



# 6LoWPAN in Wireless Sensor Network with IoT in 5G Technology for Network Secure Routing and Energy Efficiency

**Dr. Vishal Goar**

Associate Professor,

Govt. Engineering College Bikaner (Rajasthan)

<https://orcid.org/0000-0002-2267-6912>

## Abstract:

Today, interconnection and routing protocols must discover the best solution for secure data transformation with a variety of smart devices due to the growing influence of information technology, such as Internet of Things (IoT), in human life. In order to handle routing concerns with regard to new interconnection approaches like the 6LoWPAN protocol, it is required to offer an improved solution. This research propose novel technique in 6LoWPAN network secure routing and energy efficiency (EE) for WSN in IoT application based on 5G technology. Here the energy optimization has been carried out using clustered channel aware least square support vector machine (Cl\_CHLSSVM). Then the secure routing has been carried out using fuzzy based Routing Protocol for low-power and Lossy Networks with kernel-particle swarm optimization (Fuz\_RPL\_KPSO). To serve needs of IoT applications, proposed method is cognizant of both node priorities as well as application priorities. Applications' sending rate allocation is modeled as a constrained optimization issue. Pxpperimental analysis is carried out in terms of throughput of 96%, weighted fairness index of 77%, end-to-end delay of 59%, energy consumption of 86%, and buffer dropped packets of 51%.

**Keywords:** 6LoWPAN, secure routing, energy efficiency, wireless sensor networks, IoT, 5G technology.

## 1. Introduction:

The IPv6 over low power WPAN (6LoWPAN) protocol, which is based on IEEE 802.15.4, has made it possible for WSNs to transform PANs into LoWPANs with IPv6 [1]. The 6LoWPANs enable communication between WSN as well as traditional internet by having low energy consumption, cheap costs, low data rates, and flexible deployment devices. Since 6LoWPAN first appeared, its expanding popularity has sparked the development of numerous novel communication proposals, which have been executed by TinyOS and verified in TOSSIM [2] and a working WSN testbed. 6LoWPAN was limited in all of its numerous uses to interacting with environment or responding to certain events. In order to balance traffic load as well as prevent energy-based network partition, a feasible and effective routing protocol should be developed. It should be able to operate on both static as well as mobile networks, be self-managed, and be memory-efficient as it tends to reduce energy required for routing. 6LoWPAN network is thought to be a critical network as well as significant component of IoT, where 6LoWPAN nodes will make up majority of IoT devices [3]. Although it was initially created for wired networks, 6LoWPAN is utilized

for complete integration of WSNs with Internet where sensor nodes (SNs) execute Internet Protocol (IP) layer. However, because of the constrained energy and buffer resources, the TCP/IP method implementation in 6LoWPAN as well as WSNs faces numerous challenges. UDP (user datagram protocol) does not include a congestion control method, while TCP (transmission control protocol) needs connection establishment as well as termination before to and following data transmission. TCP as well as UDP are ineffective for 6LoWPAN and WSN networks, respectively [4]. Congestion, which increases energy consumption, decreases throughput, and causes packet loss is thus one of the key problems with WSNs and 6LoWPAN networks. The applications for 6LoWPAN networks are expanding as more WSNs are connected to Internet via this technology, including those in the industrial, automation, healthcare, military, environmental, and logistics sectors. According to the data delivery technique, applications may generally be divided into four groups [5]. Network traffic in event-based systems is normally minimal and spikes in reaction to a recognized event. Congestion is brought on by these high data rate packets, hence congestion control is crucial to take into account. In applications that use queries, the sink node sends a query to SNs, and sensor nodes answer by delivering





packets. Last but not least, the aforementioned three types are integrated into hybrid applications, in which SNs simultaneously emit packets on a regular basis, in reaction to events, and in response to sink queries. Future WSN integration with the Internet to create the IoT will make this kind of application more prevalent. Applications should be effectively and robustly isolated from the unreliability of WSNs using transport protocols, which should also be scalable. While UDP has been incorporated into the 6LoWPAN stack, which enables low-power and limited computing devices to participate in the IoT, approaches like TCP are notorious for not being acceptable in sensor network contexts. UDP is unreliable, and it does not guarantee retransmission in the event that a packet is dropped at an intermediate node, which may not be acceptable for applications with low or no requirements for packet drops. Packet forwarding will not be efficient in the sense that resources are utilized at intermediate nodes and packets are not reaching their destination, in addition to not being acceptable for particular applications [6].

## 2. Literature Review:

This section serves as an example of a quick literature assessment of current routing algorithms that use metaheuristics. For instance, writers in [7] provided a network function virtualization-based assignment design for intelligent cities based on blockchain and SDN. [8] contains a number of surveys on energy use, energy gaps, and attacks on RPL and LLN systems. To reduce energy consumption brought on by control overhead, a cast-off energy-based transmission mechanism based on glowworm swarms has been proposed [9]. A modified particle swarm optimization (MPSO)-based least-square support vector machine (LS-SVM) is created. An LS-SVM-based MPSO's prediction method is first built, and inertial weight of the MPSO is modified to enable faster iterations. Second, using the MPSO's optimized parameters, the predictive simulation was run and verified, and MPSO and PSO projected values were compared [10]. A durable clustering routing mechanism for WSNs is presented in this paper. The method uses Locust Search (LS-II) strategy to determine the best cluster heads and determine the number of cluster heads. Other sensor components are assigned to CHs nearest to them once cluster heads have been discovered [11]. Method for an EE clustering method for data collection and transfer is created using Optimal Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol. To choose CH, a new, optimized threshold function is applied. The cluster

headcount is higher but the power consumption is higher with LEACH, a hierarchical routing technique that selects CH nodes at random in a loop. The Centralised LEACH Protocol is best for increasing energy per unit node as well as asPDR with less energy utilization [12]. WSNs are intended for specialized applications where battery power is a critical concern, such as monitoring or tracking in both indoor as well as outdoor settings. To address this issue, various routing techniques have been developed. Performance enhancement is also offered using a sub-cluster LEACH-derived method. Simulation results [13] showed that the Sub-LEACH with LMNN outperformed its rivals in terms of EE. In order to avoid network resource competitions in 6LoWPAN, [14] and [15] modified the AODV protocol. However, because they are still based on broadcast routing lookup mechanisms, network consumptions are enormous and are not compatible with 6LoWPAN low power restrictions. However, it has drawbacks in practical settings. Work [16] presented a lightweight configuration protocol for static 6LoWPAN networks as well as assumed network is flat as well as has a decent connectivity. For 6LoWPAN, work [17] suggested a hierarchical routing system. The address allocation will, however, worsen the routing algorithm's performance as the sensor network grows. The distributed address allocation (DAA) [18] protocol can speed up network by allocating addresses while facilitating communication between network's key nodes. However, in actual use, there are more instances of complicated network architecture as well as unevenly distributed nodes, and it loses value as the network grows larger and more dynamic. According to the activities surrounding this protocol within the 6LoWPAN, the 6LoWPAN ad hoc on-demand distance vector routing (LOAD) [19] is one of the most promising on-demand routing protocols and is based on AODV [20].

