Multi-objective Bhattacharyya Enzyme Fossa Action-based CH Selection and Multi-agent Reinforcement Routing for Secure Cognitive Radio WSN

Anu Maria Joykutty * and B. Baranidharan
Department of Computing Technologies, School of Computing, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India
Email: aj7049@srmist.edu.in (A.M.J.); baranidb@srmist.edu.in (B.B.)

Manuscript received April 23, 2026; revised July 20, 2026; accepted August 10, 2026; published November 25, 2026
*Corresponding author

***Abstract*—Cognitive radio-Wireless Sensor Networks (WSN) are used to improve spectrum utilization in IoT systems with dynamic spectrum access. Nevertheless, they are highly challenged by energy imbalance, hot spot development at the sink, large computational burden, and susceptibility to malicious and byzantine attacks. Cluster Heads (CHs) that are nearer to the sink dissipate more energy because of the heavy data aggregation and relaying, resulting in network partitioning and decreased lifetime. In addition, existing routing and security protocols do not cooperatively optimize energy use, boundaries, throughput, and confidence. To solve the problems, this study designed a federated multi-agent reputation-based deep reinforcement learning framework with optimized cluster head selection for secure communication in cognitive radio-WSN. A Multi-Objective Bhattacharyya Enzyme Fossa Action (MO-BEFA) algorithm selects the optimal CHs based on residual energy, centrality, link quality, and spectrum availability. Then, a multi-agent reputation-based practical byzantine deep Maxout Attention Pyramid Spiking Local Federated Reinforcement Learning (MARPB-DASLFRL) framework is introduced for secure multi-hop routing. In this approach each CH learns optimal forwarding policies locally and shares model updates under a federated learning paradigm. A Reputation-based Practical Byzantine Fault Tolerance (R-PBFT) blockchain scheme provides more robust routing choices and prevention of model poisoning attacks. The simulation outcomes show improved energy efficiency, packet delivery ratio, throughput, latency reduction, and robustness than the current methods.**

***Index Terms*—cognitive radio-WSN, secure data transmission, federated learning, deep reinforcement learning, cluster head selection, routing, blockchain, Bhattacharyya distance, and enzyme fossa action**

I. Introduction

Over the past years there has been a huge rise in the number of countries where wireless services are available. This has led to widespread spectrum usage and has also resulted in spectrum scarcity. Cognitive radio technology has emerged as a solution to this issue [1]. A cognitive radio is an intelligent wireless communications system, capable of making use of its cognition of the radio operating environment. This knowledge renders the system to be able to adopt intra-cover opportunistic spectrum usage [2]. The natural characteristic of such a system to use the available spectrum when needed without compromising the efficiency in the utilization of the spectrum is a good solution to the spectrum problem [3]. By applying the unique features of cognitive radio technology, it is possible to manage many of the requirements and concerns introduced by Wireless Sensor Networks (WSNs) [4].

In a cognitive radio setup, WSN deploys multiple energy-limited sensor nodes in hostile conditions to construct an ad hoc network. Therefore, a routing protocol that is both scalable and energy-efficient is needed. Clustering has been suggested to satisfy these needs and increase the network lifespan [5]. However, a new technological issue arises when a routing algorithm is implemented is cognitive radio-WSN. The literature that is currently accessible demonstrates that the routing algorithms in Cognitive Radio Network (CRN) focus on solving the problem of spectrum scarcity and enhancing spectrum usage by merging the spectrum and route, but they do not address energy scarcity or hardware constraints associated with traditional WSNs [6]. The main focus of non-cognitive radio-WSN clustering methods is to decrease energy usage among resource-constrained nodes. They frequently disregard the crucial component of spectrum sensing and management. Consequently, these algorithms are unable to accomplish significant energy reduction inside cognitive radio-WSN and are unable to fully utilize spectrum resources [7]. The superior use of this precious resource is hampered by the complex problem of managing spectrum resources, which is yet unresolved. It becomes clear that traditional non-cognitive radio-WSN clustering algorithms lack a vital synergy between energy optimization and spectrum usage. The unrealized potential of these networks to provide the mutually beneficial advantages of lower energy consumption and optimal spectrum utilization highlights the need for creative solutions that achieve a harmonious balance between these elements [8, 9]. This complete

doi: 10.18178/ijeetc.15.6.421-437

integration offers to open new performance and efficiency dimensions inside cognitive radio-WSN, changing the WSN landscape and opening the way to more powerful and sustainable technological future.

Clustering is a popular method for lowering energy use by minimizing duplicate Data Transmission (DT) between nodes. This method groups sensor nodes into clusters, each of which is overseen by a Cluster Head (CH), who coordinates with the Base Station (BS) and other cluster members [10]. Several optimization techniques, including Particle Swam Optimization (PSO) [11], Artificial Bee Colony (ABC) [12], and Golden Jackal Optimization (GJO) [13], have been developed to determine the ideal CH, which is crucial for energy efficiency and network lifetime. Data security, dependency, and privacy are major issues for Internet of Things (IoT) [14]. Strong authentication and data encryption are critical security measures for preventing unauthorized access [15]. Lightweight encryption approaches like block and stream ciphers are appropriate for resource-constrained sensor nodes. Despite standard security methods [16], WSNs are nevertheless vulnerable to assaults, particularly insider attacks. To mitigate these insider threats, trust-based security measures have been devised, which rely on the cooperation and trustworthiness of network nodes [17]. Another essential component of Cognitive Radio-Wireless Sensor Networks (CR-WSNs) is secure and adaptive routing, along with effective clustering. CR-WSNs are particularly susceptible to attacks including jamming, packet dropping, malicious node intrusion, and spectrum sensing data fabrication because of their open wireless channels and dynamic spectrum access [18]. The intelligence to adjust to quickly shifting network conditions and security risks is lacking in conventional routing protocols. A possible solution is provided by Reinforcement Learning (RL) [19], especially agent-based learning which allows nodes to interact with their surroundings and learn the best routing principles. Reinforcement agents can dynamically choose safe and energy-efficient pathways by continuously assessing rewards based on energy consumption, latency, packet delivery ratio, and trust indicators [20].

Motivation:

The need for dependable, secure, and energy-efficient wireless communication has grown because of the quick development of smart environments, IoT devices, and mission critical monitoring systems. The lifespan and efficacy of conventional WSNs are greatly diminished by their restricted spectrum availability, unequal energy consumption, and susceptibility to malicious attacks. Through dynamic spectrum access, CR-WSNs increase spectrum utilization; nevertheless, they also present new difficulties, including sophisticated cluster management, changing channel availability, and increased security threats. These dynamic and multidimensional criteria cannot be addressed by traditional single-metric Cluster Head (CH) selection and static routing algorithms. An intelligent, flexible framework that can optimize several performance metrics at once while maintaining secure communication is therefore desperately needed. The requirement to improve energy economy, spectrum awareness, routing adaptability, and network security in next-generation cognitive radio WSNs drives the combination of an agent-based RL routing system with a multi-objective action module-based CH selection mechanism. The main objectives of this research are as follows:

- Energy-aware Cluster Head (CH) selection employing a Multi-Objective Bhattacharyya Enzyme Fossa Action (MO-BEFA) optimization, considering residual energy, node centrality, link quality, and spectrum availability.
- Multi-hop routing established using a multi-agent reputation-based practical byzantine deep Maxout Attention Pyramid Spiking Local Federated Reinforcement Learning (MARPB-DASLFRL) framework, allowing each CH to learn optimal forwarding policies locally while sharing model updates with a global aggregator.
- A deep maxout attention pyramid spiking network for local learning within a Federated Learning (FL) architecture.
- A Reputation-Based Practical Byzantine Fault Tolerance (PBFT) blockchain consensus mechanism implemented to validate routing and federated updates, addressing malicious behavior and attacks.

The remaining paper is organized as follows: Section II discusses the related state-of-the-art models, and the proposed methodology is explained in Section III. Section IV describes the results and discussion and the conclusion and future direction of this work are outlined in Section V.

II. Related Work

This section describes some of the recent related works as follows. Farhi *et al*. [21] presented a new approach to resource allocation optimization in CRNs based on RL and the Snake Optimizer (SO). Then, three distinct scenarios with variable numbers of Secondary Users (SUs), Primary Users (PUs), and accessible frequency bands are used to test snake optimizer. Finally, the achieved outcomes show that the suggested algorithms may effectively meet the previously described objectives and guarantees low collision rates and high spectrum utilization efficiency, which ultimately result in the highest feasible spectral capacity.

Kamatham *et al*. [22] suggested Node Grade Factor Multipath Routing (NGFMR), a node-disjoint multi-path routing protocol that is centered on the Node Grade Factor (NGF). Then, NGFMR optimizes route selection to improve network performance by including Quality of Service (QoS) variables such as node delay, communication costs, spectrum accessibility, and distance to the sink. According to the experimentation outputs, NGFMR greatly outperforms current protocol factors such as network throughput, packet delivery ratio, energy consumption, and end-to-end latency, indicating its fit for the sophisticated needs of next generation wireless networks.

A unique clustering strategy called the Unequal

Clustering Energy Hole Avoidance (UCEHA) was developed by Joon *et al*. [23]. UCEHA selects CHs by considering node energy consumption channels, neighbors, and sink distance, and divide the network into clusters according to sink proximity. Afterward, an improved spectrum-aware Ad Hoc on-Demand Distance Vector (AODV) method makes data routing more effective. Lastly, comprehensive experiments were carried out to test and validate the developed strategy, and the results show that UCEHA is better than current approaches in terms of lower energy use (19%), better network load balance (26%), longer network lifetime (40%), and higher throughput (8%).

To improve spectrum sensing efficiency, a study that integrates Stackelberg game theory into Tuna-Sandcat Optimization-Based Spectrum Sensing Approach (TSI-SCSA) was represented by Chilveri *et al*. [24]. The energy efficiency in CRN is considered while limiting the impact of Malicious Users (MUs) that provide misleading information. Even in hostile situations, the method guarantees sound decision-making and the best channel option. Finally, comparing the suggested TSI-SCSA model to current methods for effective spectrum sensing in CRN, the latter obtained a minimal miss detection value of 0.25 at iteration 100.

Namib beetle algorithm-based clustering and jump gain integral recurrent neural network for IoT (NBAC-JGTRNN-EE-IoT) is an energy-efficient framework introduced by Regilan *et al*. [25]. The NBA carries out clustering, and a depth-wise separable convolution combined with deep support vector machine (DSC-DSVM) was used to choose the Cluster Head (CH) to guarantee high accuracy in CH identification. In comparison to the other traditional models, simulation outcomes show that the suggested NBAC-JGTRNN lowers packet transmission to sink nodes by 26.78%, 31.82%, and 29.31%, respectively.

