INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2026, VOL. 72, NO. 2, PP. 1-6
Manuscript received August 19, 2025; revised April, 2026. doi: 10.24425/ijet.2026.157916

Congestion avoidance in DEC protocol using Leaky Bucket Algorithm

Habibulla Mohammad, Ch Gangadhar, K. Phani Rama Krishna, and Riazuddin Mohammed

***Abstract*—A wireless sensor network is vital in various fields and entails of a large number of sensor nodes. These nodes perform multiple functions, such as sensing, processing, communication, and power management. In such networks, a large amount of data is generated and transmitted from sensor nodes to the destination. During data transmission, congestion may occur at the cluster head as well as during communication between sensor nodes. This congestion mainly arises due to inefficient resource allocation and uneven traffic distribution. The main reason for packet loss and the need for retransmission are due to traffic imbalance and congestion. To avoid such circumstances an efficient flow control and congestion controlling schemes need to be adapted. To manage congestion control, a limited number of wireless sensor networks with different protocols are employed. The Deterministic Energy Efficient Clustering (DEC) protocol, which relies entirely on residual energy, is considered to minimize energy consumption. This protocol, along with the leaky bucket algorithm, is used to regulate data flow at the cluster head and effectively control congestion in the network. This idea offers a solution to address traffic congestion and makes improvements to the situation in the event that it arises. Simulation results show that the defined method can considerably advance increases in lifetime, energy and throughput.**

***Keywords*—Packet size; Base station; Residual energy; Congestion**

I. INTRODUCTION

AT Present, light-weight sensor nodes are made possible by improvements in microchip design. These sensor nodes are deployed in various applications in which oversight is required. These applications include disaster-prone areas, agriculture, healthcare, and other critical sectors.The efficiency of these sensor networks frequently relies on how well battery life is managed. The limited battery capacity in wireless sensor networks increases costs and poses a challenge to their widespread adoption. As a result, creating a system that improves the amount of energy used has become essential. An energy-aware protocol that can manage its own energy consumption is needed, and this has been recognized.

Using clustering, you can control Wireless Sensor Networks [1][2][3][4][5][6][7]. Clusters are produced by the sensors that are closer together. The available sensor nodes in each cluster elect a cluster head. It's called clustering process. The data is given to the cluster head by the others in the cluster when they are elected. They employ methods like data compression to reduce the amount of collected data before sending it to the base station (BS). Additionally, the cluster head role requires extra energy because it must receive data from all cluster members, perform data processing and aggregation, and transmit the aggregated data over a longer distance to the base station, resulting in higher communication and computational overhead. Rotating the cluster-head among the respondents would yield greater energy gains than fixing the same node. Thus, one of the key elements influencing the success of an improved protocol is how well distributive wireless sensor networks are designed to control energy usage. Prior to this, the cluster heads were randomly rotated, and the election of cluster head was not guaranteed to be optimized. For enhancing the energy efficiency, a protocol that selects an optimal cluster head should be employed. The DEC protocol illustrates this method, selecting cluster heads by tracking the sensor nodes' remaining energy.

Congestion in a wireless sensor network occurs when the traffic volume at a particular sensor node exceeds its available buffer capacity. Unfair usage of network resources can also lead to congestion. Congestion can be classified into two categories: link-level congestion and node-level congestion. Node-level congestion typically occurs at sensor nodes due to limited processing and buffer resources. In such cases, buffer utilization is assessed by measuring the difference between the traffic arrival rate and the traffic departure rate. Conversely, link-level congestion is identified by excessive network usage on communication links. Early congestion detection and effective mitigation are critical in both cases. These objectives can be achieved by employing the leaky bucket algorithm, which regulates the packet transmission rate and prevents buffer overflow, thereby reducing congestion and improving overall network stability. By enforcing a controlled output rate, the algorithm ensures fair bandwidth utilization among sensor nodes. This controlled traffic flow enhances reliable delivery and supports efficient network operation under varying traffic conditions.

II. MATERIAL AND METHODS

A probabilistic energy consumption control scheme has been implemented in wireless sensor networks to extend network lifetime by utilizing global information rather than local parameters such as residual node energy. However, a major limitation of these protocols is that the selected Cluster Heads

This work was supported by Research & Development Cell-ECE Department, PVP Siddhartha Institute of Technology.

