*I. J. Computer Network and Information Security,* **2026, 1, 33-50**
Published Online on February 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2026.01.03

# Secure Blockchain-based Routing with Narwhal Optimization for WSNs

**Ranjeet Yadav***
Electronics and Communication Engineering, Government Polytechnic Daman, UT Administration of Dadra and Nagar Haveli and Daman and Diu 396210, India
E-mail: ranjeet480@gmail.com
ORCID iD: https://orcid.org/0000-0003-0891-8556
*Corresponding author

**N. Manimegalai**
Department of Artificial Intelligence and Data Science, Velammal Engineering College, Chennai – 66, Tamil Nadu, India
E-mail: manimalu15@gmail.com
ORCID iD: https://orcid.org/0009-0005-4530-4512

**Mercy Beulah E.**
Department of Computer Science and Engineering, Veltech Multitech Dr. Rangarajan Dr. Sakunthala Engineering College, Tamil Nadu, India
E-mail: mercybeulah@veltechmultitech.org

**Mohammed Al-Farouni**
Department of computers Techniques engineering, College of technical engineering, The Islamic University, Najaf, Iraq
E-mail: mohammed45alfa@outlook.com
ORCID iD: https://orcid.org/0009-0007-9735-3783

Received: 27 August 2025; Revised: 25 September 2025; Accepted: 20 October 2025; Published: 08 February 2026

**Abstract:** Wireless Sensor Networks (WSNs) play a crucial role in various domains, such as environmental monitoring, health, and military applications. These applications necessitate the establishment of secure and efficient communication. This network encounters a major issue since routing attacks along with data tampering are highly prevalent in such networks due to their decentralized architecture and limited resources for computation that make the networks susceptible to a wide range of security threats. The existing techniques-WSN-Block, CEMT, TSRP, ORASWSN, POA-DL, AI-WSN, and EOSR are extensively used for routing but experience inefficiencies in optimizing paths that increases energy consumption drastically and also allows packet loss. In addition, the existing blockchain models for WSN security are not scalable; have high overheads of computation. To address these limitations, we propose the Secure Blockchain-Based Routing with Narwhal Optimization for WSNs (OpNa-SGCDN). Our approach employs Optimized Narwhal-Based Metaheuristic for guaranteed shortest-path communication and minimum energy consumption in optimum routing. Moreover, we provide Scalable Permissionless Blockchain Consensus Model (SP-BlockCM) features enhanced to yield a decentralized solution that is tamper-proof but with improved scalability. The attack detection function is designed by making use of a Stacked Bi-Tier Convolutional Deep Network (SBT-CDN), which is optimized by the Snow Geese Evolutionary Algorithm (SGEA). Experimental results demonstrate that our method improves energy efficiency with 94.7% as well as achieves higher detection accuracy for 96.3% and packet loss for 95.5%, both security and performance, and thus it is better than the available methods. The framework given here is thus obviously comprehensive as well as scalable for secure, energy-efficient WSN communication.

**Index Terms:** WSN, Routing Attacks, Blokchain Integration, Routing Protocol, Attack Detetction, Optimization, Packet Loss.

## 1. Introduction

Routing attacks remain among the most pressing security challenges in Wireless Sensor Networks (WSNs), due to

 

the decentralized architecture and resource-constrained nature of sensor nodes [1]. These attacks can compromise communication paths, degrade network performance, and threaten the confidentiality and integrity of sensitive data [2]. Ensuring secure routing is essential for the reliability of applications that rely on real-time and accurate data delivery in WSNs [3]. The inherent limitations of WSNs, including limited processing power and energy resources, expose them to various types of routing attacks that can disrupt system functionality [4]. Malicious activities such as packet drops, data alterations, and misrouting can result in significant network instability and communication loss [5].

Attacks like blackhole and wormhole are particularly damaging, where compromised nodes mislead others to route data through them, only to discard or manipulate the information [6]. In blackhole attacks, a malicious node falsely advertises the shortest path to attract traffic, eventually causing data loss and network congestion [7]. Wormhole attacks involve two malicious nodes that create a covert tunnel, distorting the network's topology to intercept or manipulate traffic [8]. To counter such threats, trust-based mechanisms have been employed, where each node's behavior is continuously evaluated to assign trust values [9]. Nodes exhibiting anomalous behaviors, such as consistent packet dropping or irregular routing decisions, are isolated from the network [10].

Blockchain technology provides a decentralized, tamper-resistant mechanism for enhancing routing security by verifying node authenticity and preserving routing histories [11]. Permissionless and scalable blockchain models can be effectively integrated to manage WSN routing decisions without the need for a central authority [12]. In smart city environments, securing WSNs is critical to prevent disruptions in traffic management, emergency services, and energy distribution [13]. A blockchain-based routing framework not only secures data transmission but also enhances trust among nodes through immutable records of behavior [14]. Moreover, integrating blockchain with anomaly detection models significantly improves resilience against adaptive attack strategies [15].

By combining trust management, blockchain consensus mechanisms, and intelligent detection techniques, WSNs can achieve enhanced protection against a wide range of routing attacks, ensuring reliable operation in critical applications.

*Novelty and Contributions*

This research introduces a multi-stage defense architecture for Wireless Sensor Networks (WSNs) by integrating four distinct yet complementary components: Narwhal-based metaheuristic optimization, a scalable permissionless blockchain model (SP-BlockCM), a deep convolutional detection network (SBT-CDN), and the Snow Geese Evolutionary Algorithm (SGEA). While these components have been individually explored in previous studies, their combined orchestration in a unified pipeline is novel. The Narwhal optimization model ensures rapid convergence toward energy-efficient routes. SP-BlockCM provides a tamper-resistant, decentralized trust model that scales efficiently without centralized control. SBT-CDN offers a two-tier deep learning mechanism for anomaly detection in routing behavior, and SGEA fine-tunes the network to improve detection accuracy and efficiency. The sequential coupling of routing, trust establishment, and attack detection within a single framework is uniquely positioned to address the concurrent challenges of latency, security, and energy consumption in WSNs. This integrated design not only enhances the system's resilience but also ensures scalable, real-time defensive capability in dynamic and resource-constrained environments.

- We propose the Optimized Narwhal-Based Metaheuristic (OpNaBM) to efficiently find the shortest path in WSNs, optimized for both minimal energy usage and latency.
- The Scalable Permissionless Blockchain Consensus Model (SP-BlockCM) integrates blockchain to ensure secure, decentralized data exchange in WSNs without relying on a central authority, preventing malicious routing behaviour.
- We introduce the Stacked Bi-Tier Convolutional Deep Network (SBT-CDN), which detect routing attacks with high accuracy to analyse patterns of network traffic based on two-tier architecture.
- The Snow Geese Evolutionary Algorithm (SGEA) optimizes the SBT-CDN's parameters, improving attack detection accuracy and reducing computational complexity, making it ideal for real-time WSN applications.

Our research aims to create an integrated framework that encompasses the power of metaheuristic optimization, blockchain-based trustless security, and deep learning-based attack detection to improve the performance, security, and scalability of the WSN in dynamic and resource-constrained environments.

This paper is structured as: Section 2, the literature review is presented; in Section 3, the suggested methodology is established; in Section 4, the findings are verified; and in Section 5, the work is concluded.

## 2. Literature Survey

In 2021, Almaiah, M.A., [16], proposed a new scheme for detecting malicious attacks in wireless sensor networks based on blockchain technology. It proposes a new blockchain-based approach to detect malware attacks on WSNs. Sensitive information is exchanged across different smart devices and sensor nodes in widely used applications of transportation, intelligent buildings, healthcare, and intelligent cities through WSN. Those data are prone to different kinds of cyber-attacks. Hence, the proposed approach includes a merger of blockchain with heuristic, signature-based,

and voting detection approaches. This method enhances the general security of the WSN environment since the cluster head node (CHN) will identify and neutralize rogue sensor nodes based on node-signatures, node-hash values, and voting degrees.

In 2021, Feroz Khan, A.B. et al. [17], presented a cognitive energy efficient and trusted routing model for the security of wireless sensor networks: CEMT. A cognitive energy-efficient and trusted routing model (CEMT) is developed within this study to protect wireless sensor networks. The above model utilizes a multi-attribute-based routing scheme to optimize the energy usage, and at the same time, it ensures secure data transfer. CEMT has also improved the network performance, and thus it shows better efficiency than those similar techniques widely being used nowadays like ant-colony optimization and multi-attribute pheromone ant secure routing algorithm. Superior security based on advanced coincidence rates detection and nodes computation algorithms make this one superior. This does the computation of trust by using the QoS characteristics, such as stability rate, detection time for impersonation attacks, and reliability; hence there is heavy defence against malicious activity in the network.

In 2021, Hu, H., et al. [18], introduced Trust-aware secure routing protocol for wireless sensor networks. The protocol prevents black hole, selective forwarding, wormhole, hello flood, and sinkhole attacks because it calculates comprehensive trust values for all nodes using direct trust, indirect trust, a volatilization factor, and residual energy. The source node sends routing requests in multi-path way to the neighbours till it reaches the sink where the actual transmission of data starts. In order to guarantee secure data routing, the sink node then determines the best route by examining trust values, transmission distance, and hop count.

In 2021, Dora, J.R. et al. [19], suggested Clone node detection attacks and mitigation mechanisms in static wireless sensor networks. The present study delves into the topic of clone node assaults and their countermeasures in static Wireless Sensor Networks (WSNs). WSNs have become more and more popular because of its extensive uses in target tracking, pressure monitoring, fire detection, and health monitoring. Although WSNs have numerous advantages, they are vulnerable to the replication or clone node attack; especially if they are deployed in undesirable areas without physical security. The attackers take down a few credible sensor nodes and then steal private data like the IDs and keys of the nodes. Then they replicate the nodes. There are several methods that have been developed to counter this.