TABLE I: SUMMARIZATION OF EXISTING SCHEMES PERFORMANCE AND DRAWBACKS

Resource	Schemes	Performance	Drawbacks
[21]	RL-SO	High energy efficiency via state prediction.	High training time; poor in dynamic setups.
[22]	NGFMR	Reliable multi-path routing & high throughput.	Challenges of high routing overhead; path discovery failure.
[23]	UCEHA	Solves "hotspot" issues via unequal clusters.	Poor scalability; complex parameter tuning.
[24]	TSI-SCSA	Optimized data scheduling & node lifespan.	High computational load; prone to local optima issues.
[25]	NBAC-JGTRNN-EE-IoT	Strong privacy & reliable IoT data transmission.	Excessive overhead for simple sensor nodes.
[26]	EPSTM scheme	Low power via smart sleep/wake scheduling.	Inefficient with unpredictable/bursty traffic

IoT node authentication and harmful behavior mitigation using Evolutionary Particle Swarm Optimized Trust Management (EPSTM) scheme was proposed by Alotaibi *et al*. [26]. To inform secure routing decisions, the scheme assesses trust using network-specific characteristics such as device proximity, energy usage, data transmission dependability, and message delivery timing. According to the simulation results, the EPSTM scheme for a network of 100 nodes achieved an accuracy of 98.66%, demonstrating its efficiency for dependable, secure, and energy-efficient IoT communication. Table I gives a summary of all the works discussed above.

Problem Statement:

CR-WSNs use dynamic spectrum access to enhance communication and spectrum efficiency. However, short lifespan of the sensor nodes, cluster management, energy balance, routing effectiveness, and network security are the challenges of integrating CR technology with WSNs. Poor Cluster Head (CH) selection can lead to a network's energy depletion and reduced operational lifespan. Current methods for CH selection which focus on a single object (e.g.: remaining energy) do not sufficiently capture the complexity of CR-WSNs environments. In CR-WSNs environment, conditions like link quality and trust level of nodes, as well as channel stability and spectrum availability balance must be examined together. Routing in CR-WSN environments also becomes more complicated due to fluctuating channel conditions and security threats; with rogue nodes and spectrum sensing data falsification. Traditional routing systems are not capable of adapting to dynamic conditions, and because of this, routing delays, packet loss, and degraded quality of data are inescapable in the case of CR-WSNs. Therefore, a multi-agent-based reinforcement learning routing strategy that can dynamically identify safe and energy-efficient pathways is developed. This multi-objective function module-based CH selection mechanism optimally strikes a balance between energy efficiency, spectrum awareness, and reliability. Developing an integrated intelligent framework that enhances energy efficiency, spectrum usage, routing performance, and security inside CR-WSNs is the primary mission.

III. PROPOSED METHODOLOGY

This section describes the proposed methodology of selecting the CHs and the routing process of Cognitive Radio-WSN. The architecture of the proposed methodology is shown in Fig. 1.

Here, the cognitive radio-WSN starts with a random deployment of the sensor nodes in target area. Every sensor node considers energy, sensing, and cognitive radio capabilities to access the dynamic spectrum. At the sensor node, the spectrum sensing is used to detect the available channels and prevent interference with primary users. Once the sensor nodes are deployed, the fuzzy-C-means clustering is performed to cluster the sensor nodes. The optimal CH is then selected using the MO-BEFA algorithm, which is a multi-objective optimization approach based on residual energy, node centrality, quality of links, and spectrum availability. Finally, the secure MARPB-DASLFRL framework is employed. Through this framework, the deep maxout attention pyramid

spiking network is performed by a local learning model on the Federated Learning (FL) structure. Following this solution, CHs learn a deep reinforcement model individually, depending on the local states of the network, such as residual energy, spectrum availability, link quality, and reputation. Moreover, this approach improves the security of the framework using a R-PBFT blockchain consensus mechanism. Finally, the proposed algorithm ensures secure, and energy-efficient transmission of data in cognitive-WSN environment.

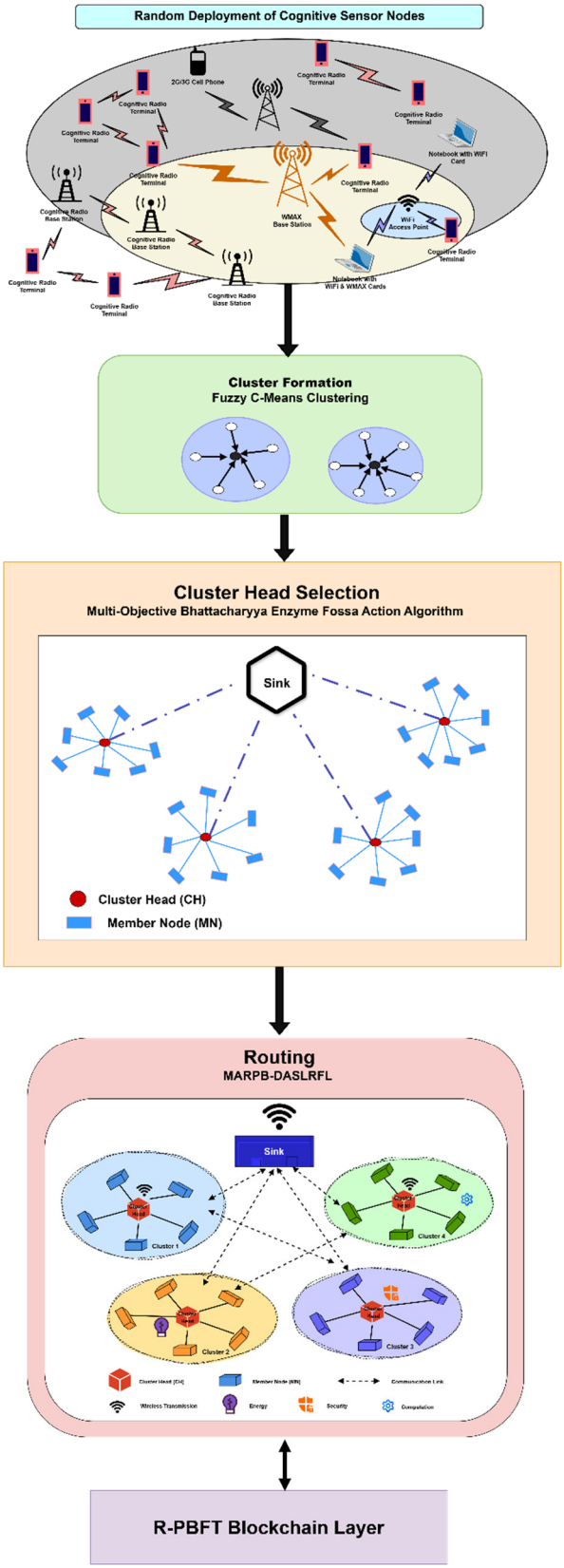

Fig. 1. Work flow structure of the proposed cognitive radio-WSN framework.

A. Random Node Deployment

In this phase, the cognitive WSN randomly deploys 100 sensor nodes over the sensing field. To enhance the practical applicability in a remote environment, a random deployment process is used. Each IoT node becomes self-configured by calculating the important communication parameters used to connect to the network. Communication parameters can be network topology, node density, and environmental conditions [27]. The deployed sensor nodes are given as:

$$\boldsymbol{d}_k = [d_{\mathrm{k1}},\ d_{\mathrm{k2}}, \cdots,\ d_{\mathrm{k100}}] \tag{1}$$

where $\boldsymbol{d}_k$ is denoted as the set of all deployed sensor nodes, each of which corresponds to the 100 sensor nodes in the WSN. Each sensor node is set with a starting energy, sensing ability, and capability of cognitive radio, as well as a communication radius. The configuration of the communication parameter of each node is:

$$\mathrm{Parametersetup}(\boldsymbol{v}_k) = [v_{\mathrm{k1}}, v_{\mathrm{k2}}, \cdots, v_{\mathrm{k100}}] \tag{2}$$

where $\boldsymbol{v}_k$ is the communication parameters associated with each sensor node. All sensor nodes consist of common parameters such as energy, sensing capability and cognitive radio capability in the WSN. The network connectivity between the sensor nodes is represented using the adjacency matrix $\boldsymbol{N}$ as given below:

$$\mathrm{Networkconnection}(\boldsymbol{N}) = \begin{bmatrix} n_{1(1)} & n_{1(2)} & \cdots & n_{1(100)} \\ n_{2(1)} & n_{2(2)} & \cdots & n_{2(100)} \\ \cdots & \cdots & \cdots & \cdots \\ n_{100(1)} & n_{100(2)} & \cdots & n_{100(100)} \end{bmatrix} \tag{3}$$

where $\boldsymbol{N}$ is denoted as the adjacency matrix of the network topology, where $n_{k(g)}$ is denoted as the connection status between the sensor nodes d_k and d_g. Each entry $n_{k(g)}$ can take on a value of 1 if a connection between sensor nodes d_k and d_g exists, and 0 if there is no connection. Once the node deployment process is completed, then spectrum sensing is performed. This helps to identify available channels and avoid interference with primary users. The set of channels available is given as:

$$\boldsymbol{C} = [C_1, C_2, \cdots, C_j] \tag{4}$$

where $\boldsymbol{C}$ is the available communication channels and C_J is the j_{th} channel. The nodes are used to assess the availability of the channel based on the detection probability P_d, the false alarm probability P_f, the Signal-to-Noise Ratio (SNR), and activity by the primary user. A channel C_j is assumed to be available when it meets the conditions of sensing threshold:

$$P_d(C_j) \geq \gamma_d \text{ and } P_f(C_j) \leq \gamma_f \tag{5}$$

where the predefined detection threshold is denoted as γ_d, and threshold of false alarm is denoted as γ_f. The connectivity matrix is modified between communication range R and channel availability, which is represented as:

$$a_{ij} = \begin{cases} 1, & \text{if } d(d_i, d_j) \leq R \text{ and common channel exists} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

Lastly, this deployment phase assists in developing the original network topology.