Habibulla Mohammad, Ch Gangadhar, and Phani Rama Krishna are with PVP Siddhartha Institute of Technology. India (e-mail: haji.habibulla.md@gmail.com).
Riazuddin Mohammed is with University of Alberta, Edmonton, AB, Canada (e-mail: riaz70md@yahoo.co.in).

(CHs) and the total number of CHs may not possess sufficient energy to perform network operations efficiently. To overcome this issue, a deterministic cluster-head selection strategy is preferred, as it provides better energy balance compared to probabilistic approaches. By explicitly considering energy consumption, the deterministic method ensures more reliable CH selection. Based on this observation, a generalized probabilistic energy model is formulated and expressed in Eq. (1).

$$T(n) = \begin{cases} \dfrac{p}{1-\left(r mod \dfrac{1}{(P_x)}\right) * P} \times Q \ if n_x \in G; \\ 0 \qquad\qquad Otherwise, \end{cases} \qquad (1)$$

Here nrm, int, adv represent normal, intermediate, or advanced nodes, respectively, and correspond to x, while Q can either be a constant value or a function of each node's residual energy ratio. For example, assume Q is set to one[8][9]. During each round, the sensor node randomly generates a number between 0 and 1. A node will decide to become a cluster leader based on the threshold value defined by Eq. (2). A sensor node will become a cluster leader (n) if the generated value is less than the threshold. The nodes that are not selected as leaders, along with their probability of being chosen as cluster members (CMs), are denoted by G and P_x.

According to the DEC protocol, clustering can take place when E_{T_x} represents the energy consumed per bit by the transceiver circuit, and $d_{to\ CH}$ indicates the distance threshold for selecting amplification models. During the setup phase, all nodes use the indication function to elect Cluster-heads (CHs). The selected Cluster-heads will receive an A_k-bit message, and radio resources will be announced through an Advertisement message (ADV), utilizing the non-persistent Carrier Sense Multiple Access (CSMA MAC) protocol. The Cluster-heads ID and a header are two components of an announcement message. This deterministic selection mechanism ensures balanced energy consumption among sensor nodes and prevents premature energy depletion at cluster heads. As a result, the network stability period is extended, thereby significantly improving the overall network lifetime.

The equation for the Cluster-head selection

$$\mathrm{T(n)} = \begin{cases} 1 & \text{if } \mathrm{E_n} > \mathrm{E_{T_x}} \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$

The cluster-members are non-elected nodes (CMs). [10].[11][12][13][14][15]. Cluster members will send join requests to the cluster head with the simplest communication cost after establishing their cluster, utilizing the CSMA MAC technique that depends on the strength of the signal received of the advertisement message. This message contains additional data, $\mathrm{CM_{ID}}$, $\mathrm{CH_{ID}}$, and a header indicating that it is a request. For intra-cluster communication, TDMA configuration will be done by the cluster heads [16]. This step marks the end of the setup procedure. The steady-state phase begins with the transmission of sensed data from Cluster Members (CMs) to Cluster Heads (CHs) and then from CHs to the Base Station. One approach for facilitating inter-cluster communication is through the direct sequence spread spectrum (DSSS) technique.

Multiple factors can lead to congestion in a network, including buffer overflow and time-varying channel conditions. Congestion detection refers to the process of identifying abnormalities in normal traffic flow, which affect reliable packet transmission between sensor nodes [17]. Node-level congestion results in significant packet loss due to buffer overflow at sensor nodes. An increase in packet loss leads to higher energy consumption and inefficient utilization of communication links. Link-level congestion arises when multiple sensor nodes attempt to access the shared wireless channel simultaneously. As nodes transmit data concurrently, packet collisions occur, resulting in reduced link utilization. All the above-mentioned issues can be avoided or effectively managed through efficient congestion control mechanisms. One such technique is the leaky bucket algorithm [18][19]. By enforcing a controlled packet transmission rate, the leaky bucket algorithm mitigates excessive traffic bursts at both node and link levels. Consequently, it enhances reliable data delivery while improving energy efficiency and overall network performance.

Leaky Bucket data transmission

To effectively regulate incoming packets under dynamic network conditions, an adaptive traffic control mechanism is employed. The algorithm continuously monitors buffer occupancy and dynamically adjusts packet transmission rates to prevent congestion and ensure stable network performance.

