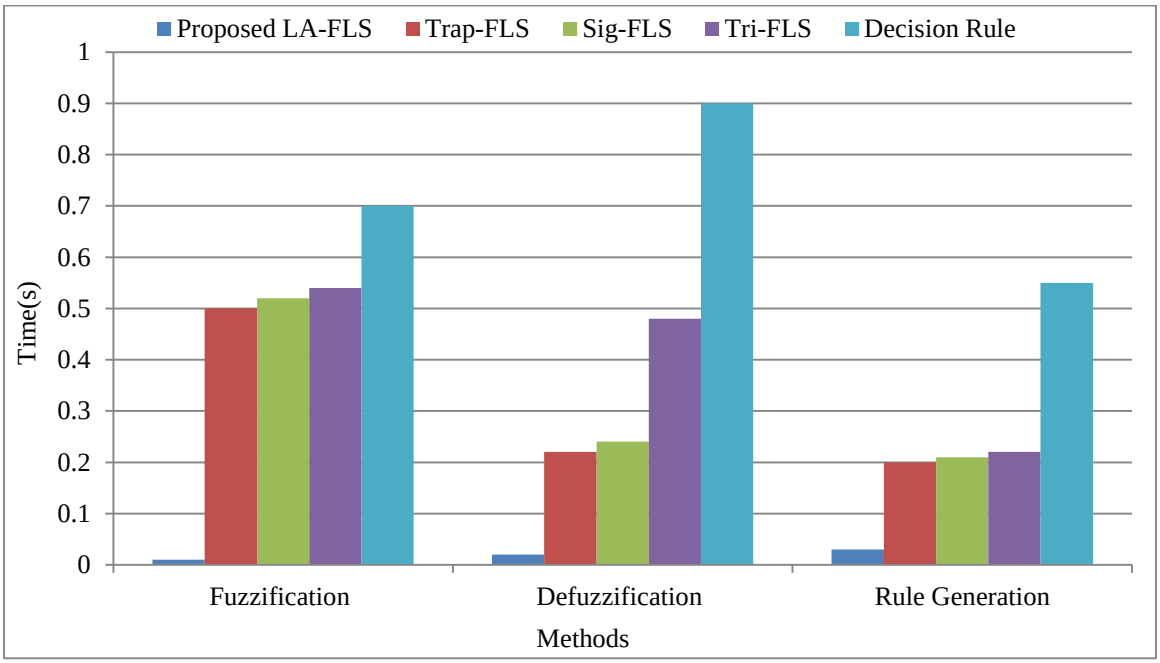
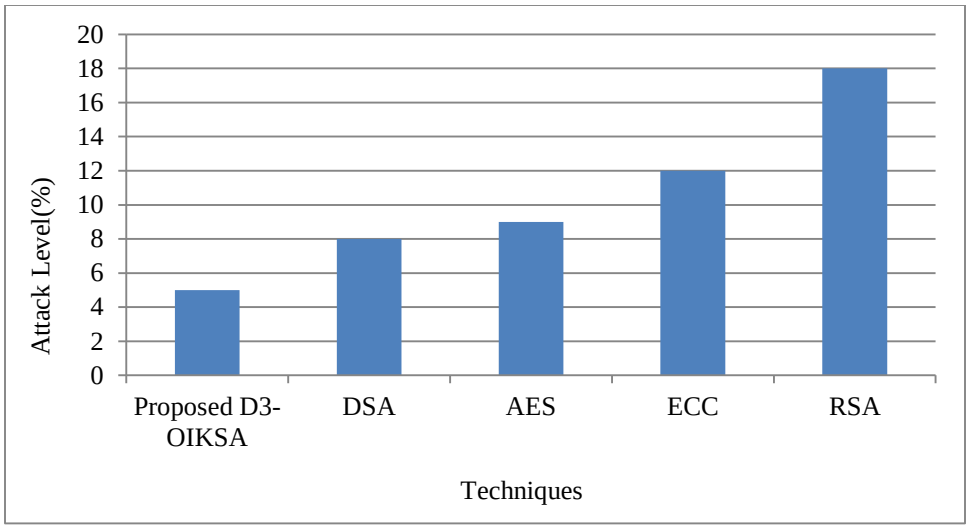
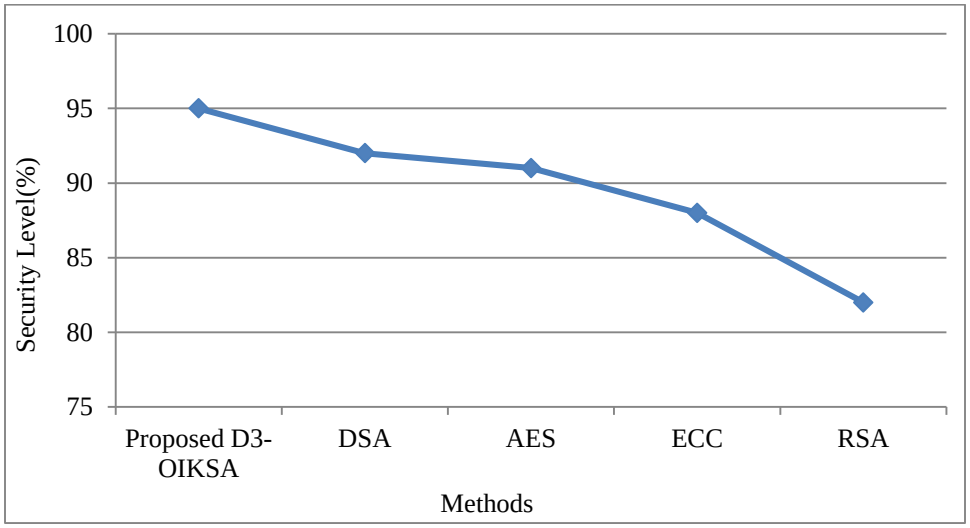