B. Cluster Formation Using Fuzzy C-Means Clustering

The network nodes are prorated according to the clustering algorithm like Fuzzy C-Means Clustering (FCM) [28]. The technique is common in pattern recognition in which membership is assigned to the data point of each cluster center such that the summation of membership of all data points must be equal to one. The minimization of the following objective function is used to obtain FCM:

$$\min_{\mu_{xy}, K_y} (\vartheta_u) \tag{7}$$

and

$$\vartheta_u = \sum_{x=1}^{L} \sum_{y=1}^{K} \mu_{xy}^{u} \left\| \theta_x - K_y \right\|^2 \tag{8}$$

where K represents the number of clusters, u represents the fuzziness exponent, which is more than 1, μ_{xy} represents the degree of membership of x_{th} sensor in the cluster y, and K_y represents the center of the cluster y. The value of μ_{xy} is between 0 and 1 for all the sensor in the network and each cluster center. This fuzzy partitioning is accomplished by extensive iterative optimization of the objective function membership μ_{xy} and the new cluster center K_y as:

$$\mu_{xy} = \frac{1}{\sum_{y=1}^{K} \left(\frac{\theta_x - K_y}{\theta_x - K_m} \right)^{\frac{2}{u-1}}} \tag{9}$$

and

$$K_y = \frac{\sum_{x=1}^{L} \mu_{xy}^{u} \cdot \theta_x}{\sum_{x=1}^{L} \mu_{xy}^{u}} \tag{10}$$

The iteration optimization terminates when the termination criterion σ is achieved i.e., $\{\mu_{xy}^{m+1} - \mu_{xy}^{m}\} < \sigma$, where m is the iteration step. The process of the cluster formation is described in Algorithm 1.

After cluster formation, the network is split into clusters and the CH selection process is started locally within each cluster.

C. Cluster Head Selection Using MO-BEFA-algorithm

In this section, the CH selection strategy is introduced using the suggested Multi-Objective Bhattacharyya Enzyme-Fossa Action (MO-BEFA) optimization algorithm. The MO-BEFA algorithm is designed to effectively choose the best CHs, by considering the residual energy, centrality of the node, link quality, and

spectrum availability simultaneously. It optimizes these parameters and enhances the energy efficiency, network stability, communication reliability, and provides efficient spectrum utilization.

Algorithm 1: Cluster Formation

Initialization:
Membership value μ_{xy} $\forall y = 1,2,\ldots,K\ \forall x = 1,2,\ldots L$
Cluster center initialized
While $\{\mu_{xy}^{m+1} - \mu_{xy}^{m}\} < \sigma$ do
  For $y = 1,2,\ldots,K$ do
    $K_y = \frac{\sum_{x=1}^{L} \mu_{xy}^{u} \cdot \theta_x}{\sum_{x=1}^{L} \mu_{xy}^{u}}$
  End for
  For $x = 1,2,\ldots,L$ do
    For $y = 1,2,\ldots,K$ do
      Which is μ_{xy}^{m}
        If $\|\theta_x - K_y\| > 0$ **then**
        Calculate μ_{xy} as $\mu_{xy} = \frac{1}{\sum_{s=1}^{K} \left(\frac{\theta_x - K_y}{\theta_x - K_m}\right)^{\frac{2}{u-1}}}$ **Which** is μ_{xy}^{m+1}
        End if
      End For
  End For
End While

The proposed MO-BEFA algorithm incorporates two optimization approaches such as enzyme action-based exploration and fossa-based exploitation. Enzyme action [29] mechanism improves global search (exploration) by searching in different parts of the solution space, which prevents premature convergence and increases the diversity of solutions. The fossa optimization [30] method reinforces the use of local search (exploitation) which ensures optimization of promising candidate solutions and quick convergence of optimal cluster head configurations. Moreover, Bhattacharyya distance is introduced to the optimization process to increase diversity and solution discrimination. Based on the assumption that candidate solutions are modelled based on Gaussian distribution, this metric is the measure of the closeness between probability distributions by quantifying their overlap to minimize redundancy and maximize selective accuracy.

Let $\boldsymbol{S} = [\mathrm{CH}_1, \mathrm{CH}_2, \cdots, \mathrm{CH}_k]$ corresponds to the selection of cluster heads in a cluster. The multi-objective fitness function is defined as:

$$\text{Fitnessfunction} = \text{Maximize(FS)} \tag{11}$$

where

$$\mathrm{FS} = \omega_1 f_{\mathrm{RE}} + \omega_2 f_{\mathrm{NC}} + \omega_3 f_{\mathrm{LQ}} + \omega_4 f_{\mathrm{SA}} \tag{12}$$

Here f_{RE} is denoted as the residual energy of candidate node, f_{NC} is denoted as the centrality measure, f_{LQ} is denoted as the link quality indicator, f_{SA} is denoted as the spectrum availability factor, and ω_1, ω_2, ω_3, ω_4 is denoted as the weight coefficients adjusting the significance of each objective.

Mathematical model of MO-BEFA-algorithm:

The mathematical formulation of the MO-BEFA-Algorithm is described in this section, with the initial stage being the use of cluster initialization. The number of search agents is denoted by *G*, the maximum number of iterations by *I*, and the dimensionality of the search space (number of candidate CH node within a cluster) by *d*. Each decision variable is bounded by lower bound (P) and upper bound (Q), defining the feasible region. The initial population of search agent is generated randomly as:

$$\boldsymbol{M}_x^{(0)} = P(Q-P)\Theta \boldsymbol{j}_x,\ x = 1,2,\cdots,G \tag{13}$$

In this case, the *x*th search agent starting position is represented by vector $\boldsymbol{M}_x^{(0)}$, the uniformly distributed random vector is represented by $\boldsymbol{j}_x \in [0,1]^d$, and the symbol of Θ is denoted as the element-wise multiplication. Every search agent corresponds to a possible CH setup in a cluster.

The enzyme action mechanism then uses adaptive catalytic upgrades to improve global exploration. For iteration $i = 1, 2, \cdots, I$, an Adaptive Factor (AF) is defined as:

$$\mathrm{AF}_i = \sqrt{i/I} \tag{14}$$

This square root relationship ensures a gradual increase in the AF as the iteration progress. Each $\boldsymbol{M}_x^{(i+1)}$ search agent at iteration *i* produces two potential positions. The first search agent candidate position is defined as:

$$\boldsymbol{M}_{x,1}^{(i)} = \left(\boldsymbol{M}_{best}^{(i-1)} - \boldsymbol{M}_x^{(i-1)}\right) + \boldsymbol{\rho}_x \Theta\, sin\left(AF_i \boldsymbol{M}_x^{(i-1)}\right) \tag{15}$$

where $\boldsymbol{\rho}_x$ is denoted as the random vector in $[0,1]^d$, and sin is used in an element-wise manner. The resultant position $\boldsymbol{M}_{x,1}^{(i)}$ is constrained to the interval $[P, Q]$ to attain the feasible position. To produce the second candidate position of the search agent, two distinct search agent indices a, b are selected from the set $\{1, 2, \cdots, G\}$. These indices characterize the vector as in Eq. (16).

$$\boldsymbol{v} = \boldsymbol{M}_a^{(i-1)} - \boldsymbol{M}_b^{(i-1)} \tag{16}$$

where this difference serves to inject additional diversity into the search process. The second candidate position of search agent is defined in Eq. (17). It is built by incorporating the element introduced above:

$$\boldsymbol{M}_{x,2}^{(i)} = \boldsymbol{M}_x^{(i-1)} + s_1 \boldsymbol{v} + AF_i \cdot s_2 \left(\boldsymbol{M}_{best}^{(i-1)} - \boldsymbol{M}_x^{(i-1)}\right) \tag{17}$$

The coefficients s_1 and s_2 are random scale values in the interval [EC, 1], where EC is the enzyme concentration (a small constant, e.g. 0.1). This position is also constrained with $[P, Q]$ to preserve feasibility. These adaptive catalytic updates improve the global search performance of the MO-BEFA algorithm by performing effective searching of new regions in the solution space

through the enzyme action mechanism.

Once enough exploration has been done, the algorithm switches to the fossa exploitation stage to optimize the promising candidate solutions. In this phase, small and controlled positional adjustments are used in a limited area within the search space to enhance accuracy of convergence. The local update rule of each search agent is given as:

$$M_{x,y}^{\mathrm{E2}} = m_{x,y} + (1 - 2j_{x,y}) \cdot \frac{(Q_y - P_y)}{i} \quad (18)$$

where $j_{x,y} \in [0,1]$ is the random uniform number, i is the current iteration and $M_{x,y}^{\mathrm{E2}}$ is the yth dimension of the new position of the x_{th} search agent that has been computed in the exploitation phase. $m_{x,y}$ is the yth dimension of the current position of the x_{th} search agent, Q_y and P_y are respectively the upper bound and lower bound at the yth dimension. The step size decreases progressively as the iteration number increases because of the division by i, which allows the fine-grained local adjustments in the high-quality solutions.

By using Eq. (18) for all y=1, 2, ···, d, gives the full exploitation position vector $\boldsymbol{M}_x^{E2}$. The new position is only adopted when it positively affects the objective function value, which is defined as:

$$\boldsymbol{M}_x^{(i)} = \begin{cases} \boldsymbol{M}_x^{E2}, & if\ F(\boldsymbol{M}_x^{E2}) > F(\boldsymbol{M}_x^{(i-1)}) \\ \boldsymbol{M}_x^{(i-1)}, & \text{otherwise} \end{cases} \quad (19)$$

where $M_{x,y}^{\mathrm{E2}}$ is denoted as the y_{th} dimension of this new position.

Bhattacharyya distance is used to measure the similarity of the candidate solutions which are distributed as gaussian and to further improve the exploration and exploitation of solutions. To find the Bhattacharyya distance [31] of two Gaussian distributions $P_i = (X; \mu_i,\ \Sigma_i)$ and $P_j = \left(X; \mu_j,\ \Sigma_j\right)$ the formula used is:

$$B_{\mathrm{ij}} = \frac{1}{8}\left(\mu_i - \mu_j\right)^T \Sigma^{-1}\left(\mu_i - \mu_j\right) + \frac{1}{2}\left(\frac{\det \Sigma}{\sqrt{\det \Sigma_i \det \Sigma_j}}\right) \quad (20)$$

where the shared covariance matrix is represented as $\Sigma = \frac{\Sigma_i + \Sigma_j}{2}$. Finally, the algorithm effectively selects the CH node in the clustering nodes. The detailed process of the MO-BEFA selecting the CH is explained in Algorithm 2.