The bucket capacity B= maximum bucket size

Let $X(t)$ represent the amount of data in the bucket at time t. The following differential equation describes how the bucket status changes over time.

$$\frac{\mathrm{dX(t)}}{\mathrm{dt}} = \lambda - \mu \qquad (3)$$

Where λ represents the rate at which data arrives and μ denotes the rate at which data is sent out

If $\mathrm{X(t)} > \mathrm{B}$ (4)

The incoming packets are dropped when it signals the arrival of the packet and the bucket's capacity is reached. As shown in Figure 1, it works in a FIFO manner. The DEC protocol adopted this technique.

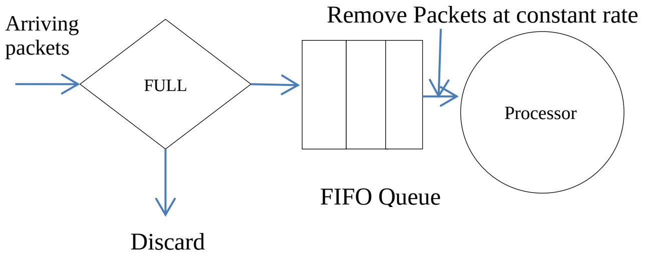


Fig.1 Leaky bucket algorithm Operation

Once the queue size reaches its maximum capacity, a predefined threshold value is computed. The generated packets are released until sufficient buffer space becomes available, which is then allocated for storing incoming packets. This mechanism protects the network from sudden packet bursts at the network access point. By regulating the packet release rate, the leaky bucket algorithm ensures smooth traffic flow and prevents abrupt congestion. As a result, packet loss is

minimized, and stable network operation is maintained under varying traffic conditions.
This can be modelled as:

$$X(t) = \begin{cases} \lambda t - \mu t & \text{if } \lambda t - \mu t \leq B \\ B & \text{if } \lambda t - \mu t > B \end{cases} \tag{5}$$

In the steady state,
where the bucket is neither empty nor overflowing

$$\lambda = \mu \tag{6}$$

The amount of data in the bucket remains constant, and the bucket does not overflow

$$X(t) = \text{constant} = \frac{\lambda - \mu}{\text{steady state}} \tag{7}$$

if λ is greater than the leak rate μ

$$\text{Dropped data} = \lambda t - \mu t - B \tag{8}$$

t is the time during which the overflow occurs.
The effective transmission rate R_{eff} after considering the Leaky Bucket algorithm

$$R_{eff} = \mu \tag{9}$$

To control traffic, the average queue size L and the mean wait time W can be examined by considering the balance between B, λ, μ

$$L = \frac{\lambda - \mu}{\mu} \ (\text{if } \lambda > \mu) \tag{10}$$

$$W = \frac{L}{\mu} \tag{11}$$

As a result, system traffic activity is redirected onto worthy stages, preventing congestion and enhancing quality of service.

Leaky Bucket Algorithm

The leaky bucket algorithm is a traffic shaping and congestion control mechanism that regulates packet transmission by enforcing a fixed output rate. Incoming packets are temporarily stored in a buffer and released in a controlled manner to prevent sudden traffic bursts. This approach effectively reduces congestion, minimizes packet loss, and improves overall network stability.

```
Start
Enter the size of the packet.
Enter the buffer (bucket) size.
Enter output rate (leak rate).
Set queue = 0
Set time interval = 0
 For (i = 0; i < packet number; i++)
 If (queue + packet size[i] ≤ buffer size)
      Accept packet
      Add packet to queue
  Else
      Discard packet (overflow)
 If (queue > 0)
      Transmit data at fixed leak rate
      queue = queue − leak rate × time interval
 If (queue < 0)
       queue = 0
Update time interval
End For
End
```

III. EXPERIMENTAL

To avoid packet loss and guarantee seamless data transmission, the leaky bucket algorithm is integrated into the energy-efficient DEC protocol. This approach helps to mitigate congestion at both the nodes and the channel. As a result, controlled traffic flow is maintained, leading to improved throughput and enhanced network reliability.

Energy computation:

The energy necessary for transmitting a bit payload over a specific distance is estimated to be as indicates.

$$E_{t_x}\ k, d = E_{elect} * k + \varepsilon f_s * k * d * d \ \ if\ (d < d_0) \tag{12}$$

$$E_{elect} * k + \varepsilon amp * k * d * d * d * d \ \ if\ (d > d_0) \tag{13}$$

Where εf_s represents the allowed space, εamp denotes the multipath loss, d is the distance among the origin and target nodes, and d_0 is the threshold distance.

$$d_0 = square\ root\ (\frac{\varepsilon f_s}{\varepsilon amp}) \tag{14}$$

The received radio energy

$$E_{Rx}(k) = k * E_{elect} \tag{15}$$

Leaky Bucket principle application in DEC protocol:

Stage 1: By adopting a standard technique, any arrangement of sensor hosts can be put together.