## 3. System design:

This section discuss novel technique in 6LoWPAN network secure routing and EE for WSNs in IoT application based on 5G technology. Here the energy optimization has been carried out using clustered channel aware least square support vector machine (CI\_CHLSSVM). Then the secure routing has been carried out using fuzzy based Routing Protocol for low-power and Lossy Networks with kernel-particle swarm optimization (Fuz\_RPL\_KPSO).



### Clustered channel aware least square support vector machine (CI\_CHLSSVM) based energy optimization:

The cluster head and number of clusters do not need to be chosen by humans before the clustering phase begins. After deployment of WSN nodes, agglomerative HC algorithm is used to achieve an acceptable clustering to enhance data transmission method inside every cluster. As a result, during the WSN monitoring phase, data are collected from nodes and CHs as well as combined into data packages. The fact that CHs are responsible for transmitting these data packages to BS suggests that the CHs are more energy-hungry than the other nodes in the cluster. As demonstrated in figure 1, the clustering technique can be used to optimize data routing and ensure great energy efficiency.

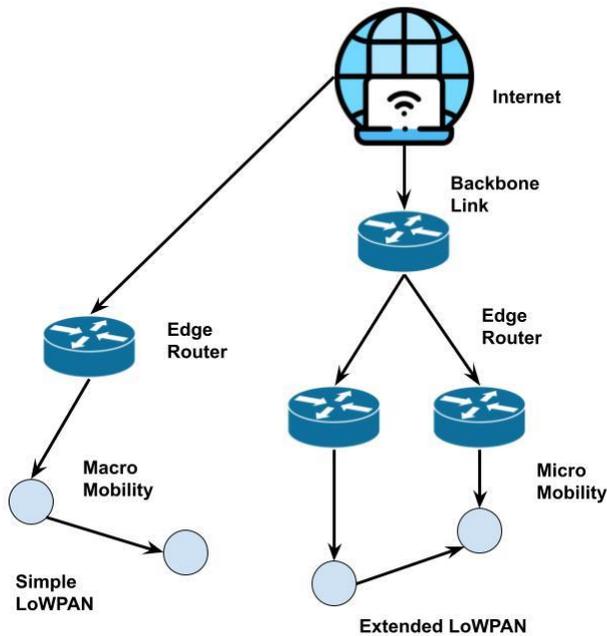


Figure-1 channel aware clustering

Consider M nodes are optimally placed throughout placement region, and that their a priori fixed and known coordinates. These M nodes are regarded as starting clusters in accordance with theory underlying aggregation hierarchical clustering method as shown in eq.(1):

$$C_i = \{x_i\}, i \in M \quad (1)$$

where  $C_i$  stands for  $i$ th cluster that developed. Each time a clustering iteration is performed, the maximum Euclidean distance between any two groups is determined and used as the clustering cost. Once needed number of clusters or termination condition is reached, two clusters with smallest

distance would join to form a new cluster. Assuming that clusters  $C_{M+a}$  and  $C_{M+b}$  include  $\{x_i, x_j\}$  and  $\{X_k, X_l\}$  respectively, after many clustering processes. The  $X_i$  and  $X_k$  nodes are the ones that are separated from the other nodes the most. The maximum distance  $D(C_{M+a}, C_{M+b})$  between these two clusters can be shown as follows using the HC algorithm's concept of the largest distance between clusters by eq.(2):

$$D(C_{M+a}, C_{M+b}) = D(X_i, X_k) \quad (2)$$

where the cluster labels  $M + a$  and  $M + b$  are represented. The longest distance  $D_{max}$  between the deployed nodes is chosen and modified in this study to act as the clustering termination threshold T by eq.(3).

$$T = \sigma D_{max} = \sigma \max \left\{ \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2} \right\}, i, j \in M, i \neq j \quad (3)$$

coordinates of  $i$ th and  $j$ th nodes are  $X_i = (X_i, Y_i)$ ,  $X_j = (X_j, Y_j)$ , and stands for the practical factor, which is described as ratio of distance between nodes within confidence distance. Here, everything described in great depth. First, through data transmission simulations in the real-world target region, confidence distance ( $D_c$ ) for trustworthy data transmission is determined in advance.

Then, the ratio of  $d_{ij} < D_c$  could be evaluated and recorded as, and distance between  $i$ th and  $j$ th nodes is evaluated and recorded as  $\sigma$ . As a result, the calculated proportion might be used to determine the threshold T in accordance with (3). Therefore, combining confidence distance with node topology could result in a node clustering that is more logical. The wireless channel propagation model is established using the free space and multipath attenuation methods, and the energy consumption  $E_{Tx}$  for delivering k bits of data can be summarised as (4):

$$E_{Tx}(k, d) = E_{Tx-elec}(k) - E_{Tx-amp}(k, d) = \begin{cases} E_{elec} * k + \epsilon_{fs} * k * d^2, & d < d_0 \\ E_{elec} * k + \epsilon_{mp} * k * d^4, & d \geq d_0 \end{cases} \quad (4)$$

where  $E_{Txelec}$  stands for the transmission energy and  $E_{Txamp}$  for the amplification energy needed for data transmission to distance d, and  $d_0$  stands for the distance threshold by eq.(5):

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \quad (5)$$

The energy needed to receive k-bit data can also be represented as (6):

$$E_{RR(k)}(k)E_{Rx-elec}(k) = E_{elec} * k \quad (6)$$

Data transmission and reception, creation and maintenance of routing structure, all contribute to CH node's energy consumption, which is significantly higher than that of other nodes in cluster. CH node must conduct energy harvesting and go to sleep after a full data transmission cycle in order to participate in following cycle and maintain uninterrupted WSN coverage. Based on cluster's remaining node energy and node location data, successor CH node is chosen. Assuming a cluster consists of Q nodes, BS is symbolised by B, total amount of data that needs to be transmitted to BS for this cluster in a cycle is kBs bits, distance between CH node and BS is dBs bits, qth node in cluster needs to transfer to CH node s in a cycle is kqs bits, and distance between q and s is dq.