Algorithm 2: CHs using MO-BEFA- Algorithm

Input: G, I, P, Q, *fitness function*

Output: CHs

Step 1: Initialization

Clustering search agent initialization as performed by Eq. (13)

For $x = 1 \rightarrow G$ do

Evaluate fitness function using Eq. (11)

End For

Step 2: Exploration

For $i = 1 \rightarrow I$ **do**

Compute AF_i using Eq. (14)

For $x = 1 \rightarrow G$ do

Calculate the first search agent candidate position using Eq. (15)

Calculate the second search agent candidate position using Eq. (17)

End For

End For

Step 3: Exploitation

Calculate local update rule of each search agent using Eq. (18)

New position of local search agent position is calculated using Eq. (19)

Step 4: Enhance the candidate solution in Exploration and Exploitation

Bhattacharyya distance is calculated by the formula Eq. (20)

Return CHs

D. Multi-hop Routing Using MARPB-DASLFRL Framework

The multi-agent reputation-based practical byzantine deep Maxout Attention Pyramid Spiking Local Federated Reinforcement Learning (MARPB-DASLFRL) structure for multi-hop routing constitutes a safe, decentralized network that makes use of blockchain for consensus and federated learning for local training of algorithms as shown in Fig. 2. It emphasizes sustainable, privacy-preserving routing within flexible networks. Each CH uses a local system of learning called the Deep Maxout Attention Pyramid Spiking Network (DMAPSN). It incorporates Spiking Neural Networks (SNNs) for energy-effective event-driven interpreting, attention systems for path prioritizing, and maxout units for enhanced activation. The system uses a federated learning model in which every CH occasionally sends its trained model variables and biases to a global aggregator, to avoid sharing unprocessed network data. Routing decisions and model updates are validated through the reputation-based PBFT Blockchain to offer security and confidence.

Multi-agent module: Each CH is represented in the multi-agent module as an independent reinforcement learning agent in charge of making local routing choices for the cluster. The best next hop for packet routing is determined through these agents by monitoring dynamic network metrics like buffering load, neighbor accessibility, residual power, and connection reliability. The approach improves stability across dense networks and prevents single-point errors through decentralizing decision-making processes. Adaptive policy learning for dependable and energy-effective multi-hop routing is made possible through every CH maintaining a local state representation and constantly interacting with surrounding node circumstances.

1) Local learning model using deep maxout attention pyramid spiking network

The input for the learning algorithm is the constant observation of every CH's regional network state. Residual energy, queue duration, packet delivery rate, link security, neighbor reputation score and hop length towards the sink are all included in the current state vector. These

characteristics allow the agent to generate choices about routing that are dependable, attack-resistant and energy-effective by capturing both QoS and safety context. Reliable deep reinforcement training for frameworks is achieved using feature normalizing and dimensional matching.

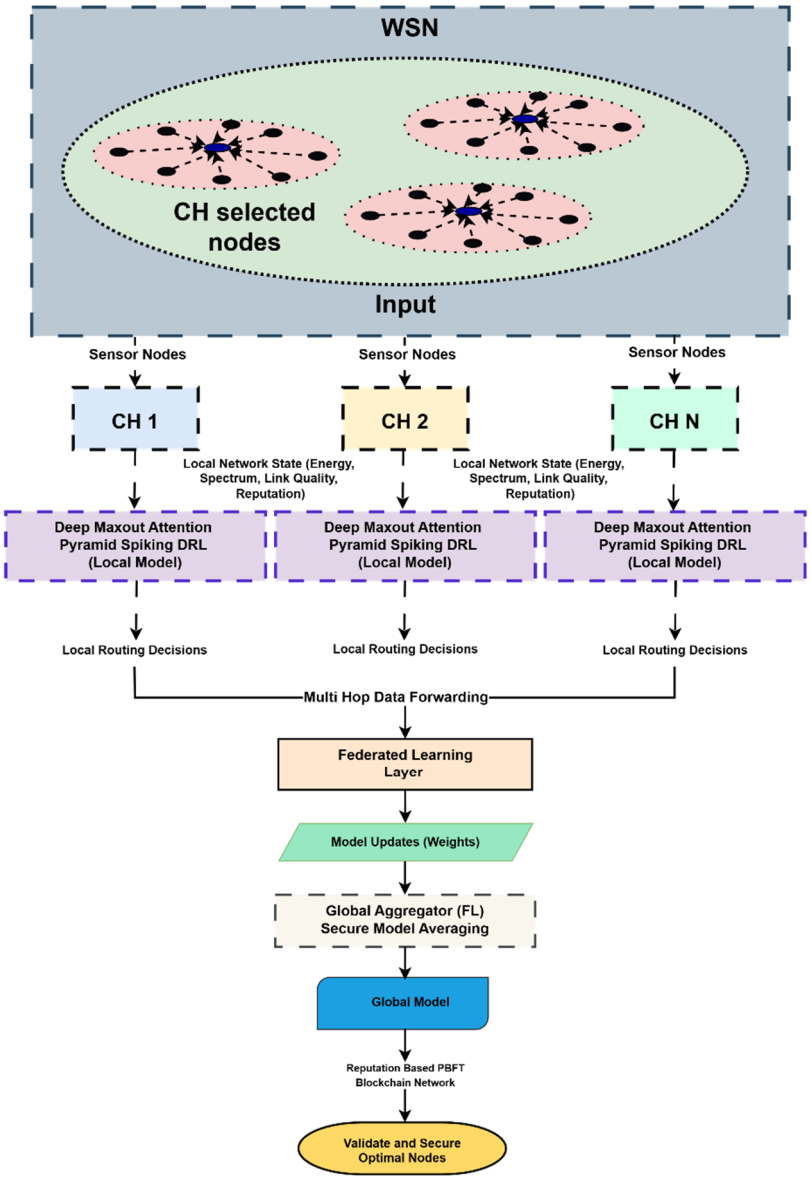

Fig. 2. Architecture of MARPB-DASLFRL.

As the reinforcement learning strategy approximator, every CH learns a deep maxout attention pyramid spiking network. While the attention method gives priority to important qualities like low-energy nodes, crowded links and neighbors with a poor reputation, the pyramidal arrangement captures a multi-scale spatial-temporal routing process. The spiking neurons allow event-driven, energy-efficient computing appropriate for WSN contexts, and the deep maxout units enhance non-linear representations and reduce vanishing gradients. An ideal next-hop decision policy that optimizes progressive reward according to trust, energy utilization, delay, and execution success is produced through the framework. Implementing Deep Maxout Network (DMN) while selecting the CH improves the selection process effectiveness by avoiding non-optimal nodes [32].

Maxout: For an input $\delta \in \eta^{\varepsilon}$ the output maxout unit is represented as:

$$\mathrm{MO}_x = \frac{\max}{y \in [1,l]} \tau_{xy} \tag{21}$$

From Eq. (21), $\tau_{xy} = \delta^k \omega_{xy} + b_{xy};\ \omega \in \eta^{\varepsilon\phi l}$; l denotes the total amount of linear sub-hidden units and $b \in \eta^{\phi l}$ indicates the training characteristics. Maxout proves to be more resilient than ReLU for the task at present as it offers a learnable, adaptable activation function that prevents problems like fading neurons, integrates effectively with dropout and expands the model's ability to represent complicated nonlinearities.

DMN: An activation function may be utilized to train

the multi-layer architecture. The following formula can be used to determine the hidden layer unit's activation. For the input $\delta \in \eta^{\varepsilon}$,

$$Q_{x,y}^{1} = \frac{\max}{y\in[1,l_1]}\ \delta * \ \omega_{xy} + b_{xy} \tag{22}$$

$$Q_{x,y}^{2} = \frac{\max}{y\in[1,l_2]}\ Q_{x,y}^{1}\ \omega_{xy} + b_{xy} \tag{23}$$

$$Q_{x,y}^{\phi} = \frac{\max}{y\in[1,l_\phi]}\ Q_{x,y}^{\phi-1}\ \omega_{xy} + b_{xy} \tag{24}$$

$$Q_{x,y}^{n} = \frac{\max}{y\in[1,l_n]}\ Q_{x,y}^{n-1}\ \omega_{xy} + b_{xy} \tag{25}$$

$$MO_{x} = \frac{\max}{y \in [1, l_{n}]} Q_{x,y}^{n} \tag{26}$$

where l_ϕ denotes the quantity of units within the ϕth layer and n indicates the numbers of DMN levels, according to Eq. (21). Deep maxout network has been chosen as it offers adaptable nonlinear depiction, better regularization, computational effectiveness, and compatibility for noisy information sets, making it well suited for WSN applications.

Attention pyramid: Scale agnostic attention may be expanded to pyramid attention, which calculates correlations across several scales, to fully utilize self-predictive capability [33]. Nodes relationships are recorded over the whole feature pyramid in these units. In particular, a feature pyramid $\boldsymbol{F} = [F_1, F_2, \cdots, F_n]$ is formed given a set of scales $\boldsymbol{S} = [1, s_1, s_2, \cdots, s_n]$. The area descriptor map of input x, $F_i\left(\frac{H}{s_i}\cdot\frac{W}{s_i}\right)$ generated through down-scaling the scales. In this situation, it is possible to observe the correlations among any pyramid levels and the initial input x as scaling agnostic attention. Since area descriptor z_i at multiple tiers aggregates data over regions with various sizes, the pyramid attention can be stated as follows:

$$y^{i} = \frac{1}{\sigma(x,F)}\sum_{z\in F}\sum_{j\in z}\varphi(x^{i}, z^{j})\,\theta(z^{j}) \tag{27}$$

This enables cross-scale simulation. Non-local multi-scale data is combined to create new responses when the results are transmitted back to the initial location i. This response provides more accurate data than the feature representations from individual node positions.

Spiking neural network: This study uses a Spiking Neural Network (SNN) to extract node features from the input data created during the cluster head stage [34]. Multiple nodes constitute an SNN; each node receives input data and produces an output pattern independent of other node decisions. A weight is assigned to every node connection and this weight is modified during learning through supervised and unsupervised techniques. Encoding technique converts the input data into spike trains to represent the highs and lows in the data. The SNN used has two layers, each of which has two conventional layers and fully interconnected layers. The input convolutional layer with a size of 3 × 3 and 128 feature channels, processes the sensor node input data and generates a feature map of size 32 × 32 × 128, which is passed to the subsequent layer. To reduce the feature map dimensions while maintaining important features, the spike outputs from the convolutional layer are collected across the temporal dimensions and handled utilizing an average pooling layer of size 2 × 2, producing a reduced feature map of size 16 × 16 × 128. This pooling operation is subsequently repeated using another average pooling layer with a kernel size of 2 × 2, further reducing the feature map to a size of 8 × 8 × 128. During several time steps, the node simultaneously releases spikes. Event-driven computing and low-power devices operation are made possible by the spiking system, making it appropriate for CH nodes with limited resources.

2) Optimal routing policy using reinforcement learning

The locally gathered experience tuples consisting of state, action, reward are independently used by each CH to train its routing policy [35]. The primary purpose of reinforcement learning approach is to assess the reliability and efficiency of data transmission by improving packet delivery accuracy. To determine the optimal routing path and to select the effective CH, a RL framework is used. At the beginning, the reinforcement vector is initialized randomly for all sensor nodes. Every node gathers environmental information and sends it over to the next node. Reinforcement learning manages the entire network to choose the most efficient CH by determining which node is best suited to transmit data to the base station. The learning strategy uses the state function, action and reward to assess the reinforcement value. Residual energy, spectrum availability, link quality, and reputation scores are indications of locally observed network states that are used to create a multi-objective reward function for reinforcement learning models.