Stage 2: The DEC protocol computation is utilized for clustering groups.

Stage 3: Utilize the leaky bucket technique on the nodes that are part of different clusters. In order to respond to cluster members more effectively and uniquely, Sink will issue CH-tickets.

Stage 4: After assessing, evaluate cluster chiefs who have tickets. The following describes the limit:

$$T(n) = \{ {}^{t}/_{1-t} * (r\ mod\ {}^{1}/_{t} \} \tag{16}$$

Where t is preferred CHs number

Stage 5: A lower-level node segment function as cluster members through neighboring CHs.

Stage 6: The Cluster Leader will distribute the ticket to the Cluster Members.

Stage 7: Nodes that comprise a ticket communicate with the corresponding detected CH by sharing their data.

Stage 8: Apply data to obtain information.

Stage 9: The CH then sends the information to the Sink.

Stage 10: Compute and report energy values.

Stage 11: The dead nodes in each cluster are counted and validated.

Stage 12: The lifespan is determined when all nodes in the phase are inactive; if not, proceed to step12.

IV. RESULTS AND ANALYSIS

Table I includes the operating specifications for evaluating data. The MATLAB platform is used to obtain simulation results.

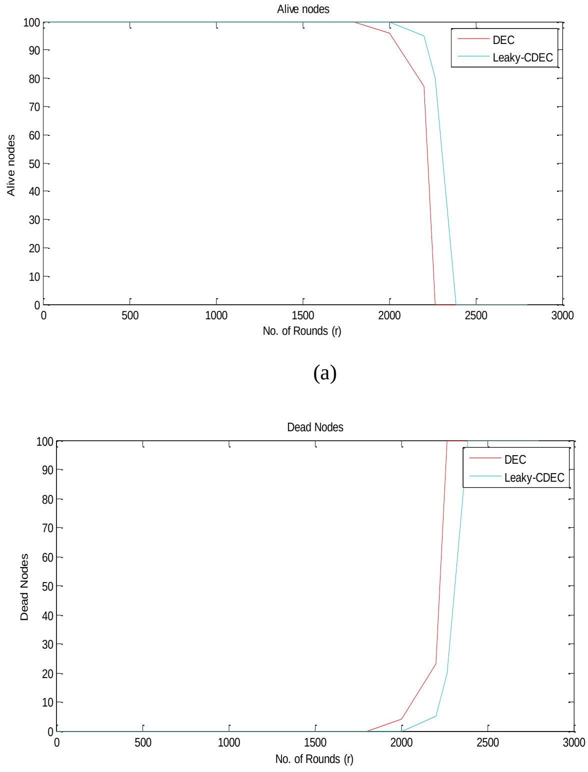
TABLE I
DESIGN SPECIFICATION

Parameter	Values
E_{elec}	50nJ/bit
E_{DA}	5nJ/bit/message
E_0	0.5J
k	4000
p_{opt}	0.1
ϵ_{fs}	10pJ/bit/m^2
ϵ_{mp}	0.0013pJ/bit/m^4
n	1000

Stability Period: The stability period indicates the number of rounds during which all sensor nodes remain alive before the first node depletes its energy. In the proposed Leaky-CDEC protocol, this period is evaluated during the steady-state phase by observing active and passive nodes, as shown in Fig. 2(a) and Fig. 2(b), and by analysing the number of live nodes across rounds in Table II. Both DEC and Leaky-CDEC maintain full node survival up to 1500 rounds; however, DEC begins to experience node failures thereafter, while Leaky-CDEC sustains all nodes alive even at 2000 rounds. The trends in Fig. 2(a) demonstrate a delayed reduction in alive nodes for Leaky-CDEC, whereas Fig. 2(b) shows a slower increase in dead nodes compared to DEC. Simulation results indicate that the first node in Leaky-CDEC dies around the 1972nd round, compared to the 1860th round in DEC, achieving a stability improvement of more than 5.9%. This enhancement is attributed to energy-aware cluster-head selection and effective congestion control through the leaky bucket mechanism, which minimizes packet loss and avoids unnecessary retransmissions. Consequently, balanced energy utilization is achieved across the network, leading to extended network lifetime and improved reliability under continuous and high-traffic data transmission, thereby making Leaky-CDEC more suitable for long-term and energy-constrained wireless sensor network applications.