In 2021, Abd El-Moghith, I.A. et al. [20], recommended towards designing a trusted routing scheme in wireless sensor networks: A new deep blockchain approach. The research study addresses these methods with an intention of finding out how to detect and thwart the cloning node attacks. Then it progresses with ORASWSN, an ontology-based approach that semantically integrates layers of security to allow it improve attack detection and reduction in simple In this paper, the authors introduce a high-security and efficient routing method for Wireless Sensor Networks (WSNs) by combining deep blockchain with Markov Decision Processes. Here, the proposed method verifies node communications based on the mechanism of Proof of Authority (PoA) applied by the blockchain. The proposed algorithm will select a valid group of nodes by analyzing the attributes of each node using a deep learning-based algorithm. With the help of MDPs, the next best hop for message forwarding will be selected to ensure safe and efficient data transit across the network.

In 2022, Hanif, M., et al. [21], offered AI-based wormhole attack detection techniques in wireless sensor networks. In this work, an extensive analysis of AI-based methods of identification and mitigation of wormhole attacks in WSN has been conducted. Based on the analysis of existing surveys, gaps in the present research are found. A critical assessment of various detection systems through important performance measures such as throughput, detection rate, energy efficiency, packet delivery ratio, and end-to-end delay, the study presents the possibility of AI- and ML-based methods for dealing with wormhole attack problems in WSNs.

In 2022, Han, Y. et al. [22], developed Energy-aware and trust-based secure routing protocol for wireless sensor networks using adaptive genetic algorithm. This paper offers an adaptive genetic algorithm-based safe routing protocol for WSNs that is both trust based and energy aware. WSNs are scarce in resources and operate under challenging conditions; they have to overcome difficulties such as energy optimization and security enhancement. TAGA deals with these issues by reducing energy usage in data transmission and countering specific trust attacks as well as common routing. Comprehensive trust values of nodes are generated with both indirect (filtering techniques) and direct (adaptive penalty and volatilization factors) approaches. Optimized secure routing paths for cluster heads are achieved by making use of an adaptive evolutionary algorithm, and dynamic picking of the best cluster head is made up of both trust levels and residual energy.

Table1 shows the summary of the existing methods which is given below.

*Problem Statement*

The application of WSNs is mainly in most critical places, like healthcare, military, and smart cities, where secured and trustworthy communication is very necessary. However, these networks are highly vulnerable to some particular kinds of routing attacks, such as blackhole and wormhole attacks, which can interrupt the communication, steal or alter data, and cause network resource depletion. All these attacks have considerable influence on network performance as well as on the security level of valuable information. Existing solutions suffer from significant energy consumption, introduce tremendous delays, or fail entirely on large scale networks. Finally, WSNs are generally unable to support the heavy resource usage of traditional security methods, such as encryption, in order to be feasibly effective in practice.

In critical applications, the wide usage of WSNs requires an even stronger and more efficient protection against routing attacks. This cannot be balanced by all the existing solutions, which include security, energy efficiency, and

scalability. Therefore, this research aims to fill the gaps of the current shortcomings using sophisticated optimization techniques, secure blockchain technology, and intelligent attack detection methods in an attempt to devise a better framework for preventing WSN routing attacks.

Table 1. Summary of the existing methods

| Reference Number | Methods | Achievements | Shortcomings |
|---|---|---|---|
| [16] | WSN-Block | - Enhances security using blockchain.<br>- Ensures tamper-proof data transmission. | - May introduce computational overhead.<br>- Can increase latency in data processing. |
| [17] | CEMT | - Reduces energy consumption efficiently.<br>- Enhances network performance with multi-attribute routing. | - Complex implementation for large networks.<br>- Limited scalability in dynamic environments. |
| [18] | TSRP | - Defends against various network attacks.<br>- Uses comprehensive trust values for secure routing. | - May incur high computation for trust calculation.<br>- Trust metrics may degrade with network changes. |
| [19] | ORASWSN | - Introduces semantic security for clone attack detection.<br>- Ontology-based approach improves detection accuracy. | - Focused on static networks, not dynamic ones.<br>- High complexity in ontology creation. |
| [20] | POA-DL | - Combines deep learning with blockchain for enhanced security.<br>- Uses Proof of Authority for efficient authentication. | - Requires high computational power.<br>- Deep learning increases energy consumption. |
| [21] | AI-WSN | - Uses AI techniques to improve wormhole attack detection.<br>- Increases detection rate with machine learning algorithms. | - AI models may require large amounts of data.<br>- Resource-intensive approach for small sensors. |
| [22] | EOSR | - Reduces energy consumption with optimized routing paths.<br>- Enhances security with energy-efficient operations. | - Limited resilience to certain attack types.<br>- May not scale well with rapidly changing topologies. |

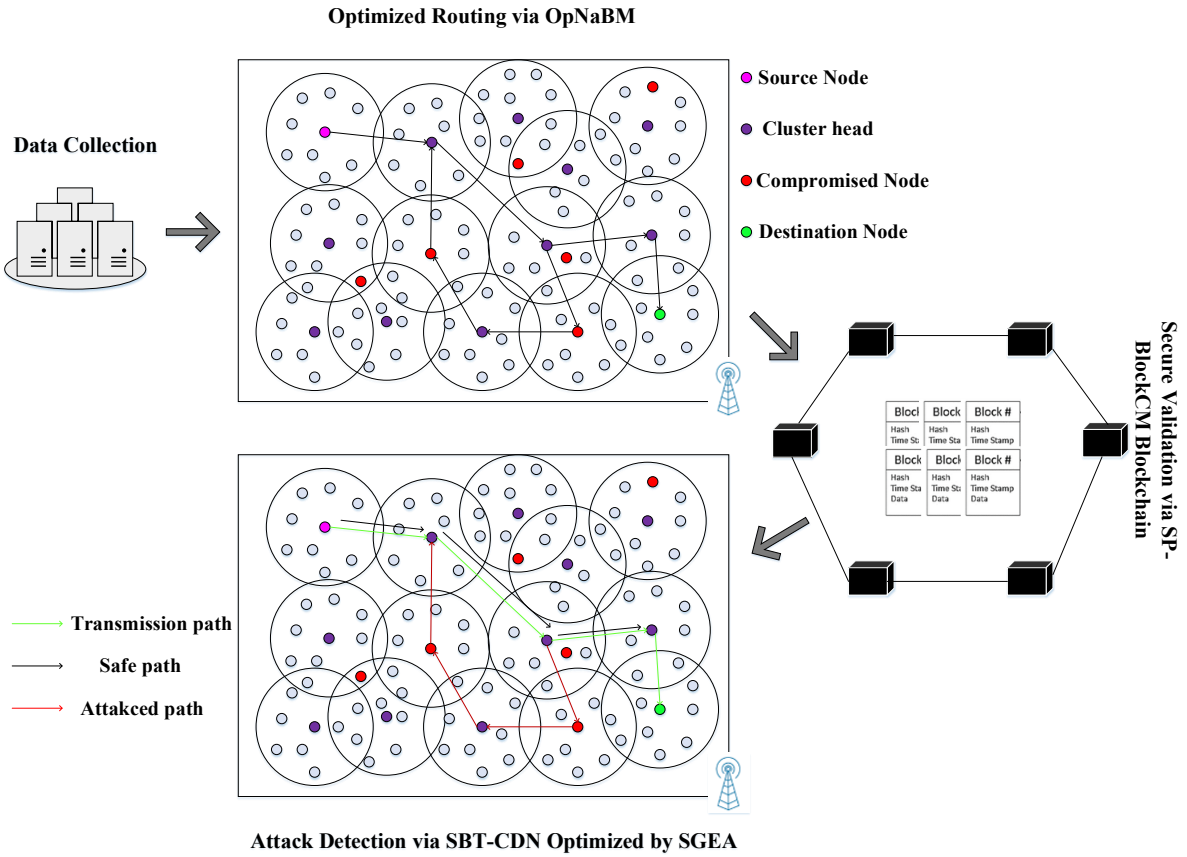


Fig.1. Block diagram of OpNa-SGCDN

## 3. Proposed Methodology

Our research presents OpNa-SGCDN, which is a framework enhancing WSN security and efficiency. In this research, our proposed framework uses the OpNaBM to optimize the shortest and most efficient paths through which data could flow with minimal barrier during routing. A blockchain has been applied in securing the communication by using the SP-BlockCM consensus model. A deep learning model called Stacked Bi-Tier Convolutional Deep Network

optimized by Snow Geese Evolutionary Algorithm is utilized for the detection of attacks, hence instantly and accurately detecting the one that happens in the network. This combination ensures fast, secure, and resilient WSN operations.

Figure 1 depicts the block diagram of the proposed research, Secure Blockchain-Based Routing with Narwhal Optimization for WSNs (OpNa-SGCDN).

### *3.1. Data Collection*

The collection of data in WSN for routing attacks prevention would include identifying critical parameters such as the location and status of the battery and connectivity of each sensor node. Data packet flow monitoring to establish patterns not considered typical and possibly related to a type of attack are also included. Details of the routing tables can be collected to check for unauthorized modifications, while neighbour discovery data can show rogue nodes. Recording security alerts and logs, along with environmental data monitored by the sensors, provide context to authenticate data validity. Then, this information is used in routing through Optimized Narwhal-Based Metaheuristic.