Fig. 5 Performance assessment based on performance metrics



(a)



(b)

Fig. 6 Graphical representation regarding (a) Attack level, (b) Security level.

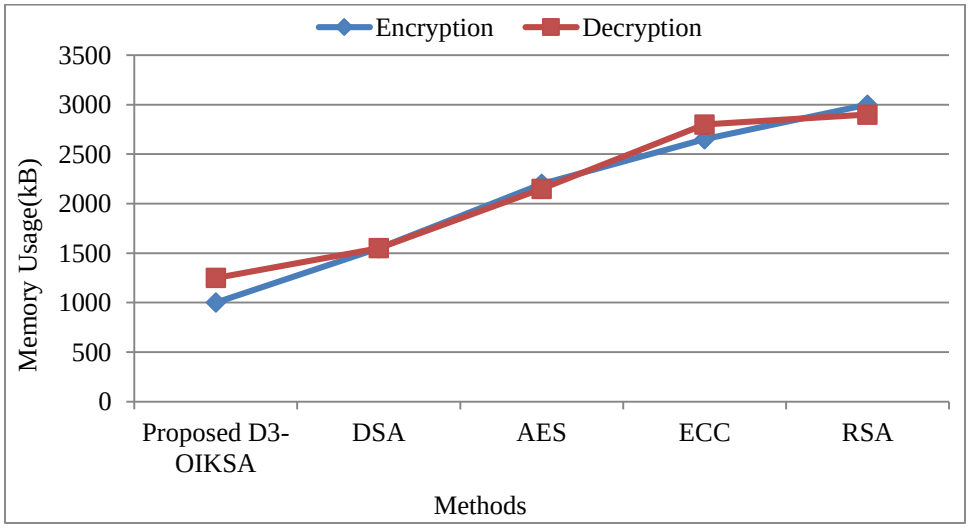


Fig. 7 Performance validation with respect to memory usage

Table 6. Comparative analysis

Author's name	Technique	Throughput (bytes)	Network Lifetime (rounds)
Proposed Model	BM-SCSO	56825.74916	1300
Rathore et al. [26]	SPSRN	43618.88163	-
Adumbabu & Selvakumar [27]	ICOA and IJO-LF	18125	1200
Bouazzi et al. [28]	Fuzzy and CSMA/CA	5400	-
Al-Kaseem et al. [29]	MOEAs	-	1000
Prithi & Sumathi [30]	PSO–GWO	19500	1000

4.2. Comparative Analysis

Here, to prove the overall dependability of the proposed model, a comparative analysis is conducted for the proposed and related works. The proposed technique's performance is weighed against the prevailing works in Table 6. Here, for 100 nodes, the proposed BM-SCSO achieved high throughput and NL of 56825.74916 bytes and 1300 rounds, respectively, owing to the inclusion of the Baker map function.Similarly, the existing Shortest Path Selection for Relay Node (SPSRN) and Fuzzy algorithm, and the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) algorithm attained low throughputs of 43618.88163 bytes and 5400 bytes, respectively. Likewise, for 100 numbers of nodes, the prevailing Improved Coyote Optimization Algorithm (ICOA), Improved Jaya Optimization Algorithm with Levy Flight (IJO-LF), Multi-Objective Evolutionary Algorithms (MOEAs), and Hybrid Particle Swarm Optimization–Grey Wolf Optimizer (PSO–GWO) attained a low NL of 1200 rounds, 1000 rounds, and 1000 rounds, respectively. Therefore, the proposed model's effectiveness is proven.

