FOREX Publication
Open Access | Rapid and quality publishing
International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 11, Issue 2 | Pages 315-322 | e-ISSN: 2347-470X

Congestion-Free Cluster Formation and Energy Efficient Path Selection in Wireless Sensor Networks using ButPCNN

S. Panimalar[1*] and Dr. T. Prem Jacob[2]

[1]Research Scholar, School of Computing, Sathyabama Institute of Science and Technology, Chennai, Tamil Nadu, India, panimalarjerome@gmail.com
[2]Professor, School of Computing, Sathyabama Institute of Science and Technology, Chennai, Tamil Nadu, India, premjac@gmail.com

***Correspondence:** S. Panimalar; panimalarjerome@gmail.com

ABSTRACT- Today, network congestion is a common occurrence that needs to be focused on and effectively addressed, particularly in Wireless Sensor Networks (WSN) for packed type networks. The main causes of congestion in WSN are a lack of channel capacity and energy waste. This study's major goal is to develop Energy Efficient Congestion Free Path Selection Protocol (ECFPSP) protocol, which aims to reduce network congestion. By selecting the most appropriate main cluster head (PCH) and secondary cluster head (SCH), the ECFPSP protocol is proposed to decrease end-to-end delay time and extend the network lifetime. The suggested protocol implements a routing protocol that provides security by avoiding hostile nodes and reducing data loss. It also routes the nodes. Hence, a Congestion-Free Cluster Formation is provided to increase the lifetime of the network by proposed ButPCNN approach. To decrease packet loss and conserve energy, this research also uses brand-new cluster-based WSNs. In comparison to other standard protocols, the simulation results reveal that ButPCNN has a reduced packet drop rate, which increases the ratio of packet distribution, network life, and residual energy. As a result, the suggested method enhances congestion control performance while using less energy and a recently developed strategy is suggested to successfully enhance network performance. The proposed ButPCNN gives 25 percent improvement to optimize traffic on overloaded node than the other traditional approaches.

Keywords: Wireless sensor networks (WSNs); Congestion Control; Bee Optimization; Butterfly Optimization and BatFuzzyBee.

ARTICLE INFORMATION
Author(s): S. Panimalar and Dr. T. Prem Jacob;
Received: 23/04/2023; **Accepted**: 22/05/2023; **Published**: 30/05/2023;
e-ISSN: 2347-470X;
Paper Id: IJEER-2304-10;
Citation: 10.37391/IJEER.110210
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-11/ijeer-110210.html

Crossref member CROSSREF.ORG THE CITATION LINKING BACKBONE

Publisher's Note: FOREX Publication stays neutral with regard to Jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

More demands are made of the transmission bandwidth, delay, and reliability of communication networks due to the rapidly evolving nature of wireless sensor network technology and the increasing complexity of the data that sensor networks transmit. Large-scale burst data flows to one or more sensor nodes will lead to congestion because of the inadequate cache queue length and limited bandwidth of these sensor nodes, preventing the incoming packets from being forwarded in a reasonable fashion and causing them to be discarded [1]. This is because wireless sensor networks are being deployed widely and are based on data-centric for many-to-one communication mode. The most frequent issue with WSNs is congestion, which demands excellent solutions to prevent negative impacts on network QoS measures. In order to finish a task in a network, the data must be processed before being transferred to its destination; when the space is occupied, congestion happens. The causes of data loss are not limited to software exploitation, hardware failure, protocol usage, low signal amplitude at the endpoint, overloaded network nodes, human or natural incursion, or unnecessary noise [2]. As data is supplied simultaneously by several sensor nodes, intermediary nodes create more data. Due to congestion, the networks become underperforming as a result of the amount of data packets and produced information coming from the source nodes exceeding the network queue's underlying capacity. Congested networks in WSNs can occur for a variety of reasons, such as queue overloading, packet collisions, sharing bandwidth through simultaneous data packet transfers, etc. In these circumstances, network congestion can easily happen, particularly at the locations where data is collected and forwarded in a many-to-one data forwarding network like WSN. Additionally, there is an increase in wireless sensor network congestion.