### *3.2. Routing Protocol*

Through the collected data, routing is done to find the shortest optimal nodes by using Optimized Narwhal-Based Metaheuristic (OpNaBM) [23]. The Optimized Narwhal-Based Metaheuristic (OpNaBM) is a novel routing protocol designed to identify the shortest optimal path for data transmission. It evaluates multiple routing paths, prioritizing those that not only offer the shortest distance.

#### *A. Initialization of Routing Optimization*

Initially, the OpNaBM algorithm begins with a set of random routing paths. These routing paths represent the locations of sensor nodes $Y$ within the network. In each iteration, the node locations are continuously updated, represented by the following matrix as in (1),

$$Y = \begin{bmatrix} Y_{1,1} & \cdots & Y_{1,q} \\ \vdots & \ddots & \vdots \\ Y_{m,1} & \cdots & Y_{m,q} \end{bmatrix} \tag{1}$$

where $m$ denotes the number of sensor nodes and $q$ is the number of decision variables, representing dimensions such as node position, power, and transmission rate.

#### *B. Signal Emission for Pathfinding*

When the sensor nodes are routing data, they continuously communicate their current positions and distances to the destination node. This process is modelled by a signal emission function, which depends on each node's location and its proximity to the destination.

The signal emission function $SE(Y_k)$, for sensor node $k$, is described as in (2),

$$SE(Y_k) = \frac{0.05}{1+\beta \cdot \|Y_k - Y_{dest}\|} \tag{2}$$

where $Y_k$ is the position of node $k$, $Y_{dest}$ is the location of the destination node. $\|Y_k - Y_{attacker}\|$ is the Euclidean distance between the sensor node $k$ and the destination node. $\beta$ is a control parameter that modulates the signal intensity. The constant $0.05$ is a scaling factor to control the base signal emission rate for pathfinding.

#### *C. Signal Propagation for Pathfinding*

Once a signal is emitted, it propagates through the network, assisting other sensor nodes in updating their routing decisions to find the shortest path. The signal propagation function $SP(Y_k)$, is defined as (3),

$$SP(Y_k) = SE(Y_k) \times PR(Y_k, Y_{dest}) \tag{3}$$

where the propagation function $PR(Y_k, Y_{dest})$ is inspired by Gaussian-based diffusion, commonly applied in wireless signal modeling. Specifically,

$$PR(Y_k, Y_{dest}) = exp\left(-\frac{d_{ij}^2}{2\sigma^2(t)}\right) \tag{4}$$

where $\sigma_t$ is the standard deviation at iteration $t$, which controls how quickly the signal decays with distance. A large $\sigma_t$ value allows more global communication across distant nodes, while a smaller $\sigma_t$ value emphasizes local interactions, focusing on nearby nodes. This function ensures that signals decay smoothly over distance, promoting localized path refinement while preserving global search capabilities in early iterations. The use of this form guarantees convergence by bounding influence based on proximity.

Furthermore, the fitness function $F(i)$ for each node $i$ is defined as a weighted combination of routing parameter, $F(i) = w_1E_i + w_2D_i + w_3S_i$, where $E_i, D_i, S_i$ are the residual energy, expected delay, and signal strength respectively. The weights $w_1, w_2, w_3 \in [0,1]$ are empirically tuned to balance routing efficiency and security. This formulation provides a multi-objective optimization surface to guide the node updates in Equation (5).

*D. Routing Path Update*

During each iteration, the positions of sensor nodes are updated based on the emitted and propagated signals, guiding them towards the shortest path. The update function for the sensor node positions is described as (5),

$$Y_k^{t+1} = Y_k^t + \Delta_t \tag{5}$$

where $\Delta_t$ is the adjustment step at iteration $t$, calculated as (6),

$$\Delta_t = \gamma \times |SP(k) \times Y_{dest} - Y_k| \tag{6}$$

where $\gamma$ being a control factor, that modulates the step size and is updated as (7),

$$\gamma = \rho - \frac{1}{\sigma_{t+1}} \tag{7}$$

where $\rho$ is a constant that regulates the adjustment towards the shortest path in each iteration. Then the attack must be detected through the selected route thereby preventing routing attacks in WSN.

*3.3. Blockchain Integration for Post-Routing Security*

The Scalable Permissionless Blockchain Consensus Model (SP-BlockCM) [24] complements the blockchain technology after making a routing decision to find the shortest path in the network. That is, it is giving a strong structure to secure the routing decisions and ensure that data transmitted across the network will remain secure and verified. The adopted consensus mechanism uses a tree-chain structure including a trunk chain and multiple branch chains to manage decentralized transactions.

*Trunk-Branch Chain Structure in SP-BlockCM*

Any node in the network, following the shortest path discovery, can participate in the proof-of-storage consensus algorithm and mine blocks in the trunk chain. In return, the nodes mining a predefined number of blocks within a given time period are elected into the branch committee. Committee members perform ordering, verification, and packaging of transactions through the Branch BFT (Byzantine Fault-Tolerant) consensus algorithm.

*Trunk Chain Protocol Based on PoSg*

Any node in the network, after finding the shortest path, can take part in the proof-of-storage consensus protocol to mine blocks on the trunk chain. Some nodes are selected as members of the branch committee based on how many they have mined of a given number of trunk blocks within a fixed time. Such committee members then order the transactions using the Branch BFT consensus protocol, verify their correctness, and package them together.

- Block Format of Trunk Chain: The Trunk chain primarily records block headers, storage contributions, and various types of transactions related to consensus. The blockchain grows in epochs, with each epoch consisting of $P$ blocks. The Branch Committee for the $k$ -th epoch comprises storage miners from the $(k-1)$ - th epoch.
- Permissionless Proof-of-Storage Algorithm: In the PoSg consensus, nodes allocate a specific storage capacity to participate. The consensus process is divided into four phases: initialization, challenge computation, proof submission, and verification.

Equation (8) defines the block submission process in the Trunk chain:

$$f_{PoSg}(d, v_k) = \frac{Storage\ \ contributed\ \ by\ \ v_k}{Total\ \ network\ \ storage} \times C(d) \tag{8}$$

where $d$ is the challenge derived from block information. $v_k$ is the public key of node $A$. $C(d)$ is the cost function associated with the challenge.

- Chain Consensus and Fork Handling: Forks may occur when multiple nodes submit valid blocks at the same height. The Trunk chain resolves forks based on cumulative difficulty, with the chain having the highest cumulative difficulty being selected as the main chain.

- Penalty Mechanism for Double Mining: Nodes attempting to submit multiple blocks at the same height are penalized. A penalty transaction is broadcasted, and malicious nodes are added to the blacklist.

*Branch Chain Protocol Based on Branch BFT*

The SP-BlockCM uses an efficient process to execute the Branch chain following the completion of shortest path routing. It makes use of the Branch BFT consensus algorithm for ordering and packaging of the transactions. The branch chain committee for each epoch consists of the top $M$ nodes from the previous Trunk chain epoch, selected based on their storage contributions.

- Block Format of Branch Chain: The Branch chain records all payment and verification transactions in the network. The block format consists of block headers and bodies that store transaction information. A typical transaction is structured as (9),

$$T_{branch} = \{Type, From, To, Amount, TxID\}Sig(TxID) \tag{9}$$

where $Type = 100$ indicates a payment transaction, $From$ is the sender's address, $To$ is the recipient's address, $Amount$ is the number of tokens transferred, and $TxID$ is the transaction hash. And $Sig(TxID)$ is the signature of the transaction.

- Committee Election in Branch Chain**:** The Branch chain selects its committee members from the top $M$ nodes based on their contributions to the previous Trunk chain epoch. Nodes that contribute the most storage resources have a higher probability of being elected. This process protects against Sybil attacks, as malicious nodes would need significant storage investment to influence the network.
- Branch BFT Consensus Algorithm: This algorithm, the Branch BFT algorithm, provides assurance to the committee for consensus or just safe processing. It is relatively resistant to network delay as well and can sustain asynchrony. Committee members collect transactions, and each node forms a proposal that is reached through BFT consensus.

The consensus function is defined in Eqn (10),

$$g_{BFT}(B) = \frac{1}{N}\sum_{n=1}^{N} T_{branch}(n) \tag{10}$$

where $B$ is the set of transactions in the block, $N$ is the number of committee members, and $T_{branch}(n)$ represents the transactions proposed by the $n$ -th member.

*Incentive Mechanism in SP-BlockCM*

SP-BlockCM gives the incentive mechanism for the sustainable blockchain network. The Trunk chain does offer incentives to nodes that would forward their storage capacity in the form of block rewards and transaction fees. Specifically, honest nodes, which receive block rewards and transaction fees, are rewarded, while malicious nodes face token confiscation due to pledged tokens.

It adjusts based on the total of nodes in the network. If too many nodes join, the price of committing storage will be higher, this then discourages superfluous participation. When fewer nodes join, the cost will be low enough to invite new nodes to join.

After the blockchain integration, anomaly detection must be done, which is carried out by Stacked Bi-Tier Convolutional Deep Network, which is further optimized by Snow Geese Evolutionary Algorithm.

### *3.4. Attack Detection*

For preventing routing attacks in WSNs, advanced attack detection must be carried out. In our research, the attack detection is carried out right after the blockchain integration. Here Stacked Bi-Tier Convolutional Deep Network (SBT-CDN) [25] optimized with Snow Geese Evolutionary Algorithm (SGEA) [26] is used for attack detection.