5. Conclusion

Here, this paper presents EE optimal path selection in WSN using LA-FLS and BM-SCSO. Here, important processes, such as redundant SN reduction, SCH and VSCH selection, secure handover to VSCH, node scheduling, route generation, congestion estimation and reduction, and optimal path selection were carried out. Here, the proposed D3-OIKSA achieved a high-security level of 95% and a low attack level of 5%, which proved the reliability of the proposed model. Similarly, for optimal path selection, the proposed BM-SCSO attained a high throughput of 56825.74916 bytes and a low latency of 0.901s. Likewise, the proposed La-FLS consumed a lower total energy of 0.075771J for SCH-VSCH selection. Also, the proposed 2SRY-K-Means took less time, 750ms for clustering. Thus, the reliability and trustworthiness are improved by the proposed model. The proposed model failed to focus on model invasion attacks, although the proposed model only concentrated on secure handover to VSCH during the drop in the SCH energy.

Author Contribution

Both authors, Ab Wahid Bhat and Abhiruchi Passi, actively contributed to data collection, analysis, interpretation, and Drafting of the manuscript**.**

Availability of data and material

All data generated or analysed during this study are included in this published article [and its supplementary information files].

References

[1] Bryan Raj et al., "A Survey on Cluster Head Selection and Cluster Formation Methods in Wireless Sensor Networks," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, pp. 1-53, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Rachid Zagrouba, and Amine Kardi, "Comparative Study of Energy Efficient Routing Techniques in Wireless Sensor Networks," *Information*, vol. 12, no. 1, pp. 1-28, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] Roopali Dogra et al., "Energy-Efficient Routing Protocol for Next-Generation Application in the Internet of Things and Wireless Sensor Networks," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, pp. 1-10, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Pradeep Sadashiv Khot, and Udaykumar Naik, "Particle-Water Wave Optimization for Secure Routing in Wireless Sensor Network using Cluster Head Selection," *Wireless Personal Communications*, vol. 119, no. 3, pp. 2405-2429, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Piyush Rawat, and Siddhartha Chauhan, "Particle Swarm Optimization-based Energy Efficient Clustering Protocol in Wireless Sensor Network," *Neural Computing and Applications*, vol. 33, no. 21, pp. 14147-14165, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[6] Wisal Bassim Nedham, and Ali Kadhum M. Al-Quraba, "An Improved Energy Efficient Clustering Protocol for Wireless Sensor Networks," *2022 International Conference for Natural and Applied Sciences (ICNAS)*, Baghdad, Iraq, pp. 22-28, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Xiuniao Zhao, Wentao Zhong, and Yahya Dorostkar Navaei, "A Novel Energy-Aware Routing in Wireless Sensor Network using Clustering based on Combination of Multiobjective Genetic and Cuckoo Search Algorithm," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, pp. 1-14, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] K. SureshKumar, & P. Vimala, "Energy Efficient Routing Protocol using Exponentially-Ant Lion Whale Optimization Algorithm in Wireless Sensor Networks," *Computer Networks*, vol. 197, pp. 1-12, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] G.A. Senthil, Arun Raaza, and N. Kumar, "Internet of Things Energy Efficient Cluster-based Routing using Hybrid Particle Swarm Optimization for Wireless Sensor Network," *Wireless Personal Communications*, vol. 122, no. 3, pp. 2603-2619, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] Amir Seyyedabbasi et al., "Optimal Data Transmission and Pathfinding for WSN and Decentralized IoT Systems using I-GWO and Ex-GWO Algorithms," *Alexandria Engineering Journal*, vol. 63, pp. 339-357, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Mandli Rami Reddy et al., "Energy-Efficient Cluster Head Selection in Wireless Sensor Networks using an Improved Grey Wolf Optimization Algorithm," *Computers*, vol. 12, no. 2, pp. 1-17, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] Sathyapriya Loganathan, and Jawahar Arumugam, "Energy Efficient Clustering Algorithm based on Particle Swarm Optimization Technique for Wireless Sensor Networks," *Wireless Personal Communications*, vol. 119, no. 1, pp. 