The energy consumption EBs of CH nodes s in a single cycle consists of energy consumption ERx when receiving data from nodes which is given in eqn (7):

$$\begin{aligned} E_{Bs} &= E_{Rx}(k_{Bs}) + E_{Df}(k_{Bs}) + E_{Tx}(k_{Bs}, d_{Bs}) \\ &= E_{elec} * k_{Bs} + E_{DA} * k_{B_e} + E_{Tx-dec}(k_{Bs}, d) + E_{Tx-amp}(k_{B_2}, d) \end{aligned} \quad (7)$$

where EDA stands for the data fusion energy consumption constant. Assume that the cluster head node s receives kqs bits of data from node q in a single data transmission cycle by eq.(8):

$$\sum_{q=1}^Q k_{qs} = k_{Bs} \quad (8)$$

In order to send these data, the qth node used the following amount of energy, or Eqs: (9)

$$E_{qs} = E_{Tx-elec}(k_{qs}) + E_{Tx-amp}(k_{qs}, d), q \neq s \quad (9)$$

Based on locations and conditions of cluster's remaining nodes, suitable node would be chosen as new CH node. Assume that estimation represents the energy usage for various nodes during data delivery by eq.(10):

$$E_{estimation}(s) = E_{Bs} + \sum_{q=1}^Q E_{qs}, q \neq s \quad (10)$$

where the Eestimation(s) takes into account both the energy needed by nodes to transmit data to CH as well as energy needed by CH to transmit data to BS. Assume Erest can

represent remaining node energy, and the probability of being chosen as CH for qth is represented by symbol by eq.(11).

$$\rho(q) = \begin{cases} 1 - \frac{E_{satroum}(q)}{E_{res}(q)}, q \in G \\ 0, q \notin G \end{cases} \quad (11)$$

where G stands for group of nodes that have not been chosen for the current data transmission cycle. Successor cluster head's energy requirements for data delivery should be minimal, and energy that is still in successor CH should be adequate, so node with highest probability becomes successor CH. Energy collection of CH nodes must be more than the threshold EΔ if each one requires to execute at least Z times base station's data transmission tasks in one cycle, as illustrated ineq.(12)

$$E_{\Delta} \geq Z * (E_{Bs} + \sum_{q=1}^Q E_{qs}), q \neq s \quad (12)$$

where Eqs is the energy used when a non-cluster head node sends data to succeeding CH and EBs is energy used when node s is adopted as CH. Assume that ΔT represents time needed to finish gathering information for one cluster and EΔ represents needed for charging time of up to σE by

$$\sigma_E \leq (30\% * Q) * (Z * \Delta T) \quad (13)$$

**Algorithm for channel aware clustering:**

```

Input: Xi
1 Produce initial clusters: Ci = {Xi}, i ∈ M
2 Find node clusters adaptively based on T :
3 T = σ max { √((Xi - Xj)2 + (Yi - Yj)2), i, j ∈ M, i ≠ j
4 for i = M + a, ... do
5   EΔ ≥ Z * (EBs + ∑q=1Q Eqs), q ≠ s
6   ≤ (30% * Q) * (Z * ΔT)
7 for every data transmission do
8 energy consumption of sthCH:
9 EBs = Eelec * kBs + EDA * kBe + ETx-elec(kBs)
10 + ETx-amp(kBs, d)
11 energy consumption of q th node:
12 Eqs = ETx-elec(kqs) + ETx-amp(kqs, d), q ≠ s
13 if Erost < Eestimation(s) then
14   ρ(q) = { 1 - Eestimation(q) / Erest(q), q ∈ G
15           0, q ∉ G } -Charging the exhausted
node

```

Calculating joining weight value of CH is done by SN that receives joining message. SN joins CH as a CM if highest CH joining weight value is attained. Figure 1 depicts how CHs are chosen.

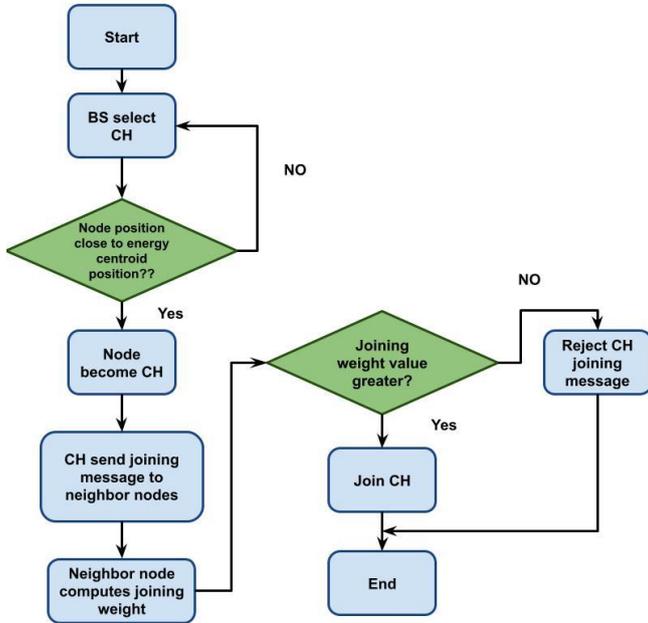


Figure 2: CH selection flow chart

As a result, the numerical representation of the test statistics of energy detection (TED) of signal received at  $SU_i$  is given by eq.(14):

$$\begin{aligned}
 & S_{\mathcal{L}}[n] \leq \tau_e; \\
 & 0 < \mathcal{M}_{sg(i)} \leq M_{sg} g_{sg(j)} \leq 1 \\
 & n > 1 \leq k \\
 & E_{\mathcal{L}}[n] \geq E_{\zeta} \\
 & \rho_e \leq \rho_e(\epsilon) < 1 \\
 & \tau_{\zeta} > 01 \leq \zeta \leq T \vee n \in \mathcal{PU}_a; \\
 & E_{\zeta} \geq 01 \leq \zeta \leq E \vee E[n] \in \mathcal{PU}_j; \\
 & \rho_e \geq 01 \leq e \leq E
 \end{aligned} \tag{14}$$

Number of samples each SU collects is dependent on sensing duration, and longer sensing delays within predefined time intervals result in greater primary signal recognition performance. Since it is assumed that primary user signal as well as noise are independent, following formulation of the binary hypothesis testing issue of received energy signals at SU is possible by eq.(15):

$$\mathcal{T}_{\text{EnD}}(SU_i)_{S_{\mathcal{L}}} = \sum_{i=1}^m |n_{o(i)}|^2 \in \mathcal{H}_0$$

$$\mathcal{T}_{\text{EnD}}(SU_i)_{S_{\mathcal{L}}} = \sum_{i=1}^m |C_{Q(i)} S_{\mathcal{L}(i)} + n_{o(i)}|^2 \in \mathcal{H}_1 \tag{15}$$

where  $m, n_{\text{noise}(i)}$  denotes the  $i$ th time slot's noise, channel gain between PU and SU, and time-bandwidth product. To determine if  $PU_j$  is present or not, the received energy at every  $SU_i$  is compared to predetermined detection threshold ( $\sigma$ ). Using a predetermined network threshold, each SU in a locally sensing mechanism determines if the  $i$ th channel is available ( $\tau_{\text{off}}$ ) or occupied ( $\tau_{\text{on}}$ ).