The service quality and lifespan of the network are significantly impacted by congestion in wireless sensor networks, which causes the network to repeatedly discard incoming packets in transmission, lengthen transmission delays, decrease network throughput, and even cause some serious network loss of function [3]. When there is a backlog of packets at the routers' outgoing queues, performance suffers and there is a poor level of network dependability. As a result, one of the fundamental technologies of sensor networks is congestion control. Several technologies have lately been proposed [4] to handle the issue of traffic congestion control (CC). Congestion in wireless nodes has a significant impact on a number of QoS elements, including power utilization, end-to-end delay, and packet delivery rate. Thus, it is essential to concentrate on the issue of sensor network congestion in order to provide the necessary delivery ratio for WSN applications and to increase the network lifetime [5].

Many CC strategies have been put into practice over the last few years. But the slow start, congestion control, rapid rebroadcast, and rapid resume congestion control strategies all fail to outperform in the current network environment. All sensor nodes provide data to the sink node in the current network scenario, which then collects the data and delivers it to the base station. There is a high likelihood of network congestion when numerous sensor nodes send information to one sink node at the exact same time. The main causes include a restricted amount of available bandwidth and a limited amount of network capacity [6, 21, 22]. The inadequate node possibilities and the characteristics of the wireless channel are the two main causes of congestion in WSN. Secondly, because nodes have a finite amount of memory, a sluggish CPU, and a certain amount of energy, congestion in WSN occurs in nodes. Second, network congestion in WSN occurs as a result of channel interference, reporting rate, event-driven nature of the network, and network nature. So as to maximize the network's lifespan, protocols created for WSNs must be small and scalable. To overcome these problems, an optimal approach to finding a path is needed. Therefore, this work introduced the ButPCNN method. The following are the major contributions of this work:

(i) To develop a Congestion-Free Clustering Formation in the WSN using the ButPCNN approach.
(ii) To introduce Energy Efficient Congestion Free Path Selection Protocol (ECFPSP) to avoid the congestion.
(iii) Using primary and secondary cluster centers and mobile sinks to avoid congestion.
(iv) To packet drop ratio of WSN by designing hop-by-hop and end-to-end level congestion control mechanism.

Another significant contribution is the application of congestion control techniques described in the WSN, with the results analyzed and compared to current methods to demonstrate the suggested method's efficacy.

The remainder of this research paper is structured as follows: *Section 2* delves deeper into and discusses some of the most recent and important research publications that have been used to address the traffic congestion issue. The architecture offered in *Section 3* is described in detail. *Section 4* shows experimental data, which include ButBeeRoute performance results as well as comparison results. The work is completed in *section 5* by leaving room for further research.

1.1 Related Work

To reduce delay while maintaining location privacy and congestion avoidance, the Jellyfish Dynamic Routing Protocol (JDRP) was proposed by Bibin et al [7]. The routing method was used to segment the entire sensor field into several sections. The radial line and virtual ring radius of the network were calculated with in network connection out from node to the sink. JDRP, however, had no effect on how difficult computing was. To indicate the degree of congestion on the basis of evaluation measures, a unique rate-aware congestion management (RACC) technique was developed by Amit el [8]. At specific hotspot sites, RACC improves supply bandwidth management to avoid congestion. The testing made verified that the modulation of the information transmitted to the farther sensor was correct. The RACC approach did not, however, lower the cost of computing. An adjustable cuckoo search-based optimum rate adjustment (ACSRO) was developed by Vaibhav et al [9] for congestion mitigation and management. In order to lessen the likelihood of congestion, the rate modification balanced the node share. Nevertheless implementing a hybrid search approach that included many optimization methods did not effectively lessen congestion. For improve and enhance the data upcoming rate, a Congestion management algorithm built around the optimization technique known as PSOGSA was introduced by Karishma et al [10]. However load harmonizing as well as safety issues remained unresolved. Also, it wasn't appropriate for potential occurrences. Incorporating aggregator nodes (ANs) and normal nodes (NNs) when fine-tuning the time it takes to live (TTL) ratio for false signals and predefined the limit value for original packet counter is how the Mimicking Base-Station (MimiBS) approach was first described by Yawar et al [11]. To achieve scalable, flexible, and managed BS design, software-defined networking models do not use the MimiBS approach. A worldwide congestion control method for IoT-enabled WSNs was presented in [12] to address the congestion issues for smart healthcare. Congestion was prevented in the intended scheme by using a primary concerned data transmitting approach.