TABLE II
ANALYSIS OF LIVE NODES

Rounds	Live nodes	
	DEC	**Leaky- CDEC**
0	1000	1000
500	1000	1000
1000	1000	1000
1500	1000	1000
2000	981	1000



(a)

(b)

Fig. 2 (a) & (b) Analysis

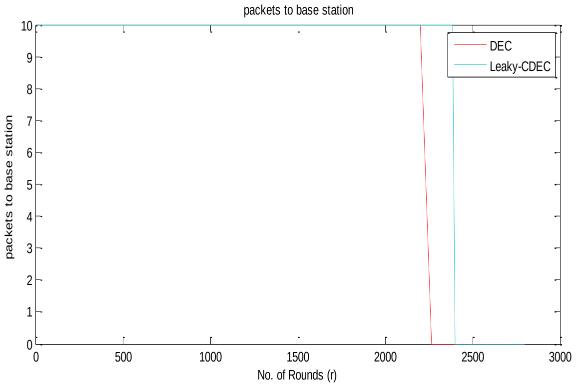


Fig. 3 Cluster head count analysis

CH's evaluation: Figure 3 shows a consistent number of Cluster Head (CH) nodes in a network of 100 deployed sensor nodes. The CH selection is based on residual energy comparison, ensuring that nodes with higher remaining energy are more likely to be elected, which balances energy consumption and prevents early node exhaustion. Dynamic CH rotation further reduces the communication burden on individual nodes and enhances network robustness, while controlled traffic flow supports stable CH operation across rounds. As a result, the proposed approach sustains network operation up to approximately 2396 rounds, compared to about 2265 rounds in the DEC protocol, indicating an earlier transition to the dead stage in DEC. Consequently, after each cycle, CH nodes revert to normal nodes and communicate independently with the sink until all nodes deplete their energy.

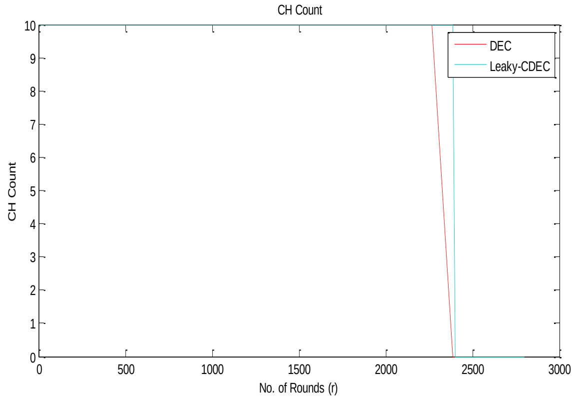


Fig.4 Packet analysis to base station

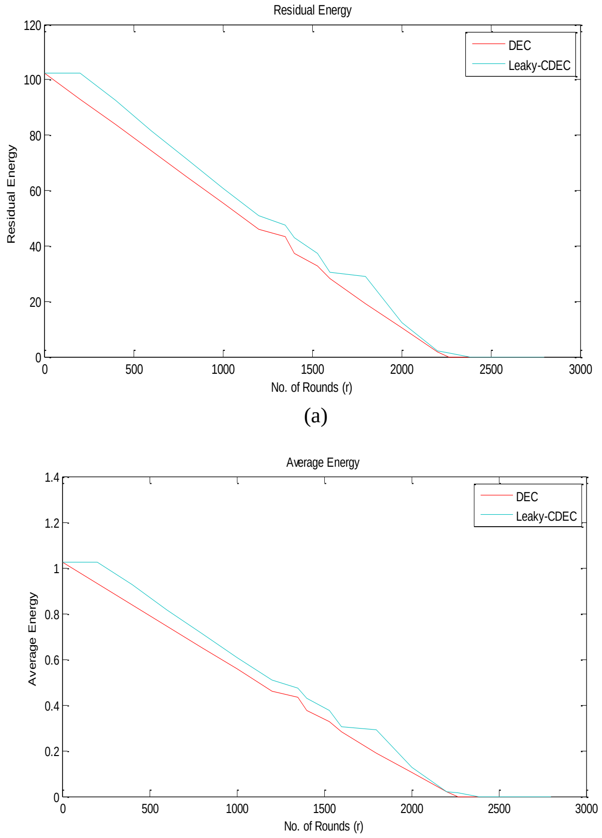
Packet Delivery Ratio (PDR): The percentage is calculated as the amount of packets transmitted scaled by the amount of packets received. PDR serves as a key indicator of the reliability and efficiency of data transmission in wireless sensor networks. A higher PDR reflects effective congestion control and reduced packet loss during communication**.** As depicted in Figure 4, the designed technique is far better than the existing one. This improvement confirms the effectiveness of the proposed method in maintaining reliable data delivery even under high traffic conditions. Consequently, overall network performance is significantly enhanced.