Residual energy: The amount of battery power that remains in a sensor node after energy is used for various tasks and adding energy obtained through harvesting or replenishing is given as:

$$E_{\text{res}}(i) = E_{\text{initial}}(i) - E_{\text{consumed}}(i) \tag{28}$$

Spectrum availability: The possibility that a Secondary User (SU) can use a certain channel for communication since the Primary User (PU) does not utilize the channel at present.

$$S_{\text{aval}} = P(H_0|Y_{\text{SU}}) \tag{29}$$

where H_0 denotes probability of hypothesis value and Y_{SU} denotes the receiving node.

Link Quality (LQ): A numerical indicator of a wireless channel's dependability that shows the probability of packet transmission success among two nodes.

$$LQ = \frac{\text{Successfully recieved nodes}}{\text{Total node transmitted}} \quad (30)$$

Reputation scores: A numerical representation of the node's reliability, that ranges between malicious and trusted, based on its past actions such as forwarding packets, falling, and malicious activities.

$$R_{\text{score}} = \frac{\sum \text{Forwared nodes}}{\sum \text{Forwarded nodes} + \sum \text{Dropped nodes}} \quad (31)$$

The reward for the whole program is determined using the metrics that were gathered in the reward function. The reinforcement value is calculated using these reward computations.

$$R = \alpha E_{\text{res}}(g_k) + \beta S_{\text{aval}}(g_k) + \gamma LQ(g_k) + \delta R_{\text{score}}(g_k) \quad (32)$$

where $\alpha, \beta, \gamma, \delta$ denotes weight balancing parameters. For simplicity, and to ensure equal contribution of all reward components, the weight values are set uniformly at 0.25. The network's most efficient and effective CH are produced by the obtained reinforcement value. As a result, the suggested approach minimizes energy usage and latency while avoiding malicious and low-reputation networks.

3) Aggregation using FL model

Every CH regularly uploads its local model settings and variations into a global aggregator rather than exchanging raw routing information [36]. A global routing strategy is created by the aggregator through secure federated averages, and it is subsequently disseminated to every CH. In addition to saving training time and protecting data privacy, this collaborative learning improves generalization across various clusters. The approach is appropriate for networks with limited bandwidth because it also drastically reduces communication cost as compared to centralized learning. A decentralized machine learning approach called Federated Learning (FL) enables several organizations to work together to train a model without disclosing their raw data. By allowing collaborative training of frameworks and maintaining the information local among clients, FL tackles the issues of privacy, security and accessibility.

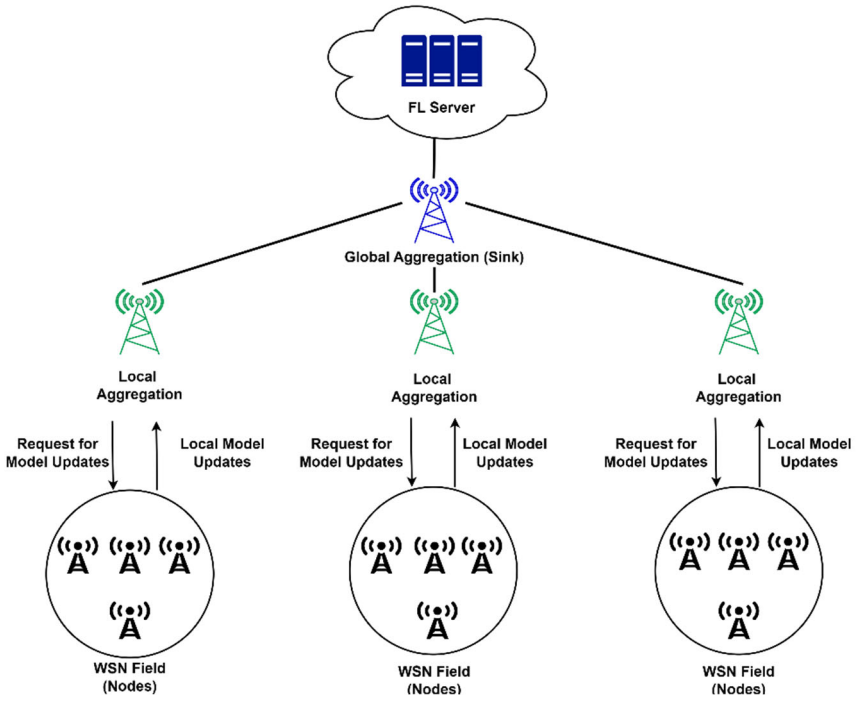

Fig. 3. Federated learning process.

To guarantee productive and successful cooperation among sensor nodes while maintaining data privacy and lowering communication overhead, the combination of WSN and federated learning necessitates carefully planned architectures. The architectures for combining WSNs and FL components are covered in Fig. 3.

Sensor Nodes (Clients): Every sensor node gathers information, trains the model locally, and transmits the updated model to the central server.

Central Server: The central server creates a global model by combining the model updates from every sensor node and then sending it through the sensor nodes.

Initialization: All sensor nodes receive the global model after it has been initialized by the central server.

Local Training: Using its own data, each sensor node teaches the local model before sending the updated model to the centralized server.

Model Aggregation: To create an entirely new global approach, the central server combines the model's updates from every sensor node.

Model Distribution: The sensor nodes receive an upgraded global model from the centralized server.

FL in WSN environment: In the context of WSNs, FL involves training a model using machine learning across

several decentralized sensors (nodes) that keep local data without transferring it. Instead, a central server receives model updates such as gradients or weights from the nodes and aggregates them to enhance the global model. This study assumes a few things to construct a mathematical formula for federated learning through WSN. The WSN is compose of N sensor nodes, each of which has a local database D_i for $i = 1, 2, 3, 4, 5, \cdots, N$. Every sensor node can broadcast its updates to an aggregation center (base station) and perform local calculations. The local databases are dispersed among the nodes independently and non-identically. Power, latency and bandwidth limitations affect interactions between nodes and the aggregator. The global objective function $F(\boldsymbol{w})$, where $\boldsymbol{w}$ stands for the model parameters, has been optimized through the FL algorithm. Updates are combined and distributed to nodes during synchronized rounds of communication.

Objective Function for Federated Learning: FL aims to reduce the global objective function $F(\boldsymbol{w})$, which is represented by Eq. (33),

$$F(\boldsymbol{w}) = \frac{1}{D}\sum_{i=1}^{D} F_i(\boldsymbol{w}) \tag{33}$$

where D denotes the total data, $F_i(\boldsymbol{w})$ denotes the local objective function of node i, which is defined as follows:

$$F_i(\boldsymbol{w}) = \frac{1}{|D_i|}\sum_{(x,y)\in D_i} l(\boldsymbol{w}, x, y) \tag{34}$$

The loss function $l(w, x, y)$ measure the prediction error of the model with parameters $\boldsymbol{w}$ on a given data sample (x, y).

Local Updates: Every sensor node i uses a data called D to conduct local training. Stochastic Gradient Descent (SGD) is used to update the local model;

$$\boldsymbol{w}_i^{t+1} = \boldsymbol{w}_i^t - \eta \nabla F_i(\boldsymbol{w}_i^t) \tag{35}$$

Thus, $\boldsymbol{w}_i^t$ denotes model parameters by node i at time t, η indicates learning rate and $\nabla F_i(\boldsymbol{w}_i^t)$ indicates the gradient of the function $F_i(\boldsymbol{w}_i^t)$.

Global Aggregation: Following local updates, the global model $\boldsymbol{w}_t$ is updated through the aggregator for calculating the weighted mean of the local models:

$$\boldsymbol{w}^{t+1} = \frac{1}{D}\sum_{i=1}^{D} |N|\, \boldsymbol{w}_i^{t+1} \tag{36}$$

where $|N|$ stands for the local data size. Latency and energy limitations have a big influence on communication for WSNs. Considering this, the global federated learning process can be stated as follows:

$$F(\boldsymbol{w})_{min} = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{|N|}\sum_{(x,y)\in N} l(\boldsymbol{w}, x, y) \tag{37}$$

Although the FL architecture balances resource environment flexibility, and communication overhead, it necessitates precise cluster head management.

4) Reputation based practical Byzantine blockchain validation

The federated updates to models and routing decisions are secured by integrating a reputation-aware Practical Byzantine Fault Tolerance (PBFT) [37] method. Based on previous results, including forwarding packets integrity, update reliability, and energy reporting efficiency, every CH keeps a reputation score. Block validation, which records validated routing policies and FL modifications in a permanent database, is carried out by high-reputation nodes following consensus. This ensures trust, integrity, and byzantine resistance by preventing malicious nodes from inserting poisoned models, initiating sybil attacks, and altering routing pathways.

An enhanced consensus procedure has been formed to minimize complicated interaction and enhance resiliency towards node-capture assaults. In the proposed framework only the central node submits the commit message to all clients, and this differs from the traditional PBFT where it is done by multiple nodes. Interaction among the nodes takes place through two-by-two patterns instead of three-by-three patterns during the commit and reply stages of consensus. The node having the second-most reputation score serves as both a replica and a potential node for master's registration. Defective node takes role in the consensus procedure in order to distinguish the compromised node and the failed node. To reduce the false positive rate, a node having a reputation score is permitted to take over the consensus procedure when it is adversely affected through problems like slight latency, which might have penalized the new node. The five stages of conventional PBFT consensus process are shown in Fig. 4.

Request stage: The client generates a request response and sends it to both the candidate nodes and the master node comprising a valid signature that includes a timestamp.

Pre-prepare stage: After confirming receipt of the request message, the master node transmits the prepared message across the replica nodes. Before receiving the message, the replica nodes verify that it has the proper signature including the allotted sequence numbers. The candidate node compares the initial request message by the client with the pre-prepared response from the master node.

Prepare stage: The replica nodes broadcast the prepared message across every other replica after obtaining the pre-prepared signal and verifying data. To notify all replicas that the main node was recently compromised, the possible candidate node may include the deny message headers and evidence of the client's legitimate request within the prepared message if it has received a "master denied" verification update from the pre-prepare stage.