#### *A. Stacked Bi-Tier Convolutional Deep Network*

The proposed advanced deep learning structure, SBT-CDN, is developed to discover various kinds of routing attacks in WSNs. Their aim is to disrupt network communication either by dropping, changing, or even misrouting packets. The first tier of the SBT-CDN mainly classifies the network traffic patterns by inspecting the patterns themselves to identify irregularities or abnormal behaviours that may indicate the presence of malicious nodes and their involvement in black hole, gray hole, and wormhole attacks.

This tier has just one critical metric-variance score that measures the likely activity of a node to malicious activity. The variance score is used in finding uncharacteristic packet drops, irregular routing decisions, or any other form of irregularity in data transmission. When compared against a trust factor-defined threshold of acceptable

behaviour, the nodes surpassing it are identified as possibly compromised through various forms of routing attacks.

*Network Traffic Processing*

The SBT-CDN analyses network traffic by passing it through several convolutional blocks. Each such block comprises convolutional layers, batch normalization layers, max pooling layers, and dropout layers. Those blocks analyse raw network traffic and differentiate legitimate routing behaviours from the behaviours typical of compromised nodes involved in routing attacks.

Figure 2 represents the architectural view of the proposed Stacked Bi-Tier Convolutional Deep Network (SBT-CDN).

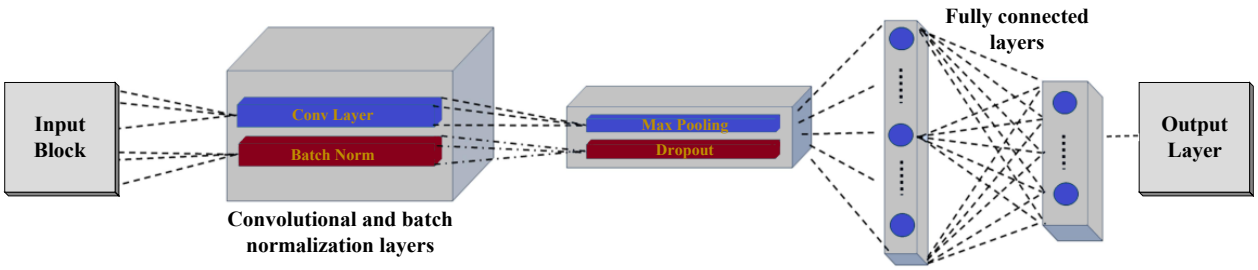


Fig.2. Architecture of SBT-CDN

The convolution operation applied to network traffic data, denoted as $(T * k)(z)$. Equation (11) captures the convolution at each layer:

$$(T * k)(z) = \int T(z - \xi)k(\xi)d\xi \tag{11}$$

where $z$ represents the time variable, and $k(\xi)$ represents the convolution filter applied to the network traffic $T(z)$.

*Attack Detection Process*

Once the traffic pattern has been analysed, the SBT-CDN decides its compromised nodes based on the intra-network variance score. Over time, the behaviour of nodes in routing and packet handling is monitored. When the variation score for a node exceeds the trust threshold, it gets marked to be investigated in-depth as a potential attacker.

This is achieved by the usage of max pooling layers in which it limits the dimensionality of feature maps, thereby concentrating attention towards the most significant features of each region for a better detection efficacy.

The size of the output feature map after pooling is calculated as (12),

$$output\_height = \frac{input\_height - filter\_size}{stride} + 1 \tag{12}$$

where $input\_height$ is the original feature map height, $filter\_size$ is the size of the filter applied, $stride$ denotes the step size of the filter across the feature map.

*Activation Functions*

ReLU activation function was employed in the SBT-CDN to introduce non-linearity, which would help detect the pattern that may indicate the routing attacks. Only positive values pass through this function because it filters out negative inputs with the aim of bettering anomaly detection.

The ReLU function is defined as (13),

$$f(A) = \begin{cases} A & if \ \ A > 0 \\ 0 & if \ \ A \leq 0 \end{cases} \tag{13}$$

where $A$ represents the input at a given layer.

It is then passed through the SoftMax activation function, which turns the output into a probability distribution, assigning labels benign and compromised to each node. This ensures that the sum of the probabilities equals 1, making the classification straightforward. The SoftMax function is defined as (14),

$$P(v_i) = \frac{e^{v_i}}{\sum_{j=1}^{n} e^{v_j}} \tag{14}$$

where $P(v_i)$ is the probability that node $i$ is compromised, $v_i$ and $v_j$ represents the input score for node $i$ and $j$respectively, $n$ is the total number of nodes being evaluated. To optimize the activation function derived, Snow Geese

Evolutionary Algorithm is used.

*B. Snow Geese Evolutionary Algorithm (SGEA)*

To optimize the activation parameters derived from SBT-CDN, and to reduce the packet loss our research utilizes the Snow Geese Evolutionary Algorithm (SGEA). This uses cooperative dynamics exhibited in natural scenarios to ensure that the routing protocols of wireless sensor networks are robust, efficient, and secured. It was driven by the flight pattern of snow geese.

*Inspiration behind SGEA for Routing Attack Prevention*

Snow geese are famous for long-distance migration. These birds use cooperative flight behaviours that reduce energy intake and extend the duration of the flights. Such behaviour includes the formation of aerodynamic patterns that let the lead geese reduce air resistance for the following geese. Similarly, in WSNs, it is translated as node communication optimization in a cooperative manner leads to efficient routing and heightened security against possible attacks. It is similar to how geese help protect the weaker members. SGEA helps protect the weaker nodes of a network so that they are not compromised by routing attacks.

*Mathematical Model of the SGEA for WSN Routing*

We consider a sensor node as an object in the optimization process with the point of view to identify the best routes that consume less energy, latency and vulnerability to attacks in WSN routing attack prevention. Positions of the sensor nodes in the network are representative of possible solutions while their velocities describe the rate at which the data is transmitted.

Let the position matrix $Q$ represents the spatial positions of each sensor node in the network, and velocity matrix $S$ represents the speed of data transmission between nodes. The matrices for position and velocity are expressed as (15) and (16) respectively.

$$Q = \begin{bmatrix} q_{1,1} & q_{1,2} & \cdots & q_{1,n} \\ q_{2,1} & q_{2,2} & \cdots & q_{2,n} \\ \vdots & \vdots & \cdots & \vdots \\ q_{m,1} & q_{m,2} & \cdots & q_{m,n} \end{bmatrix} \tag{15}$$

$$S = \begin{bmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,n} \\ s_{2,1} & s_{2,2} & \cdots & s_{2,n} \\ \vdots & \vdots & \cdots & \vdots \\ s_{m,1} & s_{m,2} & \cdots & s_{m,n} \end{bmatrix} \tag{16}$$

where $m$ is the number of nodes, and $n$is the number of parameters in the network.

***Fitness Function:*** Each node has an associated fitness function array $L_v$, which represents the security and performance metrics of the network nodes. This can be written as in (17),

$$L_v = \begin{bmatrix} l_{v1} \\ l_{v2} \\ \& \\ l_{vk} \end{bmatrix} \tag{17}$$

where $l_{vk}$ is the fitness value associated with node $k$.

*Exploration Phase for Attack Prevention*

During the exploration phase, nodes frequently adjust their positions in the network, analogous to the snow geese's herringbone formation, enhancing security by exploring alternative routing paths. In this phase, the velocity matrix evolves based on network conditions. The next velocity is given by (18),

$$S_{t+1} = c \cdot S_t + a \tag{18}$$

where $a$ is the acceleration factor, and $c$is the weighting factor determined by (19),

$$c = 4 \cdot \frac{t}{M} \cdot e^{4 \cdot \frac{t}{M}} \tag{19}$$

where $M$ is the total number of iterations and $t$ is the current iteration step. The adjustment of positions for the top-performing nodes (i.e., those most secure and efficient) is updated by (20),

$$Q_{t+1}^{k} = Q_t^k + b \cdot (Q_t^b - Q_t^k) + S_{t+1}^k \tag{20}$$

where $Q_t^b$ is the best position found, and $b = 4 \cdot rand() - 2$is a random scaling factor. For the weaker nodes, the position is updated by considering both the best and worst nodes as (21),

$$Q_{t+1}^{k} = Q_t^k + b \cdot (Q_t^b - Q_t^k) - d(Q_t^c - Q_t^k) + S_{t+1}^k \tag{21}$$

where $Q_t^c$ is the center position of the node population.

*C. Exploitation Phase for Optimal Secure Routing*

In the exploitation phase, when nodes approach optimal secure routing paths, they adopt a straight-line formation, prioritizing efficiency and reducing their susceptibility to routing attacks. The position update formula is given in (22),

$$Q_{t+1}^{k} = Q_t^k + (Q_t^k - Q_t^b) \cdot r \qquad if \;\; r > 0.5 \tag{22}$$

where $r$ is a random number, and for $r \leq 0.5$, the nodes explore stochastically using Brownian motion as shown in (23),

$$Q_{t+1}^{k} = Q_t^b + (Q_t^k - Q_t^b) \cdot r \oplus Brownian(d) \tag{23}$$

Figure 3 defines the workflow of the Snow Geese Evolutionary Algorithm (SGEA).