A priority queue concerned routing technique was developed to improve dependability. Yet, integrated congestion management did not shorten the wait time. Visualization techniques were employed in [13] to provide a more effective packet modelling strategy across an IoT network. In the optimal zone, the suggested method produced higher throughput and decreased latency. Energy utilization was not considered, even though latency was maintained to a minimum. To choose the best rate for transmitting data while preventing congestion at the relay node, an asymmetric gaming method for congestion control with minimal power consumption was presented by Srijit et al [14]. The non-cooperative game strategy succeeded in lowering energy use but failed to reduce delay.

R & P. Kuppusamy's [15] objective was to decrease the Internet of Things's traffic signal operation times. To accomplish their objective, they had developed a cutting-edge smart system for managing traffic utilizing a wireless networking intelligent server. The vehicle's shift was captured using this technology. The acquired data was subsequently used to follow the elevated automobiles. Before presenting an optimized regression approach, the researchers had gathered information on multiple paths and assessed individual wise choices in relation to the number of autos. Because of the suggested strategy, the authors were able to reduce the working duration of the traffic light. Therefore, the delivery ratio must be increased.

Zhili et al [16] attempted to boost the capacity of the WSN network by tackling the issue of congestion. To do this, the researchers proposed the FSMC fuzzy sliding mode congestion management algorithm. They had initially developed a unique

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 11, Issue 2 | Pages 315-322 | e-ISSN: 2347-470X

cross-layer congestion monitoring system between the MAC and transport levels. Moreover, they were known as fuzzy sliding mode controllers since they combined sliding mode control and fuzzy control. This approach was used to manage the size of the buffer queue on busy nodes. The suggested technique has allowed them to increase throughput. Nonetheless, the accuracy of the suggested model still has to be improved. Azham et al's [17] goal was to effectively handle traffic among the IoT devices while providing a service with minimal delays. They have suggested an adaptive off-loading based on the genetic algorithm known as GA-OA to accomplish this goal. The authors had eliminated requests processing delays by using this offloading strategy. Also, it increased the proportion of IoT requests that were successful. Between the gateways, the GA's fitness performance was distributed. Also, it met all of the communication metrics. By balancing the results of optimal and sub-optimal solutions, the GA increased the request-response rate. They had improved their request success ratio according to the suggested strategy. The writers must concentrate, though, on lengthening the network lifetime. The goal of Swarna et al [18] study was to manage the traffic in WSN-based IoT networks. The scientists used CoAP in their work because it is a successful data protocol for reducing congestion in the Internet of Things. With the low overhead of CoAP, they were able to cut the network's energy usage as well as its memory utilization. The authors had anticipated the congestion control using the effective method. Several margins were employed in the congestion control strategy and were realized via CoAP. They experienced less delay and power usage as a result of the suggested strategy. The network's delivery percentage is, nevertheless, continuing to rise.

In order to reduce traffic congestion, Soulmaz et al [19] sought to improve the high-level intelligence of IoT. The authors therefore had an IoT cognition approach. Cognitive systems, which rely on learning automata, were integrated to the Internet of Things. Then, the authors combined a Learning Automata game with a novel cognitive mechanism they called Cognitive Congestion Control. Learning Automata were applied to all adjustable parameters. The optimal value for each automaton was discovered in order to improve network performance. The article's findings demonstrated that the suggested strategy produced higher throughput and reliability. The delivery of packets must be sped up, nevertheless. The goal of Faisal et al [20] was to enhance the performance of MPTCP-based congestion control, also known as the Multipath Transmission Control Protocol. The proposed strategy helped the authors increase throughput. Nonetheless, the writers must concentrate to improve accuracy and cut down on packet delivery delays. Although the aforementioned research projects produced better outcomes, detection accuracy still has to be improved. Thus, an optimized or upgraded machine learning technique is to be presented in order to increase the performance of congestion control in IoT.