As a result, hypothesis at every secondary user states that PU is only active if received signal energy exceeds stated limit and idle when it is below defined threshold value, which may be expressed numerically aseq.(16):

$$\begin{aligned}
 \mathcal{T}_{\text{EnD}}(SU_i)_{S_{\mathcal{L}}} &= [\mathcal{H}_0 \in ES_{\mathcal{L}} > \sigma \mid \mathcal{H}_1 \in ES_{\mathcal{L}} < \sigma] \in \tau_s \\
 \mathcal{T}_{\text{EnD}}(SU_i)_{S_{\mathcal{L}}} &= \tau_{\text{on}} \mid \tau_{\text{off}} \in C_i \subseteq C_n \\
 \mathcal{H}_1(C_i) &\in C_n = 1(\tau_{\text{on}}) \\
 \mathcal{H}_0(C_i) &\in C_n = 0(\tau_{\text{off}}) \\
 \mathcal{P}_{\mathcal{A}}(C_i) &\in C_n \neq \tau_{\text{on}} \\
 C_i &\in \mathcal{H}_1 \mid \mathcal{H}_0 \leq 1 \\
 \mathcal{SU}_i(C_i) &\in C_n(\mathcal{R}_i) = \tau - \tau_s \\
 \mathcal{SU}_i \notin C_i(\mathcal{R}_i) &= \tau
 \end{aligned} \tag{16}$$

$$\mathcal{SU}_i(C_i)_{\text{busy}} \in C_n(\mathcal{R}_i) \neq \tau_s$$

The chance of detection while a  $PU_j$  is idle and probability of FAs when a  $PU_j$  is active are used to calculate how well each SU performs locally detecting spectrum, and the results can be stated numerically aseq.(17)

$$\begin{aligned}
 \rho_{\mathcal{D} \in \mathcal{T}} &= \rho(\mathcal{H}_1 \mid \mathcal{H}_1) \in \rho(ES_{\mathcal{L}} > \sigma \mid \mathcal{H}_1) \\
 \rho_{\mathcal{F} \in \mathcal{A}_3} &= \rho(\mathcal{H}_1 \mid \mathcal{H}_0) \in \rho(ES_{\mathcal{L}} > \sigma \mid \mathcal{H}_0)
 \end{aligned} \tag{17}$$

By expressing the classification problem as, a least squares variant of the SVM classifier can be created by eq.(18), (19).

$$\min_{w,b,e} y_3(w, b, e) = \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{k=1}^N e_k^2 \tag{18}$$

subject to equality constraints,

$$y_k [w^T \varphi(x_k) + b] = 1 - e_k, k = 1, \dots, N \tag{19}$$

One defines Lagrangian by eq.(20)

$$\mathcal{L}_3(w, b, e; \alpha) = y_3(w, b, e) - \sum_{k=1}^N \alpha_k \{y_k [w^T \varphi(x_k) + b] - 1 + e_k\} \tag{20}$$

Since there are no longer any equality restrictions resulting from the Kuhn-Tucker requirements, the Lagrange multipliers  $\alpha_k$  can now be either positive or negative. The following set of linear equations' solutions can be used to write the conditions for optimality by eq.(21):

$$\left[ \begin{array}{ccc|c} I & 0 & 0 & -Z^T \\ 0 & 0 & 0 & -Y^T \\ 0 & 0 & \gamma I & -I \\ \hline Z & Y & I & 0 \end{array} \right] \begin{bmatrix} w \\ b \\ e \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (21)$$

Where,  $\varepsilon Z = [\varphi(x_1)^T y_1; \dots; \varphi(x_N)^T y_N]$ ,  $Y = [y_1; \dots; y_N]$ ,  $\vec{1} = [1; \dots; 1]$ ,  $e = [e_1; \dots; e_N]$ ,  $\alpha = [\alpha_1; \dots; \alpha_N]$ . Solution is also represented by eq.(22),

$$\begin{bmatrix} 0 & -Y^T \\ Y & ZZ^T + \gamma^{-1}I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (22)$$

Mercer's condition is applied again to matrix  $\Omega = ZZ^T$  where by eq.(23)

$$\begin{aligned} \Omega_{kl} &= y_k y_l \varphi(x_k)^T \varphi(x_l) \\ &= y_k y_l \psi(x_k \cdot x_l). \end{aligned} \quad (23)$$

As a result, rather than using quadratic programming, the classifier (1) is discovered by solving the linear set of equations (20)–(21). The RBF kernel's parameters, such as  $\sigma$ , can be selected in the best possible way using (12). In contrast to (14) where the majority of the values are equal to zero, the support values  $k$  are proportional to the mistakes at the data points. Therefore, in the least squares scenario, it would be more accurate to refer to a support value spectrum.

### Fuzzy based Routing Protocol for low-power and Lossy Networks with kernel-particle swarm optimization (Fuz\_RPL\_KPSO):

The weighted sum model (WSM), a multi-attribute decision-making method, is applied to a single dimensional value set. When there are several competing ideas, decision-makers are utilised to resolve the issues. These issues might range from choosing which car or smartphone to buy to choosing best parent in network. One such method is average weighted sum method, which may be evaluated as follows when there are  $Y$  alternative alternatives and  $X$  specifications by eq.(24):

$$A_{WSM} = \sum_{j=1}^X W_j a_{ij}, \text{ for } i = 1, 2, 3, \dots, \gamma \quad (23)$$

ETX, BO, and RtMetric are considered as routing metrics. Each routing metric has a weight, which is represented by the weight vector  $W = \{w_j, j = 1, 2, 3\}$ . A decision matrix can be used to illustrate the multi-attribute decision-making process for the parent selection issue. As a result, the decision matrix  $M$  can be expressed aseq.(24):

$$M = \begin{bmatrix} Q_1(a_1) & Q_2(a_1) & Q_3(a_1) \\ Q_1(a_2) & Q_2(a_2) & Q_3(a_2) \\ \vdots & \vdots & \vdots \\ Q_1(a_n) & Q_2(a_n) & Q_3(a_n) \end{bmatrix} \quad (24)$$

For example, let there be three parents say by eq.(25),

$$P = \{P_1, P_2, P_3\} \quad (25)$$

and we have three routing metrics i.e by eq.(26).