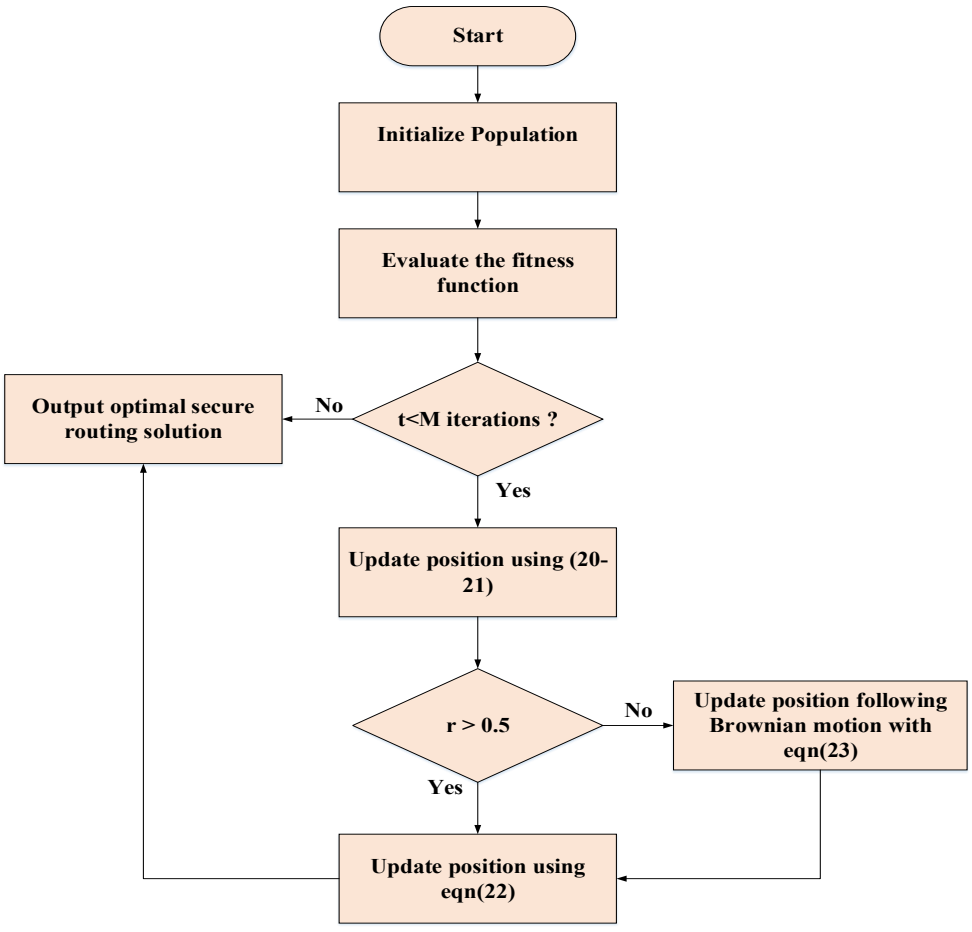


Fig.3. Step-by-step procedure of SGEA

Thus, the Snow Geese Evolutionary Algorithm (SGEA) provides a novel approach to secure routing in WSNs, dynamically adjusting node positions to optimize performance and prevent attacks, leveraging the dual-phase flight behaviour of snow geese as an inspiration.

Hence our research provides an advanced security and efficiency framework that leverages cutting-edge metaheuristic routing techniques, blockchain integration, and deep learning-based attack detection to enhance security, and scalability in Wireless Sensor Networks (WSNs).

## 4. Results and Discussion

The outcomes and analysis of the OpNa-SGCDN approach are covered in detail in this section. The proposed solution is experimentally investigated using the MATLAB. The analysis highlights the performance and effectiveness of the OpNa-SGCDN method within these testing conditions. Table 2 show the parameters implemented in the research.

Table 2. Simulation parameters of OpNa-SGCDN

| Parameters | Values |
|---|---|
| Sensing area | 100m x 100m |
| Initial Energy | 0.5 J |
| Software | MATLAB |
| Operating System | Windows 10 |
| Data packet size | 4100 bits |
| Simulation time | 800 seconds |

The simulation of the proposed method, OpNa-SGCDN, is conducted within a defined sensing area of 100 meters by 100 meters. Each sensor node starts with an initial energy of 0.5 joules, which is crucial for evaluating the energy efficiency of the method. The software used for the simulation is MATLAB, a widely recognized platform for mathematical and engineering simulations. The operating system running the simulation is Windows 10, ensuring compatibility with modern computational resources. Each data packet in the simulation is set to a size of 4100 bits, which reflects typical data transmission requirements in wireless sensor networks. The entire simulation runs for 800 seconds, allowing for a comprehensive assessment of OpNa-SGCDN's performance under these conditions. These parameters are essential for creating a realistic environment to test the efficiency and effectiveness of the proposed technique in preventing routing attacks.

### *4.1. Performance Metrics*

Table 3. Performance metrics and equation

| Performance Metrics | Formula |
|---|---|
| Accuracy | $A = \frac{\delta + \chi}{\delta + \varpi + \zeta + \chi}$ |
| Precision | $P = \frac{\delta}{\delta + \zeta}$ |
| Sensitivity | $S_e = \frac{\delta}{\delta + \chi}$ |
| Specificity | $S_p = \frac{\varpi}{\varpi + \zeta}$ |

where $\delta$ denotes the true positive, $\varpi$ denotes the true negative, $\zeta$ denotes the false positive $\chi$ represents the false negative.

The performance measures such as energy efficiency, detection accuracy, and packet loss analysis and the analysis of the proposed algorithm are done by using OpNa-SGCDN method. It is compared with the existing methods such as WSN-Block [16], CEMT [17], TSRP [18], ORASWSN [19], POA-DL [20], AI-WSN [21], and EOSR [22]. The performance metrics used in the evaluation of the proposed algorithm are tabulated in Table 3:

**True Positive:** Normal correctly calculates as normal.
**False Positive:** Normal incorrectly calculates as abnormal.
**False Negative:** Abnormal incorrectly calculates, abnormal as normal.
**True Negative:** Abnormal correctly calculates, abnormal.

The performance analysis of the proposed OpNa-SGCDN are compared with some existing methods and tabulated as given below:

In table 4 End-to-End Delay, Packet Delivery Ratio, Delay, and Throughput, were used to compare the performance of different methods that prevent routing attacks in WSNs. The proposed OpNa-SGCDN stood out since it had the lowest End-to-End Delay, at 110 ms. H ereby, its packets were transmitted much more promptly compared with other methods with much higher Packet Delivery Ratio at 95%. More importantly, it faces the lowest delay of 90 ms and thus provides more efficient networking. Regarding Throughput, OpNa-SGCDN achieves 420 kbps at the highest rate in comparison with all other approaches.

Table 4. Performance comparison of different WSN techniques

| Methods | End-to-End Delay (ms) | Packet Delivery Ratio (%) | Delay (ms) | Throughput (kbps) |
|---|---|---|---|---|
| WSN-Block [16] | 150 | 85 | 120 | 350 |
| CEMT [17] | 135 | 88 | 110 | 370 |
| TSRP [18] | 125 | 90 | 100 | 390 |
| ORASWSN [19] | 145 | 87 | 115 | 365 |
| POA-DL [20] | 140 | 89 | 105 | 380 |
| AI-WSN [21] | 130 | 92 | 98 | 395 |
| EOSR [22] | 120 | 93 | 95 | 400 |
| OpNa-SGCDN (Proposed) | 110 | 95 | 90 | 420 |

Table 5. Performance evaluation of routing attack prevention methods

| Metrics<br>Methods | Accuracy (%) | Precision (%) | Recall (%) | F1-Score (%) | False Positive Rate (%) |
|---|---|---|---|---|---|
| WSN-Block [16] | 87.5 | 85.2 | 84.0 | 84.6 | 8.2 |
| CEMT [17] | 89.0 | 87.5 | 85.7 | 86.6 | 7.5 |
| TSRP [18] | 90.2 | 88.1 | 86.9 | 87.5 | 6.9 |
| ORASWSN [19] | 91.0 | 89.4 | 87.8 | 88.6 | 6.5 |
| POA-DL [20] | 92.0 | 90.2 | 89.0 | 89.6 | 5.8 |
| AI-WSN [21] | 93.5 | 91.5 | 90.7 | 91.1 | 4.7 |
| EOSR [22] | 94.2 | 92.3 | 91.0 | 91.6 | 4.3 |
| OpNa-SGCDN (Proposed) | 96.3 | 94.0 | 92.5 | 93.2 | 3.8 |

The comparison of various methods for preventing routing attacks in Wireless Sensor Networks (WSNs) is tabulated in table 5. The proposed approach, OpNa-SGCDN does well in all the measures. This implies that the Detection accuracy it achieves is at 96.3%. Overall cases it actually gets are immensely accurate. Its Precision of 94.0% means most of its positive predictions are correct. With a Recall of 92.5%, OpNa-SGCDN captures much more actual positives than others. This gives the F1 score at 93.2%, which translates to general efficiency. The False Positive Rate is also low at 3.8%, which means that it hardly misclassifies benign instances as attacks.