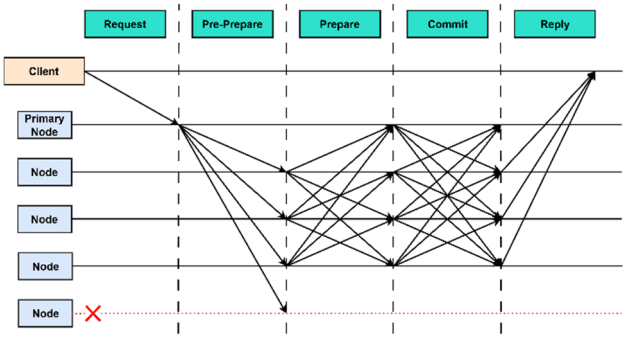

Fig. 4. Reputation based practical byzantine consensus process.

Commit stage: The replica nodes check the prepared message received from the additional replica nodes. If authentication works properly, they transmit a consent message to the master node; if not, they send a refuse message and start a change process. If the master node receives a minimum of $2f + 1$ consent messages, it confirms that the system has reached a secure committed state and broadcasts the commit phase message which includes the reputation ratings of all replica nodes to every replica node, thereby synchronizing the current state across the entire network.

Reply stage: Following the execution of the commit stage, the master node stores the content into its local records and notifies the client with the consensus set up. Based on their actions during the consensus round, each node receives incentives and penalties for reputation ratings. By giving priority to trusted nodes and streamlining communication, PBFT enables the blockchain to manage many transactions with minimal latency, making it appropriate for WSN.

The PBFT blockchain rejects harmful contributions using reputation ratings to confirm the legitimacy of model updates and decisions regarding routing. CHs receive a broadcast of the verified global model, which provides them with network-wide information to improve their local policies. The procedure is repeated, allowing for Byzantine-resilient, flexible and confidentiality multi-hop routing.

IV. Simulation Results and Discussion

In this context, a simulation with 100 sensor nodes dispersed throughout different network regions is used for analysis. The simulation environment used for the experiment is MATLAB. A data unit of 1024 bytes is carried by each node, and the simulations were executed across different network configurations. 10, 50, and 100 joules are the three starting energy levels considered. The results of multiple metrics, including energy consumption, end-to-end delay overheads, and throughput are the basis for evaluating the performance of the proposed scheme. Table II indicates the parameters used in the simulation.

Table II: Simulation Parameter

Parameter	Value
Area of Network	1000 m × 500 m 1000 m × 1000 m 2000 m × 1000 m
Size of packet	1024 bytes
Traffic protocol	CBR
Data transmission protocol	UDP
Initial Energy of nodes	10, 50, 100 joules
Random seed	42
Learning rate for RL	0.001
Training episodes for RL	100
Communication rounds for FL	50

Performance Efficiency of Proposed Scheme:

Fig. 5 and Table III give the initial energy vs energy consumption analysis. The proposed framework is compared to all 6 other techniques Stability-Aware Cluster-Based Routing (SACR) [38], Distributed Energy Efficient Clustering (DEDC) [39], Cluster-Based Multihop Medium Access Control Protocol (RMAC-M) [40], Energy Preservation And Network Critics Based Channel Scheduling (EPNCS) [41], k-hop-Spatial-Awareness Convolution Block (K-SACB) [42], and Energy Aware Q-Learning AODV (EAQ-AODV) [43]) across various energy levels. The suggested technique demonstrates lower energy consumption across different energy levels, such as 10 J, 50 J, and 100 J, where the consumed energy is 1 J, 1.34 J, and 1.9 J, respectively, which are significantly lower than those of the compared methods. The improved proposed model performance can be attributed to the integration of spiking reinforcement learning and attention pyramid mechanisms, which intelligently optimize routing decisions, enhance feature prioritization, and minimize redundant data transmissions. Consequently, the framework reduces processing overhead and communication bottlenecks, thereby ensuring better energy efficiency and scalability for large-scale WSN and IoT environments.

Table III: Initial Energy vs Energy Consumption Performance Values

Algorithm	SACR	DEDC	RMAC-M	EPNCS	K-SACB	EAQ-AODV	PROPOSED
10 J	2.5	1.8	1.8	1.7	1.8	1.75	1
50 J	3	2	2.05	2	2.05	1.8	1.34
100 J	4.3	2.4	2.2	2.25	2.15	2.05	1.9

Table IV: Network Area vs Energy Consumption Performance Values

Algorithm	SACR	DEDC	RMAC-M	EPNCS	K-SACB	EAQ-AODV	PROPOSED
1000 × 500 m	4.5	4	3.9	3.6	3.3	3.1	2.8
1000 × 1000 m	4.8	4.3	4.1	4	3.5	3.3	3.2
2000 × 1000 m	5.6	4.9	4.25	4.05	3.8	3.5	3.4

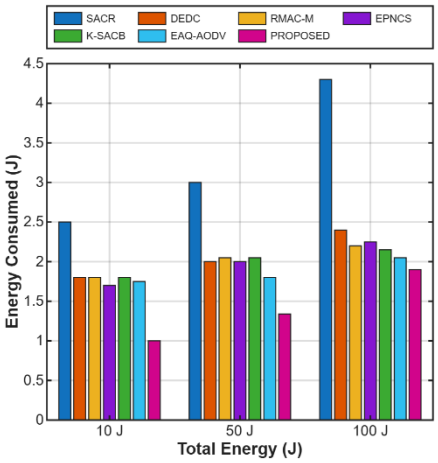

Fig. 5. Initial energy vs energy consumption performance.

Fig. 6 compares the energy consumed (J) in different network areas, highlighting that the proposed MARPB-DASLFRL framework is the most energy-efficient method. The proposed model always runs with the least energy consumption in 1000 × 500, 1000 × 1000 and 2000 × 1000 scenarios, using up energy of 2.8 J and only going up to about 3.4 J at the largest area. On the other hand, methods that are currently in use such as SACR consume energy at a much higher rate and at the peak, it is above 5.5 J, as shown in Table IV. This indicates that the proposed framework is not only capable of handling large-scale sensor networks but also it is efficient in the use of power.

Fig. 7 provides the initial energy vs delay performance for CR-WSNs. The proposed framework achieves consistently the lowest delay and largely outperforms all existing methods in this characteristic. Models such as SACR and DEDC have a very high delay that even peaks close to 0.06 ms, as depicted in Table V. On the other hand, the proposed model has maintained a very low delay with only minor fluctuations, being between 0.015 ms and 0.02 ms. Furthermore, this delay steadiness along the different energy levels demonstrates that the proposed framework is extremely well-tuned for rapid data transmission. Hence, it will be able to provide efficient performance irrespective of the network total energy capacity.

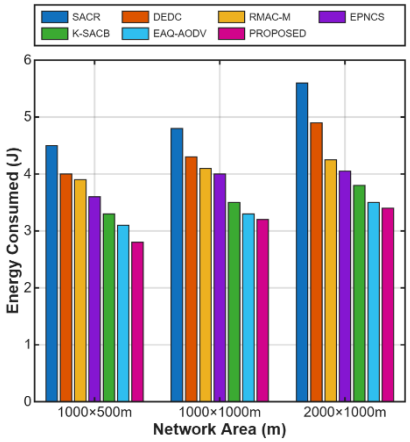

Fig. 6. Network area vs energy consumption performance.

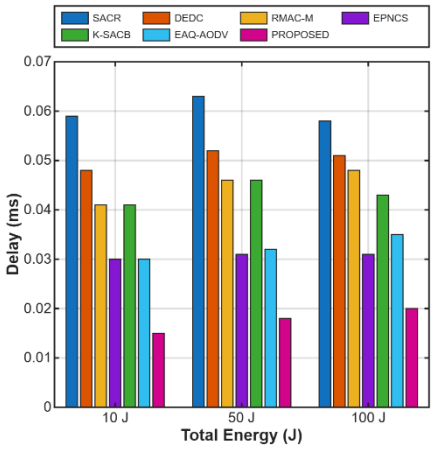

Fig. 7. Initial energy vs delay performance.

Fig. 8 displays the delay (ms) of various network areas, demonstrating that the proposed MARPB-DASLFRL framework is the one that allows the fastest data transmission. At the smallest area (1000 × 500), the proposed model records a delay of about 0.32 ms, which is significantly lower than the 0.65 ms of the SACR method. The proposed framework still delivers the best performance at the expanded network area (2000 × 1000) with a delay of only 0.42 ms while other methods such as SACR and DEDC reached 0.8 ms and 0.95 ms respectively. This indicates that the proposed framework is very efficient in reducing latency even in large-scale, optimally distributed settings. Table VI presents the network area vs delay performance values.

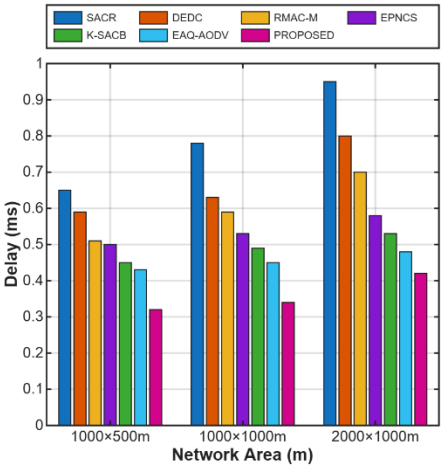

Fig. 8. Network area vs delay performance.

Fig. 9 denotes the comparison of the overhead (%) for different models at different total energy levels (10 J, 50 J, and 100 J). The proposed MARPB-DASLFRL framework always has the least overhead, which does not exceed 1.3% to 1.5%. Meanwhile, the SACR model represents the highest overhead of 2.5%, which is quite consistent, as indicated in Table VII. Hence, the proposed solution can be considered very effective in handling network control traffic internally, so that most of the energy is utilized for data transmission rather than protocol maintenance.

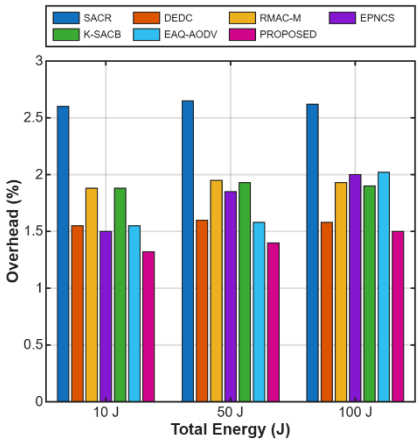

Fig. 9. Initial energy vs overhead performance.

Fig. 10 represents the network area vs overhead performance in the CR-WSN for effective energy efficient processing. Across all tested network dimensions (1000 × 500, 1000 × 1000 and 2000 × 1000), the proposed scheme preserves overhead at the lowest level all the time (3.2%, 4.5%). Existing methods such as SACR and DEDC show considerably higher overheads, with their high level of 8% and 7.5% respectively in the largest network area. The simple reduction in control traffic is a clear demonstration of how the proposed framework adjusts communication pathways and avoids unnecessary data processing; thus, it can be considered very efficient in utilizing scarce resources. Table VIII gives the network area vs overhead performance values.