$$V = \{V_{ETX}, V_{RtMetric}, V_{BO}\} \quad (26)$$

then  $M$  can be written aseq.(27):

$$M = \begin{bmatrix} Q_{ETX}(699) & Q_{RtMetric}(2320) & Q_{BO}(0.5) \\ 768 & 2048 & 0.625 \\ 640 & 1766 & 0.375 \end{bmatrix} \quad (27)$$

The decision matrix  $M$  is multiplied by the weight vector  $W$  to obtain the WSM value for each parent by eq.(28).

$$\begin{aligned} A_{WSM} &= M * W \\ A_{WSM} &= \begin{bmatrix} Q_1(a_1) & Q_2(a_1) & Q_3(a_1) \\ Q_1(a_2) & Q_2(a_2) & Q_3(a_2) \\ \vdots & \vdots & \vdots \\ Q_1(a_n) & Q_2(a_n) & Q_3(a_n) \end{bmatrix} * \begin{bmatrix} W_1 \\ W_2 \\ W_3 \end{bmatrix} \\ A_{WSM} &= \begin{bmatrix} Q_{ETX}(699) & Q_{RtMetric}(2320) & Q_{BO}(0.5) \\ 768 & 2048 & 0.625 \\ 640 & 1766 & 0.375 \end{bmatrix} \\ &= \begin{bmatrix} W_{ETX}(50) \\ W_{RtMetric}(20) \\ W_{BO}(4276) \end{bmatrix} \\ A_{WSM} &= \begin{bmatrix} R_{P1}(83488) \\ R_{P2}(82032.5) \\ R_{P3}(68923.5) \end{bmatrix} \end{aligned} \quad (28)$$

**Algorithm for RPL:**

Input: Maximum No. of nodes ( $N^{\max}$ ), Array for storing blacklisted nodes  $\chi \leftarrow \{0, \dots, N^{\max}\}$ , Array for storing node information ( $Y \leftarrow \{0, \dots, N^{\max}\}$ ), Current method clock time ( $\Phi_{\text{current}}$ ), DIS transmission interval ( $\beta, \delta, \Psi$ )

Step 1: Start

Step 2: Get Current method time

Step 3: Set the Source, Timestamp, DIS, and Elevel to 0 upon initialization.

Step 4: Verify that nodes on the blacklist are zero

Step 5: Remove the DIS message if message was received from the originating node.

Step 6: Obtain the system's current timestamp along with the source message sent by the sender.

Step 7: Add the nodes to the blacklist if message wasn't received from the source.

Step 8: If message is delivered, obtain most recent timestamp as well as preserve it.

Step 9: End

In the PSO method, the particles constantly alter their positions as they seek fresh data, and each particle is able to remember the best answer it has previously found. The First Algorithm creates a collection of random particles as well as then iteratively searches for best answer. Particles update themselves after each repetition by keeping track of two extreme values. One is individual extremum pBest, which is best solution determined by particles themselves. Second is global extremum gBest, best solution determined by entire particle swarm. Particles update their speed and position in accordance with the following two formulas after determining the aforementioned two extreme values by eq.(29).

$$v_i(t + 1) = wv_i(t) + c_1 \cdot \text{rand}_1 \cdot (pBest - p_i(t)) + c_2 \cdot \text{rand}_2 \cdot (gBest - p_i(t))$$

$$p_i(t + 1) = p_i(t) + v_i(t + 1) \quad (29)$$

Let  $(X = \{\vec{x}_1, \vec{x}_2, \dots, \vec{x}_n\})$  be a set of n unlabeled patterns in d-dimensional input space. The jth real valued feature of the ith pattern ( $i = 1, 2, \dots, n$ ) is represented by every element  $x_{i,j}$  in ith vector,  $x_i$ . Given such a set, partitioning clustering method seeks a partition  $C = \{C_1, C_2, \dots, C_k\}$  of k classes where degree of similarity between patterns within a cluster is maximised and the degree of difference between patterns within clusters is maximised. The following qualities ought to be preserved by the partitions by eq.(30):

$$C_i \neq \Phi \forall i \in \{1, 2, \dots, k\}$$

$$C_i \cap C_j = \Phi \forall i \neq j \text{ and } i, j \in \{1, 2, \dots, k\}$$

$$\cup_{i=1}^k C_i = P \quad (30)$$

Euclidean distance, which is calculated between any 2 d-dimensional patterns  $\vec{x}_i$  and  $\vec{x}_j$  and can be used to determine how similar two patterns are, is the method that is most frequently used to compare two patterns by eq.(31).

$$d(\vec{x}_i, \vec{x}_j) = \sqrt{\sum_{p=1}^d (x_{ip} - x_{jp})^2} = \|\vec{x}_i - \vec{x}_j\| \quad (31)$$

$$\varphi(\vec{x}_i) = [\varphi_1(\vec{x}_i), \varphi_2(\vec{x}_i), \dots, \varphi_H(\vec{x}_i)]^T \quad (32)$$

By applying mapping, a dot product  $\vec{x}_i^T \cdot \vec{x}_j$  is transformed into  $\varphi^T(\vec{x}_i) \cdot \varphi(\vec{x}_j)$ . Dot product  $\varphi^T(\vec{x}_i) \cdot \varphi(\vec{x}_j)$  in transformed space is estimated through kernel function  $K(\vec{x}_i, \vec{x}_j)$  in input space  $\mathcal{R}^d$ .

Let  $d = 2$  and  $H = 3$  and consider subsequent mapping:  $\varphi: \mathcal{B}^2 \rightarrow H = \mathcal{R}^3$ , and  $[x_{i,1}, x_{i,2}]^T \rightarrow [x_{i,1}^2, \sqrt{2} \cdot x_{i,1}, x_{i,2}, x_{i,2}^2]^T$  Now dot product in feature space  $H=3$  by eq.(33)

$$\begin{aligned} \varphi^T(\vec{x}_i) \cdot \varphi(\vec{x}_j) &= [x_{i,1}^2, \sqrt{2} \cdot x_{i,1}, x_{i,2}, x_{i,2}^2] \\ &\quad \cdot [x_{j,1}^2, \sqrt{2} \cdot x_{j,1}, x_{j,2}, x_{j,2}^2]^T \\ &= [x_{i,1} \cdot x_{j,1} + x_{i,2} \cdot x_{j,2}]^2 = [\vec{x}_i^T \cdot \vec{x}_j]^2 = K(\vec{x}_i, \vec{x}_j). \end{aligned} \quad (33)$$