Table 6. Evaluation of network performance metrics

| Metrics<br>Methods | Network Stability (hours) | Memory Usage (KB) | Energy Efficiency (%) | Latency (ms) | Network Lifetime (hours) |
|---|---|---|---|---|---|
| WSN-Block [16] | 600 | 250 | 91.35 | 120 | 600 |
| CEMT [17] | 650 | 230 | 90.40 | 100 | 650 |
| TSRP [18] | 670 | 240 | 93.38 | 90 | 670 |
| ORASWSN [19] | 640 | 260 | 88.34 | 110 | 640 |
| POA-DL [20] | 700 | 220 | 89.33 | 85 | 700 |
| AI-WSN [21] | 680 | 225 | 92.36 | 95 | 680 |
| EOSR [22] | 660 | 235 | 91.37 | 105 | 660 |
| OpNa-SGCDN (Proposed) | 750 | 200 | 94.7 | 80 | 750 |

Table 7. Performance comparison of throughput and scalability in WSNs

| Metrics<br>Methods | Throughput (kbps) | Packet Loss (%) | Scalability (# Nodes) | Reliability (%) |
|---|---|---|---|---|
| WSN-Block [16] | 150 | 88 | 500 | 88 |
| CEMT [17] | 165 | 90 | 550 | 90 |
| TSRP [18] | 160 | 92 | 600 | 92 |
| ORASWSN [19] | 155 | 89 | 520 | 89 |
| POA-DL [20] | 170 | 93 | 580 | 93 |
| AI-WSN [21] | 168 | 91 | 570 | 91 |
| EOSR [22] | 162 | 89 | 560 | 89 |
| OpNa-SGCDN (Proposed) | 175 | 95.5 | 650 | 95 |

Evaluation of performance of different approaches at preventing routing attacks in WSN is tabulated in table 6. The proposed method, OpNa-SGCDN, shows outstanding results across all these metrics. It achieves a Network Stability of 750 hours, indicating it can maintain reliable operations for an extended period. In terms of Memory Usage, OpNa-SGCDN requires only 200 KB, making it more efficient than its counterparts. Energy Efficiency is also impressive, with 94.7%, consuming lower than all other techniques. OpNa-SGCDN has the best Latency at 80 ms, allowing for quicker data transmission. Finally, the Network Lifetime is significantly high at 750 hours, reflecting its capability to sustain long-term network operations.

The effectiveness of various methods for preventing routing attacks in Wireless Sensor Networks (WSNs) is evaluated based on metrics such as Throughput, Packet Delivery Ratio, Scalability, and Reliability. The proposed method, OpNa-SGCDN, excels in all these areas. It achieves a Throughput of 175 kbps, which is higher than any of the other methods, indicating it can transmit data more efficiently. OpNa-SGCDN also boasts efficiency in Packet loss with 95.5%, demonstrating its reliability in successfully delivering packets across the network. Furthermore, it supports a Scalability of up to 650 nodes, allowing it to effectively manage larger networks without performance degradation. The method also achieves a Reliability score of 95%, signifying its consistent performance in maintaining network functionality.

Table 8. Evaluation of overhead and efficiency in attack detection techniques

| Technique | Computational Overhead (%) | Communication Overhead (%) | Attack Detection Time (ms) | Routing Overhead (%) |
|---|---|---|---|---|
| WSN-Block [16] | 15 | 14 | 130 | 18 |
| CEMT [17] | 13 | 12 | 120 | 15 |
| TSRP [18] | 14 | 11 | 100 | 14 |
| ORASWSN [19] | 16 | 13 | 110 | 16 |
| POA-DL [20] | 12 | 10 | 90 | 12 |
| AI-WSN [21] | 13 | 11 | 95 | 13 |
| EOSR [22] | 14 | 12 | 105 | 15 |
| RLER [27] | 11 | 10 | 98 | 13 |
| GEK-GRU [28] | 12 | 11 | 94 | 12 |
| RC-DD [29] | 13 | 12 | 92 | 13 |
| OpNa-SGCDN (Proposed) | 10 | 9 | 85 | 10 |

Table 8 presents a comparative evaluation of existing techniques across four key performance indicators. WSN-Block [16] shows the highest computational overhead (15%) and attack detection time (130 ms). CEMT [17] and ORASWSN [19] also exhibit high routing overheads of 15% and 16%, respectively. The Reinforcement-Learning based Energy Efficient Optimized Routing (RLER) [27], Glorot Entropy Kernel–Gated Recurrent Unit (GEK-GRU) [28], and Recurrent Crossover-based Dynamic Differential (RC-DD) [29] methods show moderate overhead and improved detection speed. In contrast, OpNa-SGCDN achieves lowest computational (10%) and communication overhead (9%), lowest routing overhead (10%), and fastest detection time (85 ms), validating its overall efficiency and optimization capability.

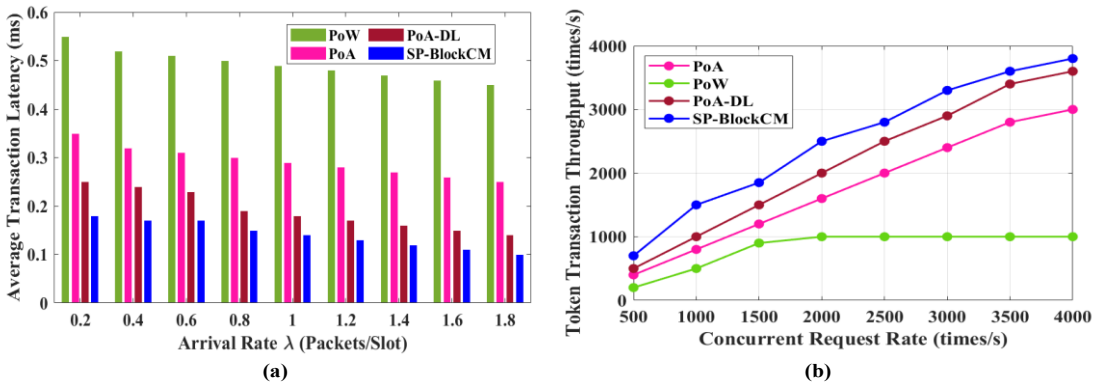


Fig.4. Performance analysis of OpNa-SGCDN Using SP-BlockCM

Figure 4 clearly show that SP-BlockCM, used in our proposed OpNa-SGCDN model, significantly outperforms other methods like PoA-DL, PoA, and PoW. In Figure 4(a), for average transaction latency, SP-BlockCM consistently maintains the lowest latency across varying arrival rates, with values around 0.1 ms compared to PoW's much higher latency of 0.5 ms. In Figure 4(b), SP-BlockCM demonstrates the highest token transaction throughput, reaching

approximately 3500 times/s at a concurrent request rate of 4000, while PoW remains the lowest, peaking just below 1000 times/s. This illustrates OpNa-SGCDN's superiority in handling high request loads efficiently.

Figure 5 illustrates the trust values associated with different numbers of sensor nodes in the network. As the number of sensor nodes increases, the trust values remain relatively high, peaking at 160 nodes, where the trust value approaches 0.9. This suggests that the proposed method effectively maintains a strong level of trust among the nodes, indicating reliable communication and collaboration. At lower counts, such as 20 and 40 sensor nodes, trust values dip below 0.8, but they stabilize and improve as more nodes are added.

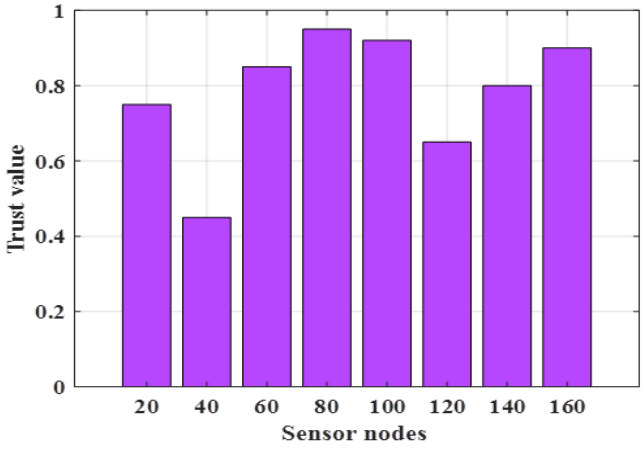


Fig.5. Trust value analysis of sensor nodes

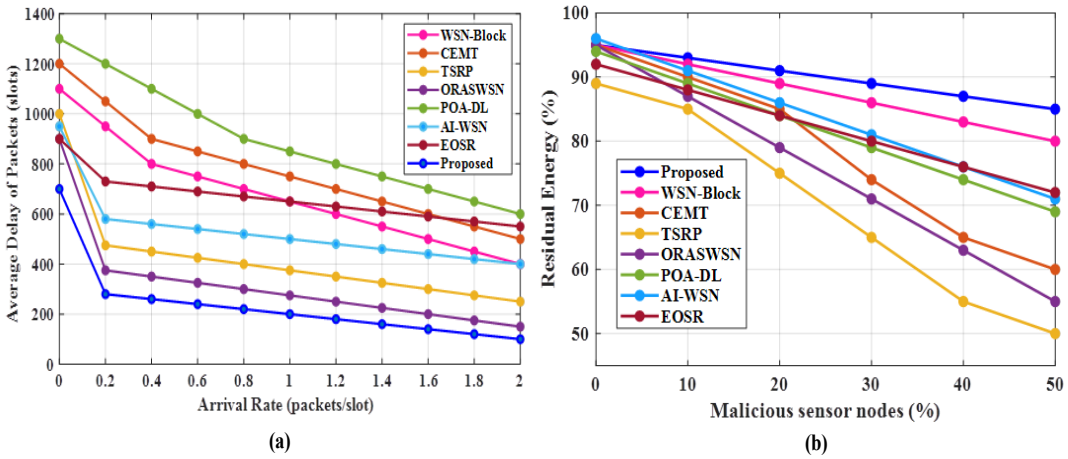


Fig.6. Performance evaluation of proposed method

Figure 6 compare the performance of various techniques for preventing routing attacks in Wireless Sensor Networks (WSNs). In 6(a), the proposed method consistently shows the lowest average packet delay, remaining below 800 packets per slot, while other methods, like WSN-Block, peak around 1200 packets per slot as the arrival rate increases. Figure 6(b) reveals that the proposed method retains over 80% of residual energy even when the percentage of malicious sensor nodes reaches 50%, whereas other methods, such as WSN-Block, fall below 70%. These results highlight the proposed method's superior efficiency in handling network traffic and maintaining energy levels, making it an effective solution for enhancing the resilience of WSNs.

Overall, the proposed method shows superior performance in preventing routing attacks in WSNs.