TABLE V: INITIAL ENERGY VS DELAY PERFORMANCE VALUES

Algorithm	SACR	DEDC	RMAC-M	EPNCS	K-SACB	EAQ-AODV	PROPOSED
10 J	0.059	0.048	0.041	0.03	0.041	0.03	0.015
50 J	0.063	0.052	0.046	0.031	0.046	0.032	0.018
100 J	0.058	0.051	0.048	0.031	0.043	0.035	0.02

TABLE VI: NETWORK AREA VS DELAY PERFORMANCE VALUES

Algorithm	SACR	DEDC	RMAC-M	EPNCS	K-SACB	EAQ-AODV	PROPOSED
1000 × 500 m	0.65	0.59	0.51	0.5	0.45	0.43	0.32
1000 × 1000 m	0.78	0.63	0.59	0.53	0.49	0.45	0.34
2000 × 1000 m	0.95	0.8	0.7	0.58	0.53	0.48	0.42

TABLE VII: INITIAL ENERGY VS OVERHEAD PERFORMANCE VALUES

Algorithm	SACR	DEDC	RMAC-M	EPNCS	K-SACB	EAQ-AODV	PROPOSED
10 J	2.6	1.55	1.88	1.5	1.88	1.55	1.32
50 J	2.65	1.6	1.95	1.85	1.93	1.58	1.4
100 J	2.62	1.58	1.93	2	1.9	2.02	1.5

TABLE VIII: NETWORK AREA VS OVERHEAD PERFORMANCE VALUES

Algorithm	SACR	DEDC	RMAC-M	EPNCS	K-SACB	EAQ-AODV	PROPOSED
1000 × 500 m	7.6	7	6.5	5.6	5	4.8	3.2
1000 × 1000 m	7.8	7.3	7.5	5.5	6	5.1	4.5
2000 × 1000 m	8	7.5	7	6.5	6	5.4	5

TABLE IX: INITIAL ENERGY VS THROUGHPUT PERFORMANCE VALUES

Algorithm	SACR	DEDC	RMAC-M	EPNCS	K-SACB	EAQ-AODV	PROPOSED
10 J	240	285	245	288	225	310	340
50 J	245	288	248	280	230	312	355
100 J	242	280	240	282	228	315	365

Fig. 11 indicates the initial energy vs throughput performance analysis in the CR-WSNs for efficient transmission of data. The proposed framework always attained the best throughput measurements across all the test cases, and it even outmatches the existing methods by high range. At 10 J, it hits the mark of about 340 Kbps, and when the energy level is further raised to 100 J, its performance escalates to nearly 365 Kbps. Conversely, the second-best model, EAQ-AODV, levels off at around 315 Kbps, while models such as K-SACB stay under 240 Kbps. This is a clear proof of the proposed scheme's enhanced capability to utilize energy for data transmission more efficiently. Table IX mentions the initial energy vs throughput performance values.

Fig. 12 provides the throughput (Kbps) in different network areas-based on which the proposed framework is shown to deliver the best performance. The proposed model is the one keeping the highest throughput level, almost constant at around 600 Kbps. It is far ahead compared to the other methods such as SACR, which is battling at roughly 450 Kbps. Consistent output for the 1000 × 500, 1000 × 1000 and 2000 × 1000 areas confirm the model's excellent scalability and the ability to trust it even in very large network environments. Table X represents the network area vs throughput performance values.

The proposed scheme also shows improved

performance in the presence of malicious nodes owing to the R-PBFT mechanism. The approach identifies and isolate malicious nodes and minimizes packet loss and routing disruptions. Fig. 13 shows the analysis of throughput (Kbps) under varying percentage of malicious nodes for all the approaches. The proposed model shows highest throughput level above 300 Kbps even when half of the nodes are deemed as malicious. Table XI shows the malicious nodes percentage vs throughput values.

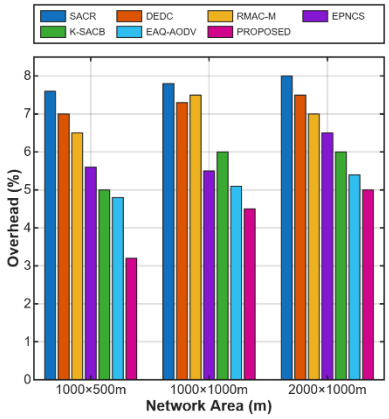

Fig. 10. Network area vs overhead performance.

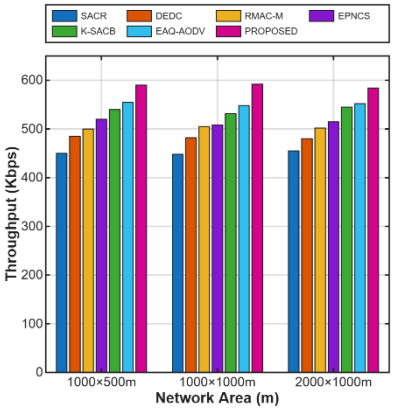

Fig. 12. Network area vs throughput performance.

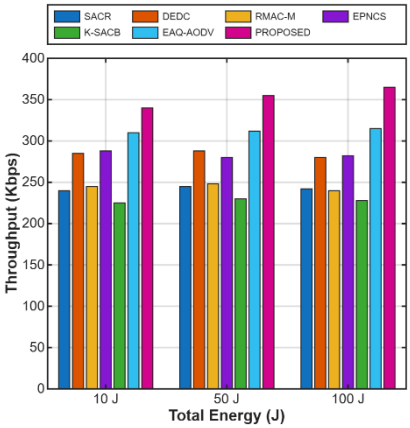

Fig. 11. Initial energy vs throughput performance

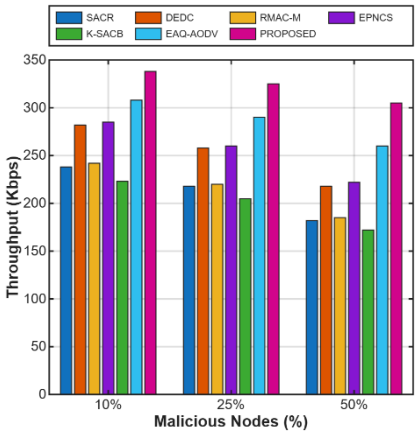

Fig. 13. Percentage of malicious node vs throughput performance.

TABLE X: NETWORK AREA VS THROUGHPUT PERFORMANCE VALUES

Algorithm	SACR	DEDC	RMAC-M	EPNCS	K-SACB	EAQ-AODV	PROPOSED
1000 × 500 m	450	485	500	520	540	555	590
1000 × 1000 m	448	482	505	508	532	548	592
2000 × 1000 m	455	480	502	515	545	552	584

TABLE XI: MALICIOUS NODES PERCENTAGE VS THROUGHPUT PERFORMANCE VALUES

Algorithm	SACR	DEDC	RMAC-M	EPNCS	K-SACB	EAQ-AODV	PROPOSED
10%	238	282	242	285	223	308	338
25%	218	258	220	260	205	290	325
50%	182	218	185	222	172	260	305

V. CONCLUSION AND FUTURE WORK

This study aims to present a framework for a cognitive wireless sensor network that combines fuzzy clustering, MO-BEFA-based optimal cluster head selection and MARPB-DASLFRL multi-hop routing in a secure and energy-efficient manner. Cognitive radio improves channel utilization by enabling dynamic spectrum availability and interference avoidance. The MO-BEFA optimization is used to analyze residual energy, node centrality, link quality, and spectrum accessibility to guarantee the strongest nodes are chosen as CHs. This lowers the extent of node re-clustering and balances energy usage. Nodes may efficiently detect available channels according to the initial frequency sensing, preventing interference among primary users and optimizing communication quality in situations involving overloaded spectrum. Decentralized training of optimal forwarding strategies is made easier through the MARPB-DASLFRL system. CHs can efficiently manage

complicated network states owing to the deep maxout attention pyramid spiking network. A Practical Byzantine Fault Tolerance (PBFT) blockchain centered around reputation is used to promise trust management. A highly secure network is produced by successfully validating routing choices and model changes, defeating malicious activity, model poisoning, and byzantine attacks. Federated Learning (FL) is used to reduce communication overhead and protect data privacy by sharing only the model updates through the global aggregator instead of raw data. The suggested cognitive WSN structure largely improves energy efficiency, spectrum utilization, routing dependability, and security, according to simulation results. The findings demonstrate that a scalable and dependable solution for dynamical cognitive sensor settings may be achieved by combining decentralized trust mechanisms, reinforcement learning and optimization. Future enhancements will need mobility-aware clustering, dynamic CH reselection, and reinforcement learning-based prediction for multi-hop routing to handle mobile sensor nodes, mobile sinks, and Unmanned Aerial Vehicle (UAV)-assisted collection of data.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Anu Maria Joykutty was involved in the conceptualization, design of the methodology, data analysis, conducting the research and writing the draft manuscript. B. Baranidharan contributed to the literature survey, review of manuscript and supervision of the research. All authors have approved the final version.

REFERENCES

[1] F. D. Zemo and S. Bakkali, "A survey on security threats in cognitive radio networks based on cooperative spectrum sensing," *International Journal of Communication Networks and Distributed Systems*, vol. 30, no. 4, pp. 433–466, 2024.

[2] H. Farhi, and A. Messai, "Toward smart and flexible spectrum usage: Application of horizontal model algorithm in cognitive radio systems," *Mathematical Modelling of Engineering Problems*, vol. 12, no. 7, 2025.

[3] A. Bazzi, R. Bomfin, M. Mezzavilla, S. Rangan, T. Rappaport, and M. Chafii, "Upper mid-band spectrum for 6G: Vision, opportunity and challenges," *IEEE Communications Magazine*, 2025.

[4] H. N. Vishwas, and T. K. Ramesh, "Recent trends in localization, routing, and security for wireless sensor networks," *IEEE Access*, 2025.

[5] S. Priya, S. Vappangi, A. K. Mishra, S. E. Mathe, A. A. Nasir, and N. Gupta, "Survey of cooperative noma for beyond 5g: state-of-the-art, applications and research directions," *Authorea Preprints*, 2025.

[6] R. V. B. Carvalho, H. A. Dinarte, R. C. Almeida Jr, and C. J. A. Bastos-Filho, "Systematic literature review of the use of computational intelligence in the routing and spectrum assignment problem in elastic optical networks," *Journal of Microwaves, Optoelectronics and Electromagnetic Applications*, vol. 24, no. 1, e2025289308, 2025.