Clearly, square of dot product of the vectors  $\vec{x}_i$  and  $\vec{x}_j$  in  $\mathcal{R}^d$  is the basic kernel function K. Consequently, the kernelized distance between the 2 patterns,  $\vec{x}_i$  and  $\vec{x}_j$ , is given by eq.(34)

$$\begin{aligned} \|\varphi(\vec{x}_i) - \varphi(\vec{x}_j)\|^2 &= (\varphi(\vec{x}_i) - \varphi(\vec{x}_j))^T (\varphi(\vec{x}_i) - \varphi(\vec{x}_j)) \\ &= \varphi^T(\vec{x}_i) \cdot \varphi(\vec{x}_i) - 2 \cdot \varphi^T(\vec{x}_i) \cdot \varphi(\vec{x}_j) \\ &\quad + \varphi^T(\vec{x}_j) \cdot \varphi(\vec{x}_j) = K(\vec{x}_i, \vec{x}_i) - 2 \cdot K(\vec{x}_i, \vec{x}_j) \\ &\quad + K(\vec{x}_j, \vec{x}_j) \end{aligned} \quad (34)$$

Due to its superior classification accuracy over linear as well as polynomial kernels on numerous test tasks, well-known Gaussian kernel—also known as RBF has been selected in the current context among different kernel functions utilised in literature. Gaussian Kernel is given by eq.(35)

$$K(\vec{x}_i, \vec{x}_j) = \exp\left(-\frac{\|\vec{x}_i - \vec{x}_j\|^2}{2\sigma^2}\right) \quad (36)$$

where  $r > 0$ . Clearly, for Gaussian kernel,  $K(\vec{x}_i, \vec{x}_i) = 1$  and thus relation reduces by eq.(37)

$$\|\varphi(\vec{x}_i) - \varphi(\vec{x}_j)\|^2 = 2 \cdot (1 - K(\vec{x}_i, \vec{x}_j)). \quad (37)$$

The centroid of a cluster is determined before the CS measure is applied by averaging data vectors associated with that cluster utilizing eq.(38),

$$\vec{m}_i = \frac{1}{N_i} \sum_{x_j \in C_i} \vec{x}_j \quad (38)$$

CS measure is described as  $d(\vec{x}_i, \vec{x}_j)$  where d is a distance metric between any two data points by eq.(39).

$$\begin{aligned} CS(k) &= \frac{\frac{1}{k} \sum_{i=1}^k \left[ \frac{1}{N_i} \sum_{\vec{x}_i \in C_i} \max_{\vec{x}_q \in C_i} \{d(\vec{x}_i, \vec{x}_q)\} \right]}{\frac{1}{k} \sum_{i=1}^k \left[ \min_{j \in K, K, Hi} \{d(\vec{m}_i, \vec{m}_j)\} \right]} \\ &= \frac{\sum_{i=1}^k \left[ \frac{1}{N_i} \sum_{\vec{x}_i \in C_i} \max_{\vec{x}_q \in C_i} \{d(\vec{x}_i, \vec{x}_q)\} \right]}{\sum_{i=1}^k \left[ \min_{j \in K, j+i} \{d(\vec{m}_i, \vec{m}_j)\} \right]}. \end{aligned} \quad (39)$$

The CS measure now reduces to employing a Gaussian kernelized distance measure as well as translating to HD feature space as shown in eq.(40).

$$\begin{aligned} CS_{kernel}(k) &= \frac{\sum_{i=1}^k \left[ \frac{1}{N_i} \sum_{\vec{x}_i \in C_i} \max_{\vec{x}_q \in C_i} \left\{ \|\varphi(\vec{x}_i) - \varphi(\vec{x}_q)\|^2 \right\} \right]}{\sum_{i=1}^k \left[ \min_{j \in K, j+i} \left\{ \|\varphi(\vec{m}_i) - \varphi(\vec{m}_j)\| \right\} \right]} \\ &= \frac{\sum_{i=1}^k \left[ \frac{1}{N_i} \sum_{\vec{x}_i \in C_i} \max_{\vec{x}_q \in C_i} \left\{ 2(1 - K(\vec{x}_i, \vec{x}_q)) \right\} \right]}{\sum_{i=1}^k \left[ \min_{j \in K, j+i} \left\{ 2(1 - K(\vec{m}_i, \vec{m}_j)) \right\} \right]} \end{aligned} \quad (40)$$

$\vec{Z}_i = (Z_{i1}, Z_{i2}, Z_{i3}, \dots, Z_{in})$  and  $\vec{V}_i = (V_i, V_{i2}, \dots, \dots, V_{in})$  and velocities are updated, and function is calculated with new coordinates at every time-step in an n-dimensional search space. The fundamental update equations for the ith particle's dth dimension in PSO can be written as eq.(41)

$$\begin{aligned} V_{id}(t+1) &= \omega \cdot V_{id}(t) + C_1 \cdot \phi_1 \cdot (P_{Rid} - X_{iu}(t)) \\ &\quad + C_2 \cdot \phi_2 \cdot (P_{gd} - X_{id}(t)) \\ V_i, \dots, V_{it} \cdot \vec{V}_i, \quad X_{id}(t+1) &= X_{id}(t) + V_{id}(t+1). \end{aligned} \quad (41)$$

The parameters of the system are the upper limit  $\phi_{max}$  and the lower limit  $\phi_{min}$ , which define the variables  $\phi_1$  and  $\phi_2$ , which are random positive values chosen at random from a

uniform distribution,  $\sim P_l$  denotes local best solution discovered thus far by ith particle.

#### 4. Performance analysis:

The working platform of NS2 has adopted the suggested method for secure routing and energy optimization in WSN. In Table I, simulation parameter is listed. 250 nodes are deployed in the WSN in our work. Six clusters are then used to cover the deployment area. To begin with, cluster heads are chosen to group together in the transmission area.