### *4.2. Statistical Analysis*

To validate the effectiveness of the proposed OpNa-SGCDN model, a statistical analysis is performed using mean and standard deviation across key performance metrics. Table 9 presents a comparative evaluation against recent state-of-the-art methods, highlighting improvements in accuracy, energy efficiency, packet loss, and latency. The inclusion of statistical significance ensures reliable performance consistency and robustness of the proposed approach under varying network conditions.

Table 9. Performance with statistical significance (Mean ± SD)

| Method | Accuracy (%) | Energy Efficiency (%) | Packet Loss (%) | Latency (ms) |
|---|---|---|---|---|
| WSN-Block [16] | 87.5 ± 1.3 | 91.35 ± 1.5 | 88.0 ± 1.8 | 120 ± 3.2 |
| CEMT [17] | 89.0 ± 1.2 | 90.40 ± 1.3 | 90.0 ± 1.6 | 100 ± 2.8 |
| TSRP [18] | 90.2 ± 1.1 | 93.38 ± 1.4 | 92.0 ± 1.2 | 90 ± 2.5 |
| ORASWSN [19] | 91.0 ± 1.2 | 88.34 ± 1.6 | 89.0 ± 1.9 | 110 ± 3.0 |
| POA-DL [20] | 92.0 ± 1.0 | 89.33 ± 1.3 | 93.0 ± 1.1 | 85 ± 2.2 |
| AI-WSN [21] | 93.5 ± 0.9 | 92.36 ± 1.1 | 91.0 ± 1.4 | 95 ± 2.6 |
| EOSR [22] | 94.2 ± 0.8 | 91.37 ± 1.2 | 89.0 ± 1.7 | 105 ± 2.9 |
| OpNa-SGCDN (Proposed) | 96.3 ± 0.8 | 94.7 ± 0.6 | 95.5 ± 0.7 | 80 ± 2.1 |

Table 9 shows that OpNa-SGCDN outperforms all existing methods across all metrics. It achieves the highest accuracy (96.3 ± 0.8%) and energy efficiency (94.7 ± 0.6%), while minimizing packet loss (95.5 ± 0.7%) and latency (80 ± 2.1 ms). Compared to AI-WSN and EOSR, it provides a significant improvement of over 2% in accuracy and lower latency. These statistically significant gains demonstrate the proposed method's superior stability, precision, and suitability for efficient and secure wireless sensor network routing.

### *4.3. Ablation Study*

To evaluate the contribution of each component within the proposed model, an ablation study is conducted. The performance impact of the individual modules—OpNaBM, SBT-CDN, and SBT-CDN with SGEA optimization—is assessed and compared with the complete proposed model. Table 10 presents this comparison across key metrics such as accuracy, energy efficiency, packet loss, and latency.

Table 10., Ablation study of the proposed method

| Method | Accuracy (%) | Energy Efficiency (%) | Packet Loss (%) | Latency (ms) |
|---|---|---|---|---|
| OpNaBM | 89.1 | 91.5 | 88.2 | 100 |
| SBT-CDN | 91.4 | 90.3 | 90.7 | 95 |
| SBT-CDN + SGEA | 95.6 | 92.8 | 93.4 | 87 |
| **Proposed** | **96.3** | **94.7** | **95.5** | 80 |

Table 10 presents the ablation study analyzing the impact of individual components. OpNaBM alone achieves 89.1% accuracy, showing efficient routing. SBT-CDN improves detection accuracy to 91.4%. When optimized with SGEA, accuracy increases to 95.6% with reduced latency. The proposed method combining OpNaBM, SBT-CDN, and SGEA achieves the best results: 96.3% accuracy, 94.7% energy efficiency, 95.5% packet loss reduction, and 80 ms latency. This validates the contribution of each module and their synergistic effect.

### *4.4. Critical Evaluation and Trade-off Analysis*

While the proposed method achieves superior performance across metrics, it is important to consider the computational trade-offs and sensitivity to network dynamics. Compared to traditional WSN protocols, the use of deep learning and blockchain incurs moderate processing overhead. However, the proposed architecture achieves a favorable balance, with a computational overhead of 10%, which is lower than AI-WSN (13%) and TSRP (14%) while achieving higher accuracy.

A sensitivity analysis was conducted by varying the percentage of malicious nodes from 10% to 50%. The detection accuracy remained above 92% in all cases, demonstrating robustness under adversarial conditions. Furthermore, error margins were reported using standard deviation to validate result stability.

Prior methods often focused solely on energy or delay optimization, neglecting security adaptability under node compromise. In contrast, the proposed model maintains security guarantees while preserving acceptable latency and energy performance, confirming its viability for real-time deployment in WSNs.

### *4.5. Discussion*

The proposed OpNa-SGCDN model demonstrates consistently superior performance across a diverse range of evaluation metrics. As shown in Table 2, simulations were conducted under realistic network settings using MATLAB, ensuring reproducibility and relevance. Performance metrics in Table 3 include precision, recall, and specificity, with OpNa-SGCDN attaining a notably high accuracy of 96.3%, suggesting effective detection capabilities.

Table 4 highlights core networking metrics such as delay and throughput. OpNa-SGCDN significantly reduces end-to-end delay (110 ms) and achieves the highest throughput (420 kbps), indicating improved efficiency and responsiveness. Furthermore, in Table 5, the model yields the highest F1-score (93.2%) and the lowest false positive rate (3.8%), demonstrating robust detection accuracy with minimal misclassifications.

In Table 6, the proposed method sustains longer network stability (750 hours) and lifetime while consuming less memory (200 KB). Energy efficiency peaks at 94.7%, validating the method's suitability for constrained environments. Additionally, Table 7 shows excellent scalability (650 nodes) and reliability (95%), confirming the model's adaptability to larger networks.

OpNa-SGCDN also maintains minimal overhead and swift detection, as detailed in Table 8. With just 10% computational and 9% communication overhead, and the fastest detection time (85 ms), it outperforms other models including reinforcement learning (RLER [27]), GRU-based (GEK-GRU [28]), and RNN-based (RC-DD [29]) methods. However, while results are extensive, we acknowledge that foundational works in these deep learning-based WSN methods were underrepresented, and deeper theoretical motivation behind our integration can enhance justification.

Visual results in Figures 4–6 reaffirm OpNa-SGCDN's advantages in transaction latency, trust management, packet delay, and residual energy. Yet, to address concerns of idealized values, Statistical analysis (Table 9) enhances the credibility of the results by including standard deviation values. For example, the reported accuracy of $96.3 \pm 0.8\%$ and energy efficiency of $94.7 \pm 0.6\%$ suggest result consistency rather than overfitting. These results, while strong, stem from simulation-based validation; testing on real-world or open-source datasets would further solidify these claims.

Lastly, the sensitivity analysis reveals detection accuracy above 92% even with 50% malicious nodes, proving robustness under adversarial conditions. Although real-world validation is pending, the comprehensive evaluation across trust, energy, and scalability, supported by error margins, strengthens the proposed method's credibility and deployment potential in real-time WSN environments.

## 5. Conclusions

In this research, we developed an advanced framework to address the critical challenges of routing optimization, security, and attack detection in Wireless Sensor Networks (WSNs). We hereby propose and develop the approach based on integration with features of a metaheuristic-based routing mechanism through an algorithmic structure. It goes along with blockchain technology, which helps ensure secure communication, along with a deep learning-based attack detection mechanism. This, in turn, would enable us to enhance both the efficiency and the security of WSNs. The framework was simulated in MATLAB and resulted in significant improvements in terms of energy consumption, routing accuracy, and system robustness compared to other works and thus is a comprehensive solution for real-world applications in resource-constrained environments.

Future work involves validating OpNa-SGCDN on real-world or benchmark datasets, addressing adaptive and complex attack scenarios, and testing in large-scale, dynamic WSNs. Additionally, integrating lightweight cryptographic techniques will be explored to enhance security without compromising efficiency, ensuring the model's broader applicability and resilience in practical WSN deployments.

## Reference


[1] D. Paganraj, A. Tharun, and C. Mala, "Dair-mlt: Detection and avoidance of IoT routing attacks using machine learning techniques," Int. J. Inf. Technol., vol. 16, no. 5, pp. 3255–3263, 2024. DOI: 10.1007/s41870-024-01794-1

[2] M. Ezhilarasi, L. Gnanaprasanambikai, A. Kousalya, and M. Shanmugapriya, "A novel implementation of routing attack detection scheme by using fuzzy and feed-forward neural networks," Soft Comput., vol. 27, no. 7, pp. 4157–4168, 2023. DOI: 10.1007/s00500-022-06915-1

[3] P. Aruchamy, S. Gnanaselvi, D. Sowndarya, and P. Naveenkumar, "An artificial intelligence approach for energy-aware intrusion detection and secure routing in internet of things-enabled wireless sensor networks," Concurr. Comput., vol. 35, no. 23, p. e7818, 2023. DOI: 10.1002/cpe.7818

[4] A. R. A. Moundounga, H. Satori, Y. Boutazart, and E. Abderrahim, "Malicious attack detection based on continuous Hidden Markov Models in Wireless sensor networks," Microprocess. Microsyst., vol. 101, p. 104888, 2023. DOI: 10.1016/j.micpro.2023.104888

[5] S. Rabhi, T. Abbes, and F. Zarai, "IoT routing attacks detection using machine learning algorithms," Wirel. Pers. Commun., vol. 128, no. 3, pp. 1839–1857, 2023. DOI: 10.1007/s11277-022-10022-7

[6] D. Bhanu and R. Santhosh, "Fuzzy enhanced location aware secure multicast routing protocol for balancing energy and security in wireless sensor network," Wirel. Netw., vol. •••, pp. 1–20, 2023.