[7] M. Zhang, X. Zhu, B. Zhang, S. Wang, and H. Sun, "A cross-layer performance evaluation system for spectrum sensing and allocation strategies in CR-WSN," *IEEE Sensors Journal*, vol. 24, no. 9, pp. 15355–15366, 2024.

[8] B. Sabir, S. Yang, D. Nguyen, N. Wu, A. Abuadbba, H. Suzuki, S. Lai, W. Ni, M. Ding, and S. Nepal, "Systematic literature review of ai-enabled spectrum management in 6g and future networks," arXiv preprint, arXiv:2407.10981, 2024.

[9] J. Wang, B. Wang, D. Tu, and Y. Li, "Green communications for wireless multimedia: energy-efficient solutions and sustainability perspective," in *Proc. International Conference on Information Processing and Network Provisioning*, Springer Nature Singapore, Singapore, 2024, pp. 258–272.

[10] S. Tong, and J. Peng, "Providing an optimal method for clustering in wireless sensor networks based on the Q-leach protocol," *Multiscale and Multidisciplinary Modeling, Experiments and Design*, vol. 8, no. 6, 294, 2025.

[11] M. Haris, and H. Nam, "Enhancing energy efficiency in iot-wsns through optimized PSO cluster head selection," *IEEE Access*, 2025.

[12] H. Alsuwat, and E. Alsuwat, "Energy-aware and efficient cluster head selection and routing in wireless sensor networks using improved artificial bee colony algorithm," *Peer-to-Peer Networking and Applications*, vol. 18, no. 2, 65, 2025.

[13] T. Mazumder, B. V. R. Reddy, and A. Payal, "Energy based multi objective golden jackal optimization for cluster-based routing in wireless sensor network," *Soft Computing*, vol. 28, no. 20, pp. 11927–11943, 2024.

[14] I. Abdulrahman, G. T. Ayodele, G. E. Egbedion, J. Alebiosu, E. J. Somtochukwu, and O. E. Akinbolajo, "Securing internet of things (IoT) ecosystems: addressing scalability, authentication, and privacy challenges," *World Journal of Advanced Research and Reviews*, vol. 26, pp. 523–534, 2025.

[15] H. V. Shah, "Secure and privacy-aware solutions for sustainable IoT," in *Design and Analysis of Green and Sustainable IoT Technologies for Future Wireless Communications*, Academic Press, 2026, pp. 95–107.

[16] M. Vamsikrishna, Y. Godiyal, A. K. Pandey, U. K. Lilhore, S. Simaiya, S. M. A. Aldossary, S. A. Hussien, E. S. Ghith, L. G. Tekeste, and Y. K. Sharma, "A modified chaos-driven enhanced cryptographic framework for lightweight and robust iot security in resource-constrained devices," *International Journal of Computational Intelligence Systems*, 2026.

[17] D. S. B. Naik, and V. Dondeti, "Trust-based secure federated learning framework to mitigate internal attacks for intelligent vehicular networks," *Peer-to-Peer Networking and Applications*, vol. 18, no. 2, 10, 2025.

[18] Y. He, G. Han, S. Zhu, J. Jiang, Y. Ma, and T. Zhang, "Trust management based on attention-weighted federated deep reinforcement learning for underwater acoustic sensor networks," *IEEE Transactions on Mobile Computing*, 2026.

[19] X. Zhang, Q. Tian, X. Xin, Y. Pan, H. Yao, F. Wang, Z. Dong *et al.*, "Asynchronous cooperative multi-agent deep reinforcement learning for joint rmsa and spectrum defragmentation in optical fiber communication networks," *Journal of Optical Communications and Networking*, vol. 18, no. 3, pp. 180–194, 2026.

[20] B. H. Taher, J. Luo, F. Ghrabat, and Y. Qiao, "Edgeplus: A multi-agent reinforcement learning framework for dynamic task allocation in 6G edge computing," *IEEE Internet of Things Journal*, 2026.

[21] H. Farhi, A. Messai, and T. Berghout, "Enhancing spectrum utilization in cognitive radio networks using reinforcement learning with snake optimizer: a meta-heuristic approach," *Electronics*, vol. 14, no. 13, 2525, 2025.

[22] Y. Kamatham, and D. Sunehra, "A novel node grade factor based multi-path routing (NGFMR) approach for improved Qos in cognitive wireless sensor networks," *Results in Engineering*, vol. 26, 105447, 2025.

[23] R. Joon, P. Tomar, G. Kumar, B. Balusamy, and A. Nayyar, "Unequal clustering energy hole avoidance (UCEHA) algorithm in cognitive radio wireless sensor networks (CRWSNS)," *Wireless Networks*, vol. 31, no. 1, pp. 735–757, 2025.

[24] P. G. Chilveri, and M. A. Dudhedia, "Game theory with tuna-sandcat optimisation-based robust and energy efficient spectrum sensing against malicious users in cognitive radio network," *International Journal of Wireless and Mobile Computing*, 2026.

[25] S. Regilan, L. K. Hema, D. Kadhiravan, and J. Jenitha, "Namib beetle algorithm based clustering and jump gain integral recurrent neural network for efficient energy in IoT networks," *SN Computer Science*, vol. 7, no. 1, 21, 2026.

[26] J. Alotaibi, "Energy-efficient trust management for secure IoT devices in information-centric wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 19, no. 2, 43, 2026.
[27] V. Jyothi, and M. V. Subramanyam, "An enhanced routing technique to improve the network lifetime of cognitive sensor network," *Wireless Personal Communications*, vol. 127, no. 2, pp. 1241–1264, 2022.
[28] A. M. K. Abdulzahra, and A. K. M. Al-Qurabat, "A clustering approach based on fuzzy c-means in wireless sensor networks for IoT applications," *Karbala International Journal of Modern Science*, vol. 8, no. 4, pp. 579-595, 2022.
[29] A. Rodan, A. Al-Tamimi, L. Al-Alnemer, S. Mirjalili, and P. Tiňo, "Enzyme action optimizer: A novel bio-inspired optimization algorithm: A. Rodan *et al.*," *The Journal of Supercomputing*, vol. 81, no. 5, 686, 2025.
[30] S. Aathilakshmi, S. Balasubramaniam, T. A. Sivakumar, and C. V. Lakshmi, "Min–max filtering and exponential fossa optimization algorithm–based parallel convolutional neural network for heart disease detection," *International Journal of Intelligent Systems*, 2025.
[31] H. Wang, Y. Liao, Y. Li, Y. Fang, S. Ni, Y. Luo, and B. Jiang, "BDR-NET: bhattacharyya distance-based distribution metric modeling for rotating object detection in remote sensing," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–12, 2022.
[32] M. S. Hema, V. C. Sharmila, V. Pandimurugan, R. Rajendran, and D. Kirubha, "Gastro-intestinal disease detection and classification using taylor spotted hyena optimization algorithm enabled deep maxout networks," *Biomedical Signal Processing and Control*, vol. 116, 109523, 2026.
[33] D. Li, U. A. Bhatti, M. Huang, L. Bruzzone, and J. Li, "Hypyramamba: A pyramid spectral attention and mamba-based architecture for robust hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 64, pp. 1–16, 2026.
[34] C. Wang, X. Dong, Z. Ji, M. Xiao, J. Jiang, X. Liu, Y. Huan, and S. Wu, "Model-agnostic linear-memory online learning in spiking neural networks," *Nature Communications*, 2026.
[35] S. Zhang, J. Wang, R. Zhou *et al.*, "MEMRL: self-evolving agents via runtime reinforcement learning on episodic memory," arXiv preprint, arXiv:2601.03192, 2026.
[36] N. Koutsoubis, A. Waqas, Y. Yilmaz, R. P. Ramachandran, M. B. Schabath, and G. Rasool, "Privacy-preserving federated learning and uncertainty quantification in medical imaging," *Radiology: Artificial Intelligence*, vol. 7, no. 4, e240637, 2025.
[37] E. Lee, N. Kim, C. Han, N. Shin, and I-G. Lee, "rpbft: reliable practical byzantine fault tolerance mechanism for faulty distributed networks," *IEEE Transactions on Big Data*, 2025.
[38] M. Zheng, C. Wang, M. Song, W. Liang, and H. Yu, "SACR: A stability-aware cluster-based routing protocol for cognitive radio sensor networks," *IEEE Sensors Journal*, vol. 21, no. 15, pp. 17350–17359, 2021.
[39] S. Gopikrishnan, P. Priakanth, and G. Srivastava, "DEDC: Sustainable data communication for cognitive radio sensors in the internet of things," *Sustainable Computing: Informatics and Systems*, vol. 29, 100471, 2021.
[40] A. Obaid, X. Fernando, and M. Jaseemuddin, "A mobility-aware cluster-based mac protocol for radio-frequency energy harvesting cognitive wireless sensor networks," *IET Wireless Sensor Systems*, vol. 11, no. 5, pp. 206–218, 2021.
[41] V. Gatate and J. Agarkhed, "Energy preservation and network critic based channel scheduling (EPNCS) in cognitive radio sensor networks," *International Journal of Information Technology*, vol. 13, no. 1, pp. 69–81, 2021.
[42] R. Prajapat, R. N. Yadav, and R. Misra, "Energy-efficient k-hop clustering in cognitive radio sensor network for internet of things," *IEEE Internet of Things Journal*, vol. 8, no. 17, pp. 13593–13607, 2021.
[43] R. Joon, and P. Tomar, "Energy aware q-learning AODV (EAQ-AODV) routing for cognitive radio sensor networks," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 9, pp. 6989–7000, 2022.

Copyright © 2026 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).



Anu Maria Joykutty has completed her master of technology in network engineering from Rajagiri School of Engineering and Technology, Kochi, India and is currently enrolled as a research scholar in SRM Institute of Science and Technology, Chennai, India. She is also working as an assistant professor in the Department of CSE, Rajagiri School of Engineering and Technology and has more than 8 years of academic experience and 11 publications to her credit. Her areas of interest include machine learning, wireless sensor networks and cognitive radio.



B. Baranidharan has completed his master of technology in computer science and engineering from SRM Institute of Science and Technology, Chennai, India, and Ph.D. in wireless sensor networks (specialization) from SASTRA Deemed University, Thanjavur, India. Currently, he is working as a professor in the Department of Computing Technologies, SRM Institute of Science and Technology. He has more than 17 years of academic experience and has published 80 papers in various International Journals and Conferences. Earlier his research involved designing new clustering architecture for wireless sensor networks and internet of things using various computational techniques. Of late he has spanned his research interests to the fields of artificial intelligence, machine learning, deep learning, and the internet of things.