Energy Performance in Leaky-CDEC and DEC:

Figures 5(a) and (b) provide an analysis of average and residual energy using the DEC and Leaky-CDEC protocols. Table III presents a comparative simulation of the DEC and Leaky-CDEC results. The observed energy trends indicate that Leaky-CDEC achieves more balanced energy consumption due to controlled data transmission and efficient cluster-head selection. This balanced utilization delays rapid energy depletion across sensor nodes**.** The collected findings show that the energy of Leaky-CDEC is 1.09 J at round 2265, whereas the overall energy of the DEC protocol reaches zero. Beginning with round 2384, Leaky-CDEC performance gradually declines. Leaky-CDEC surpasses DEC by sustaining network operation for a greater number of rounds. Overall, Leaky-CDEC outperforms the conventional DEC protocol by up to 5.36%.This improvement is mainly attributed to effective congestion control and regulated data transmission enabled by the leaky bucket mechanism.

TABLE III
ANALYSIS OF RESIDUAL ENERGY AND AVERAGE ENERGY

Rounds	Residual energy		Average Energy	
	DEC	**Leaky-CDEC**	**DEC**	**Leaky-CDEC**
0	102.5	102.5	1	1
500	79.17	82.51	0.790	0.797
1000	55.41	57.78	0.561	0.574
1500	33.12	34.52	0.332	0.351
2000	9.648	10.07	0.101	0.124



(a)

(b)

Fig. 5.(a) and (b): Evaluation of average and residual energy for DEC compared to the proposed Leaky-CDEC

V. CONCLUSION

Leaky Bucket–based congestion control integrated with the DEC protocol provides an effective mechanism for preventing network congestion during data transmission in wireless sensor networks. The defined approach addresses congestion challenges in isolated and resource-constrained environments by regulating packet transmission rates, thereby minimizing buffer overflow and packet loss at both node and link levels. By combining efficient congestion control with energy-aware cluster-head selection, the Leaky-CDEC protocol ensures balanced energy consumption across sensor nodes. This balance directly contributes to prolonged network stability and reduced premature node failures. Key performance metrics, including energy consumption, packet transmission efficiency, network lifetime, and throughput, are evaluated to validate the effectiveness of the defined approach. Simulation results confirm that the defined Leaky-CDEC protocol consistently outperforms the conventional DEC protocol, demonstrating improved network performance, higher reliability, and extended operational lifetime. Moreover, the controlled traffic flow enables scalable performance under varying traffic conditions without compromising energy efficiency. Hence, Leaky-CDEC emerges as a robust and practical solution for long-term deployment in wireless sensor networks**.**

REFERENCES

[1] F. A. Aderohunmu, J. D. Deng and M. K. Purvis, A deterministic energy-efficient clustering protocol for wireless sensor networks,Seventh International Conference on Intelligent Sensors, Sensor Networks and Information Processing, 2011 , pp. 341-346 https://doi:10.1109/ISSNIP.2011.6146592

[2] F. Comeau, Optimal Clustering in Wireless Sensor Networks Employing Different Propagation Models And Data Aggregation Techniques (PhD thesis). Dalhousie University, Halifax, Nova Scotia, 2008.