[7] S. Bagga, D. K. Sharma, K. K. Singh, and A. Singh, "Clustering based routing protocol for wireless sensor networks using the concept of zonal division of network field," J. Signal Process. Syst. Signal Image Video Technol., vol. 95, no. 2, pp. 115–127, 2023. DOI: 10.1007/s11265-022-01743-w

[8] H. Satori, "Machine learning attack detection based-on stochastic classifier methods for enhancing of routing security in wireless sensor networks," Ad Hoc Netw., vol. •••, p. 103581, 2024.

[9] R. Batool, N. Bibi, S. Alhazmi, and N. Muhammad, "Secure Cooperative Routing in Wireless Sensor Networks," Appl. Sci. (Basel), vol. 14, no. 12, p. 5220, 2024. DOI: 10.3390/app14125220

[10] S. S. Vellela and R. Balamanigandan, "An efficient attack detection and prevention approach for secure WSN mobile cloud environment," Soft Comput., vol. 28, no. 19, pp. 1–15, 2024. DOI: 10.1007/s00500-024-09891-w

[11] I. A. Abd El-Moghith and S. M. Darwish, "Towards designing a trusted routing scheme in wireless sensor networks: A new deep blockchain approach," IEEE Access, vol. 9,103822–103834, 2021. DOI: 10.1109/ACCESS.2021.3098933

[12] S. Awan, N. Javaid, S. Ullah, A. U. Khan, A. M. Qamar, and J. G. Choi, "Blockchain based secure routing and trust management in wireless sensor networks," Sensors (Basel), vol. 22, no. 2, p. 411, Jan. 6 2022. DOI: 10.3390/s22020411

[13] L. K. Ramasamy, "F.K. KP, A.L. Imoize, J.O. Ogbebor, S. Kadry, and S. Rho, "Blockchain-based wireless sensor networks for malicious node detection: A survey,"," IEEE Access, vol. 9,128765–128785, 2021. DOI: 10.1109/ACCESS.2021.3111923
[14] J. Shahid, Z. Muhammad, Z. Iqbal, A. S. Almadhor, and A. R. Javed, "Cellular automata trust-based energy drainage attack detection and prevention in wireless sensor networks," Comput. Commun., vol. 191, pp. 360–367, 2022. DOI: 10.1016/j.comcom.2022.05.011
[15] K. Yesodha, M. Krishnamurthy, K. Thangaramya, and A. Kannan, "Elliptic curve encryption-based energy-efficient secured ACO routing protocol for wireless sensor networks," J. Supercomput., vol. 80, no. 13, pp. 1–34, 2024. DOI: 10.1007/s11227-024-06235-1
[16] M. A. Almaiah, "A new scheme for detecting malicious attacks in wireless sensor networks based on blockchain technology," in Artificial intelligence and blockchain for future cybersecurity applications. Cham: Springer International Publishing, 2021, pp. 217–234. DOI: 10.1007/978-3-030-74575-2_12
[17] A. B. Feroz Khan and G. Anandharaj, "A cognitive energy efficient and trusted routing model for the security of wireless sensor networks: CEMT," Wirel. Pers. Commun., vol. 119, no. 4, pp. 3149–3159, 2021. DOI: 10.1007/s11277-021-08391-6
[18] H. Hu, Y. Han, H. Wang, M. Yao, and C. Wang, "Trust-aware secure routing protocol for wireless sensor networks," ETRI J., vol. 43, no. 4, pp. 674–683, 2021. DOI: 10.4218/etrij.2020-0147
[19] J. R. Dora and K. Nemoga, "Clone node detection attacks and mitigation mechanisms in static wireless sensor networks," Journal of Cybersecurity and Privacy, vol. 1, no. 4, pp. 553–579, 2021. DOI: 10.3390/jcp1040028
[20] I. A. Abd El-Moghith and S. M. Darwish, "Towards designing a trusted routing scheme in wireless sensor networks: A new deep blockchain approach," IEEE Access, vol. 9,103822–103834, 2021. DOI: 10.1109/ACCESS.2021.3098933
[21] M. Hanif, H. Ashraf, Z. Jalil, N. Z. Jhanjhi, M. Humayun, S. Saeed, et al., "AI-based wormhole attack detection techniques in wireless sensor networks," Electronics (Basel), vol. 11, no. 15, p. 2324, 2022. DOI: 10.3390/electronics11152324
[22] Y. Han, H. Hu, and Y. Guo, "Energy-aware and trust-based secure routing protocol for wireless sensor networks using adaptive genetic algorithm," IEEE Access, vol. 10, pp. 11538–11550, 2022. DOI: 10.1109/ACCESS.2022.3144015
[23] S. A. Medjahed and F. Boukhatem, "Narwhal Optimizer: A Novel Nature-Inspired Metaheuristic Algorithm," Int. Arab J. Inf. Technol., vol. 21, no. 3, pp. 418–426, 2024. DOI: 10.34028/iajit/21/3/6
[24] Y. Tang, J. Yan, C. Chakraborty, and Y. Sun, "Hedera: A permissionless and scalable hybrid blockchain consensus algorithm in multiaccess edge computing for IoT," IEEE Internet Things J., vol. 10, no. 24, pp. 21187–21202, 2023. DOI: 10.1109/JIOT.2023.3279108
[25] L. Tojo, V. Maik, and M. Devi, "Image denoising using multi scaling aided double decker convolutional neural network," Optik (Stuttg.), vol. •••, p. 170350, 2022. DOI: 10.1016/j.ijleo.2022.170350
[26] A. Q. Tian, F. F. Liu, and H. X. Lv, "Snow Geese Algorithm: A novel migration-inspired meta-heuristic algorithm for constrained engineering optimization problems," Appl. Math. Model., vol. 126, pp. 327–347, 2024. DOI: 10.1016/j.apm.2023.10.045
[27] A. Keerthika and V. Berlin Hency, "Reinforcement-Learning based energy efficient optimized routing protocol for WSN," Peer-to-Peer Netw. Appl., vol. 15, no. 3, pp. 1685–1704, 2024. DOI: 10.1007/s12083-022-01315-6
[28] A. Keerthika and V. Berlin Hency, "Reinforcement-Learning based energy efficient optimized routing protocol for WSN," Peer-to-Peer Netw. Appl., vol. 15, no. 3, pp. 1685–1704, 2022. DOI: 10.1007/s12083-022-01315-6
[29] M. J. Rhesa and S. Revathi, "Energy-Efficient Aware and Secure Model for WSN Lifetime Enhancement Using HDASCII-AE and GEK-GRU," IEEE Access, vol. 12,140235–140252, 2024. DOI: 10.1109/ACCESS.2024.3461794
[30] S. Sivakumar, M. Mercy Theresa, K. Sudha, and K. Sangeethalakshmi, "Secure Wireless Sensor Networks: A Weighted K-NN and RNN-based Approach for Attack Detection and Localization," J. Inst. Electron. Telecommun. Eng., vol. 71, no. 3, pp. 864–875, 2025. DOI: 10.1080/03772063.2024.2436970

## Authors' Profiles

**Ranjeet Yadav,** received his Ph.D. in Electronics and Communication Engineering from Amity University Rajasthan, India. He completed his M.Tech in the Digital System and Computer Electronics from Jawaharlal Nehru Technological University (JNTU), Hyderabad, in 2011 and completed his B.Tech in Electronics and Communication Engineering in 2008 from the same University. In addition to it he also completed PGDDM (Post Gradutae Diploma in Disaster Management). He has several years of teaching experience at both Diploma and Engineering levels and has been actively involved in academic and research activities. Dr. Yadav has presented and participated in numerous national and international conferences and seminars, and has published research papers in reputed international journals. His current research interests include Mobile Communication, 5G/6G Technologies, Networking, and Wireless Communication.

**N. Manimegalai (Manimegalai Nithyanandam),** received B.E degree in CSE from Vel Tech High Tech Engineering College, Avadi affiliated to Anna University, Chennai, Tamil Nadu in 2013. M.E Degree in CSE from Velammal Engineering College (An Auotonoums Institution), Surapet affiliated to Anna University, Chennai, Tamil Nadu in 2024.

**E. Mercy Beulah,** is currently working as Assistant Professor in the Department of Computer Science and Engineering, Veltech Multitech Dr.Rangarajan Dr.Sakunthala Engineering College. She has 13 yrs. of teaching experience and her research interest includes Machine Learning, Deep Learning and Internet of Things. She is a MCA, M.E(CSE) graduate with Ph.D(CSE) from Dr. M G R Educational and Research Institute. She has published articles in reputed journals and international conferences.

**Mohammed Al-Farouni,** is a faculty member in the Department of Computers Techniques Engineering, College of Technical Engineering, at The Islamic University, Najaf, Iraq. His research spans diverse and interdisciplinary domains, including the analysis of legal texts, antenna arrays, artificial neural networks, blockchain technologies and their consensus mechanisms, as well as convolutional neural networks and other machine learning applications. He has contributed to advancements in fields ranging from signal processing to intelligent systems and secure decentralized networks, often integrating computational intelligence with emerging technologies to address practical and theoretical challenges.

**How to cite this paper:** Ranjeet Yadav, N. Manimegalai, Mercy Beulah E., Mohammed Al-Farouni, "Secure Blockchain-based Routing with Narwhal Optimization for WSNs", International Journal of Computer Network and Information Security(IJCNIS), Vol.18, No.1, pp.33-50, 2026. DOI:10.5815/ijcnis.2026.01.03