Table 1. Simulation parameters

Parameter	Values
Number of nodes	250
Area	300*300m
Initial energy	1J
Transmitted power	0.02 watts
Data packet size	1024 bits/sec
MAC type	IEEE 802.11
Received power	0.01 watts
Transmission range	120m
Frequency range	5GHZ

Table-2 Comparative analysis of Throughput

Number of Nodes	LEACH	AODV	6LoWPAN_WSN_IoT_5G
50	65	71	77
75	68	73	81
100	71	75	85
125	73	77	88
150	75	81	91
175	77	83	92
200	81	85	96

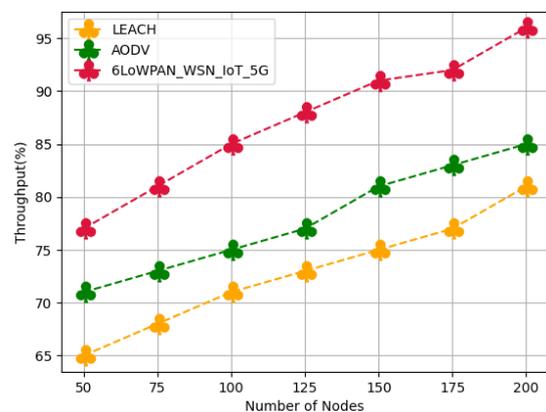
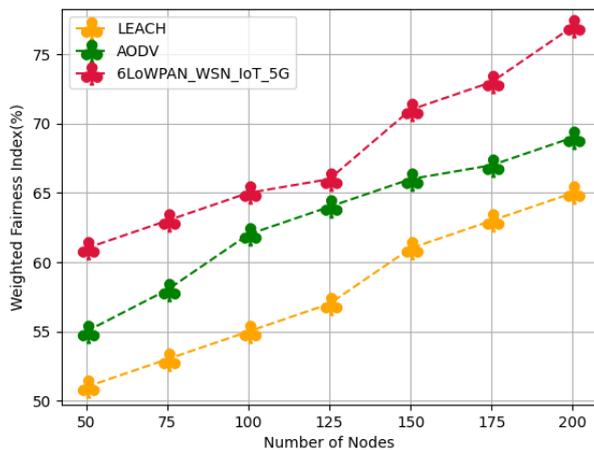


Figure-3 Comparative analysis of Throughput

The above table-2 shows comparative analysis of throughput between proposed and existing technique based on number of nodes. Here proposed method is compared with existing LEACH and AODV. Proposed technique attained throughput of 96% for 200 nodes while LEACH attained 81% and AODV obtained 85% for 200 nodes as shown in figure-3.

**Table-3 Comparative analysis of weighted fairness index**

Number of Nodes	LEACH H	AODV V	6LoWPAN_WSN_IoT_5 G
50	51	55	61
75	53	58	63
100	55	62	65
125	57	64	66
150	61	66	71
175	63	67	73
200	65	69	77

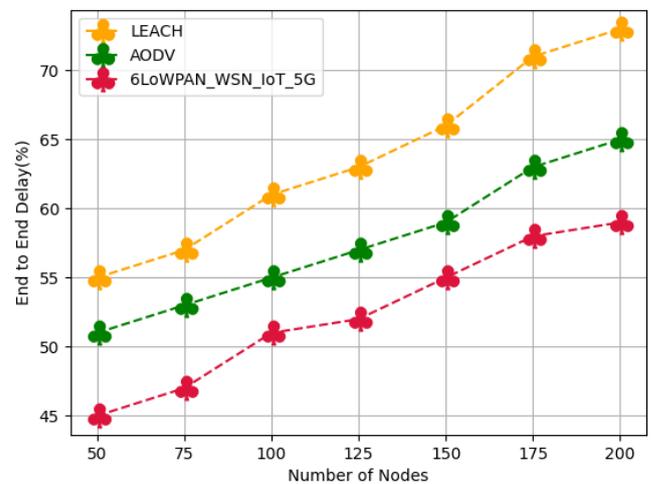


**Figure-4 Comparative analysis of weighted fairness index**

The above table-3 shows comparative analysis of weighted fairness index between proposed and existing technique. It is employed to gauge unfairness and disparities. Its range of values is 0 to 1. The lower number 0 denotes highly fair, whereas the larger value 1 suggests highly unfairness. The proposed technique attained weighted fairness index of 77% for 200 nodes and LEACH obtained 65%, AODV attained 69% for 200 nodes as shown in figure-4.

**Table-4 Comparative analysis of end-to-end delay**

Number of Nodes	LEACH H	AODV V	6LoWPAN_WSN_IoT_5 G
50	55	51	45
75	57	53	47
100	61	55	51
125	63	57	52
150	66	59	55
175	71	63	58
200	73	65	59



**Figure-5 Comparative analysis of End- end delay**

The above table-4 represents comparative analysis of end-end delay between proposed and existing technique. Proposed technique attained end-end delay of 59%, while existing LEACH protocol attained 73% and AODV obtained 65% for 200 nodes as represented in figure-5

**Table- 5 Comparative analysis of Energy Consumption**

Number of Nodes	LEACH H	AODV V	6LoWPAN_WSN_IoT_5 G
50	61	65	71
75	63	69	73
100	65	71	75
125	67	73	77
150	71	75	82
175	73	77	84
200	75	81	86

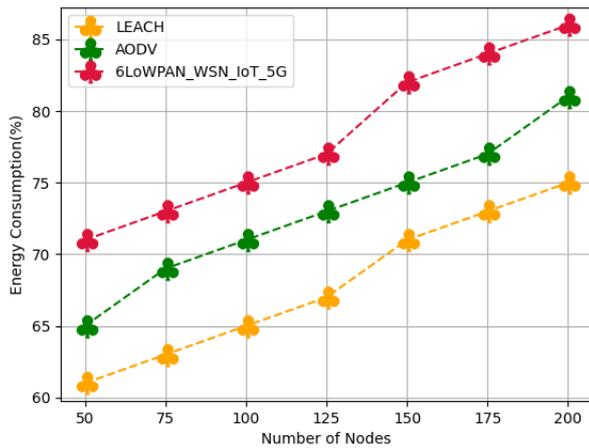


Figure-6 Comparative analysis for energy consumption

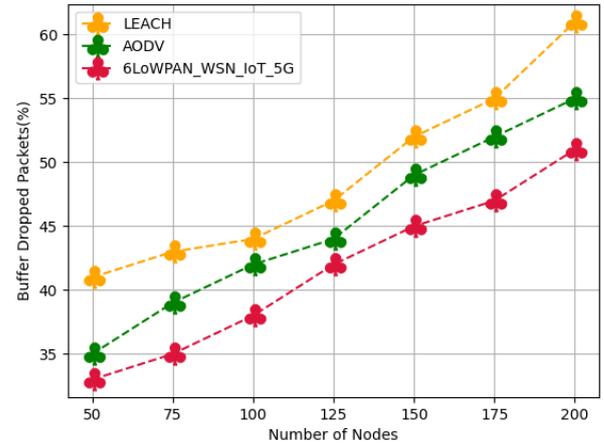


Figure-7 Comparative analysis of buffer dropped packets

The above table- 5 shows comparative analysis between proposed and existing technique in terms of energy consumption based on number of nodes. Energy consumption refers to the overall amount of energy used by the network to carry out data aggregation, transmission, and reception. The comparisons made between the various methods were based on how much energy was used by the cluster member and cluster head sensor nodes. Here the proposed technique attained energy consumption of 86%, while existing LEACH attained 75% and AODV obtained 81% of energy consumption of the network for 200 nodes as shown in figure-6.