815-843, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Elvis Obi, Zoubir Mammeri, and Okechukwu E. Ochia, "A Centralized Routing for Lifetime and Energy Optimization in WSNs using Genetic Algorithm and Least-Square Policy Iteration," *Computers*, vol. 12, no. 2, pp. 1-28, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[14] S. Radhika, and P. Rangarajan, "Fuzzy based Sleep Scheduling Algorithm with Machine Learning Techniques to Enhance Energy Efficiency in Wireless Sensor Networks," *Wireless Personal Communications*, vol. 118, no. 4, pp. 3025-3044, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[15] Oluwasegun Julius Aroba, Nalindren Naicker, and Timothy Adeliyi, "An Innovative Hyperheuristic, Gaussian Clustering Scheme for Energy-Efficient Optimization in Wireless Sensor Networks," *Journal of Sensors*, vol. 2021, no. 1, pp. 1-12, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[16] Da-Wen Huang et al., "An Efficient Hybrid IDS Deployment Architecture for Multi-Hop Clustered Wireless Sensor Networks," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 2688-2702, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Sehar Umbreen et al., "An Energy-Efficient Mobility-based Cluster Head Selection for Lifetime Enhancement of Wireless Sensor Networks," *IEEE Access*, vol. 8, pp. 207779-207793, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[18] Yun Xu, Wanguo Jiao, and Mengqiu Tian, "Energy-Efficient Connected-Coverage Scheme in Wireless Sensor Networks," *Sensors*, vol. 20, no. 21, pp. 1-19, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[19] Kale Navnath Dattatraya, and K. Raghava Rao, "Hybrid based Cluster Head Selection for Maximizing Network Lifetime and Energy Efficiency in WSN," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 3, pp. 716-726, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[20] Panimalar Kathiroli, and Kanmani Selvadurai, "Energy efficient cluster head selection using improved Sparrow Search Algorithm in Wireless Sensor Networks," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 10, pp. 8564-8575, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] P. Divya, and B. Sudhakar, "Route Optimization and Optimal Cluster Head Selection for Cluster-Oriented Wireless Sensor Network Utilizing Circle-Inspired Optimization Algorithm," *International Journal of Computational Intelligence Systems*, vol. 17, no. 1, pp. 1-15, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] Regonda Nagaraju et al., "Secure Routing-based Energy Optimization for IoT Application with Heterogeneous Wireless Sensor Networks," *Energies*, vol. 15, no. 13, pp. 1-16, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[23] G.C. Jagan, and P. Jesu Jayarin, "Wireless Sensor Network Cluster Head Selection and Short Routing using Energy Efficient ElectroStatic Discharge Algorithm," *Journal of Engineering*, vol. 2022, pp. 1-10, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[24] Omkar Singh, Vinay Rishiwal, and Mano Yadav, "Multi-Objective Lion Optimization for Energy-Efficient Multi-Path Routing Protocol for Wireless Sensor Networks," *International Journal of Communication Systems*, vol. 34, no. 17, pp. 1-15, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25] Huangshui Hu et al., "Trust based Secure and Energy Efficient Routing Protocol for Wireless Sensor Networks," *IEEE Access*, vol. 10, pp. 10585-10596, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[26] Pramod Singh Rathore et al., "Energy-Efficient Cluster Head Selection through Relay Approach for WSN," *Journal of Supercomputing*, vol. 77, no. 7, pp. 7649-7675, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[27] I. Adumbabu, and K. Selvakumar, "Energy Efficient Routing and Dynamic Cluster Head Selection using Enhanced Optimization Algorithms for Wireless Sensor Networks," *Energies*, vol. 15, no. 21, pp. 1-18, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[28] Imen Bouazzi et al., "A New Medium Access Control Mechanism for Energy Optimization in WSN: Traffic Control and Data Priority Scheme," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 1-23, 2021. [CrossRef] [Google Scholar] [Publisher Link]

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

[30] S. Prithi, and S. Sumathi, "Automata based Hybrid PSO-GWO Algorithm for Secured Energy Efficient Optimal Routing in Wireless Sensor Network," *Wireless Personal Communications*, vol. 117, no. 2, pp. 545-559, 2021. [CrossRef] [Google Scholar] [Publisher Link]