[3] M. J. Handy, M. Haase,D. Timmermann, Low energy adaptive clustering hierarchy with deterministic cluster-head selection,4th International Workshop on Mobile and Wireless Communications Network, 2002, pp. 368-372 https://doi:10.1109/MWCN.2002.1045790

[4] W. R. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy efficient communication protocol for wireless microsensor networks, In Proceeding 33rd Hawaii International Conference on System Sciences, 2000,pp. 10 https://doi:10.1109/HICSS.2000.926982

[5] W. B. Heinzelman, A. P. Chandrakasan,H. Balakrishnan, An application-specific protocol architecture for wireless microsensor networks. IEEE Transactions on Wireless Communications, 1(4), 2002: 660-670 https://doi:10.1109/TWC.2002.804190

[6] Fang Zhu, Junfang Wei, An energy-efficient unequal clustering routing protocol for wireless sensor networks, International Journal of Distributed Sensor Networks,15(9), 2019: 1-15. https://doi.org/10.1177/1550147719879384

[7] S. Gamwarige and C. Kulasekere, Performance analysis of the EDCR algorithm in a distributed wireless sensor network,IFIP International Conference on Wireless and Optical Communications Networks, 2006, pp. 5 pp.-5 https://doi:10.1109/WOCN.2006.166654

[8] Javaid N., Rasheed M.B., Imran M ,Mohsen G,Z A Khan,Turki A A ,M Ilahi, An energy-efficient distributed clustering algorithm for heterogeneous WSNs. J Wireless Com Network , 151 (2015): 1-11 https://doi.org/10.1186/s13638-015-0376-4

[9] Md.Khorshed Alom, Arif H, P K.Choudhury, Improved Zonal Stable Election Protocol (IZ-SEP) for hierarchical clustering in heterogeneous wireless sensor networks, e-Prime - Advances in Electrical Engineering, Electronics and Energy, 2(2022),2022: 1-8 https://doi.org/10.1016/j.prime.2022.100048

[10] S. Nasr , M. Quwaider, LEACH Protocol Enhancement for Increasing WSN Lifetime, 11th International Conference on Information and Communication Systems (ICICS), 2020, pp. 102-107 https://doi:10.1109/ICICS49469.2020.239542

[11] Habibulla Mohammad, A. S. Chandrasekhara Sastry, ACNM:Advance coupling networkmodelsleep/wake mechanism for wireless sensor networks. International Journal of Engineering and Technology(UAE), 7(1.1),2018: 350-354 https://doi:10.14419/ijet.v7i1.1.9851

[12] Habibulla Mohammad, A. S. Chandrasekhara Sastry, Implementation of Multihop technique in DEC Protocol,International Journal of Simulation systems, Science & Technology(UAE), 19(4), 2018: 20.1-20.9 https://doi:10.5013/IJSSST.a.19.04.20

[13] Habibulla Mohammad, A. S. Chandrasekhara Sastry ,Implementation of Three-Tier Multihop Technique in Advance Coupling Network Model-Deterministic Energy-Efficient Clustering of Wireless Sensor Networks, Journal of Computational and Theoretical Nanoscience, 16(5-6), 2019: 2581-2589 https://doi.org/10.1166/jctn.2019.7934

[14] Habibulla Mohammad, A. S. Chandrasekhara Sastry , DSWS-Distributed Sleep/Wake Scheduling Scheme for DEC Protocol in Wireless Sensor Networks, International Journal of Recent Technology and Engineering, l 8(2),2019: 2695-2701https://doi:10.35940/ijrte.B2810.078219

[15] S. Hafidi, N. Gharbi and L. Mokdad, Queuing and service management for congestion control in wireless sensor networks using Markov Chains, 2017 IEEE Symposium on Computers and Communications (ISCC), 2017,pp. 176-181https://doi:10.1109/iscc.2017.8024525

[16] L. Chen and H. X. Li, A Simple Way to Reduce Congestion in Wireless Sensor Networks, 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2016,pp. 329-332 https://doi:10.1109/IHMSC.2016.143

[17] Habibulla Mohammad, K Phani Rama Krishna,CH Gangadhar ,Riazuddin Mohammed.(2023). Token Bucket Algorithm with Modernization Techniques to avoid Congestion in DEC Protocol of WSN, International Journal of Electronics and Telecommunications,63(3):507-513 https://doi.org/10.24425/ijet.2023.146499

[18] S. J., G. Y., S. A. S., C. K., A. R. and G. Raja, "Leaky Bucket based congestion control in Wireless Sensor Networks," *2018 Tenth International Conference on Advanced Computing (ICoAC)*, 2018, pp. 172-174, https://doi: 10.1109/ICoAC44903.2018.8939078

[19] Changle Li, Wenwei Yue, Guoqiang Mao, Zhigang Xu,Congestion Propagation Based Bottleneck Identification in Urban Road Networks, IEEE Transactions on Vehicular Technology, 69(5),2020:4827-4841 https://doi:10.1109/TVT.2020.2973404