Table-6 Comparative analysis of buffer dropped packets

Number of Nodes	LEACH	AODV	6LoWPAN_WSN_IoT_5G
50	41	35	33
75	43	39	35
100	44	42	38
125	47	44	42
150	52	49	45
175	55	52	47
200	61	55	51

The table-6 shows comparative analysis between proposed and existing technique in terms of buffer dropped packets. When buffers are insufficient, packets are lost to alert the sender and slow down transmission. Dropped TCP flows begin to back off and, logically, use less bandwidth than other flows that do not. Here proposed technique attained buffer dropped packets of 45%, LEACH attained 52% and AODV of 49% for 200 nodes as shown in figure-7.

## 5. Conclusion:

This research proposed novel technique in 6LoWPAN based network secure routing as well as EE for WSN in IoT application based on 5G technology. The energy optimization has been carried out using clustered channel aware least square support vector machine (CI\_CHLSSVM). Routing is done using Fuz\_RPL\_KPSO for security. The experimental analysis has been carried out in terms of throughput of 96%, weighted fairness index of 77%, end-to-end delay of 59%, energy consumption of 86%, and buffer dropped packets of 51%. Because there is less space available for energy sources, our suggested method to increase wearability focuses on the utilisation of smaller sensor nodes, necessitating more energy-efficient networks. Smaller WSN nodes must function for extended periods of time without human intervention while also offering the necessary levels of dependability and service quality. The routing protocol, the network topology, and energy efficiency are only a few of the crucial variables that affect whether these two objectives are achieved.

## Reference:

- [1]. Han, B., Ran, F., Li, J., Yan, L., Shen, H., & Li, A. (2022). A Novel Adaptive Cluster Based Routing Protocol for Energy-Harvesting Wireless Sensor Networks. *Sensors*, 22(4), 1564.



- [2]. Qureshi, K. N., Bashir, M. U., Lloret, J., & Leon, A. (2020). Optimized cluster-based dynamic energy-aware routing protocol for wireless sensor networks in agriculture precision. *Journal of sensors*, 2020.
- [3]. Faheem, M., Butt, R. A., Raza, B., Alquhayz, H., Ashraf, M. W., Shah, S. B., ... & Gungor, V. C. (2019). QoS-RP: A cross-layer QoS channel-aware routing protocol for the Internet of underwater acoustic sensor networks. *Sensors*, 19(21), 4762.
- [4]. Das, S., Abraham, A., & Konar, A. (2008). Automatic kernel clustering with a multi-elitist particle swarm optimization algorithm. *Pattern recognition letters*, 29(5), 688-699.
- [5]. Khashan, O. A., Ahmad, R., & Khafajah, N. M. (2021). An automated lightweight encryption scheme for secure and energy-efficient communication in wireless sensor networks. *Ad Hoc Networks*, 115, 102448.
- [6]. Kumar, M., Mukherjee, P., Verma, K., Verma, S., & Rawat, D. B. (2021). Improved deep convolutional neural network based malicious node detection and energy-efficient data transmission in wireless sensor networks. *IEEE Transactions on Network Science and Engineering*.
- [7]. Nagaraju, R., Goyal, S. B., Verma, C., Safirescu, C. O., & Mihaltan, T. C. (2022). Secure Routing-Based Energy Optimization for IoT Application with Heterogeneous Wireless Sensor Networks. *Energies*, 15(13), 4777.
- [8]. Toma, D. M., del Rio, J., & Mànuel, A. Research Article Wireless HDLC Protocol for Energy-Efficient Large-Scale Linear Wireless Sensor Networks.
- [9]. Li, S., Kim, K. S., Zhang, L., Huan, X., & Smith, J. (2022). Energy-Efficient Message Bundling with Delay and Synchronization Constraints in Wireless Sensor Networks. *Sensors*, 22(14), 5276.
- [10]. Bettoumi, B., & Bouallegue, R. (2021). LC-DEX: Lightweight and efficient compressed authentication based elliptic curve cryptography in multi-hop 6lowpan wireless sensor networks in HIP-based Internet of things. *Sensors*, 21(21), 7348.
- [11]. Nashiruddin, A. H. N., & Rakhmawati, L. (2022). Literature Review: Energy Efficiency Mechanisms using Data Reduction in Wireless Sensor Networks (WSNs) Applications. *INAJEEE (Indonesian Journal of Electrical and Electronics Engineering)*, 5(1), 5-13.
- [12]. Shreyas, J., Deepa, H., Udayaprasad, P. K., Chouhon, D., Srinidhi, N. N., & SM, D. K. (2022, January). Energy Optimization to Extend Network Lifetime for IoT based Wireless Sensor Networks. In *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)* (pp. 90-93). IEEE.
- [13]. Sharma, S., & Verma, V. K. (2022). An integrated exploration on internet of things and wireless sensor networks. *Wireless Personal Communications*, 124(3), 2735-2770.
- [14]. Li, Y., Li, J., & Huang, X. (2021, December). An Energy Efficiency Evaluation Scheme of IPHC over CSMA and TSCH in 6LoWPAN Networks. In *2021 The 10th International Conference on Networks, Communication and Computing* (pp. 135-140).
- [15]. Wagle, S. K., Bazilraj, A. A., & Ray, K. P. (2021, September). Energy efficient security solution for attacks on Wireless Sensor Networks. In *2021 2nd International Conference on Advances in Computing, Communication, Embedded and Secure Systems (ACCESS)* (pp. 313-318). IEEE.
- [16]. Idrees, A. K., & Witwit, A. J. (2021). Energy-efficient load-balanced RPL routing protocol for internet of things networks. *International Journal of Internet Technology and Secured Transactions*, 11(3), 286-306.
- [17]. Gupta, S. K., & Singh, S. (2022). Survey on energy efficient dynamic sink optimum routing for wireless sensor network and communication technologies. *International Journal of Communication Systems*, e5194.
- [18]. Ali, O. M. E. R., & Ishak, M. K. (2021). Energy efficient scheme for wireless sensor networks based on contikimac protocol. *J. Eng. Sci. Technol. (JESTEC)*, 16, 4666-4686.
- [19]. Patra, B. K., Mishra, S., & Patra, S. K. (2022). Genetic Algorithm-Based Energy-Efficient Clustering with Adaptive Grey Wolf Optimization-Based Multipath Routing in Wireless Sensor Network to Increase Network Life Time. In *Intelligent Systems* (pp. 499-512). Springer, Singapore.
- [20]. Sharma, P. K., Singh, J., & Pal, V. (2021). Low power communication in wireless sensor networks and IoT. In *Smart Sensor Networks Using AI for Industry 4.0* (pp. 221-233). CRC Press.