2. PROPOSED WORK

Introduce an Efficient Congestion Free Path Selection Protocol (ECFPSP) for improved network performance in terms of controlling and avoiding the congestion and reducing the packet loss ratio is the main goal of this study. Each cluster in ECFPSP is made up of the principal CH, secondary CH, and member nodes (MNs). Data collection, information dissemination, and network administration fall under the purview of the principal CHs. If the data arrival from the CH is huge and has insufficient buffer capacity, the primary CHs will forward the message to every single MN to forward the packets to the secondary CHs. Formerly, the CH would reroute data transmission to the backup node. The secondary CHs transfer the data to the primary CHs Buffer when the primary CHs Buffer runs out of space. The MNs are in charge of observing occurrences and gathering data about their surroundings. With low overhead, load balancing, stability, durability, scalability, and fault tolerance, the proposed CCR protocol's primary goal is to prevent congestion

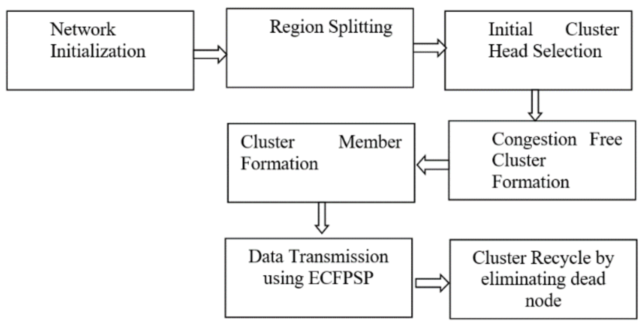

Figure 1: Network Model of Proposed Work

The proposed approach's operational flow is shown in *figure 1*. The steps are

1. System Design
2. Congestion Aware Clusters Building Phase
3. Efficient Congestion Free Path Selection Protocol

2.1 System Design

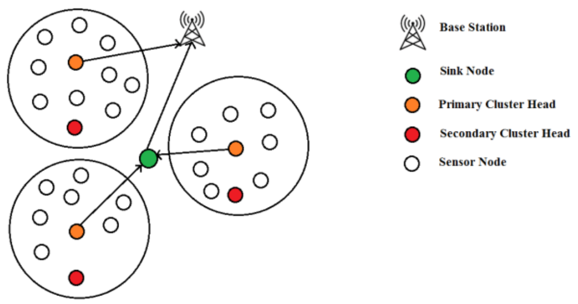

Figure 2: System Model of Proposed Work

2.1.1 Network and Energy Model

The network is situated in the N*N square area of this work, while the base station (BS) is situated outside the monitoring area. As soon as it is deployed, the Base Station (BS) is fixed. All nodes are conscious of the base station's position, and it is presumed that the base station has limitless energy. Once put in the network at arbitrary, all nodes are static. All nodes have the same beginning energy and connectivity range. It is believed that the nodes that can communicate with one another are neighboring nodes. The sink node, which is a moving node and

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 11, Issue 2 | Pages 315-322 | e-ISSN: 2347-470X

has the high amount of power rather than of all the WSN nodes, is always located outside the monitoring area. E stands for all sensor nodes' preliminary energy for I = 1, 2, or 3 plus n. the average remaining energy, or E_S. ($E_S - E$) = 0, Li is the delay, as well as q(link) represent the number of hop. The energy level decreases each time a packet of data is transmitted or received from a source node (Vr) to a sink node (Vn). Ep and E_o stand for the energy used to send and receive k bits of information from Vr to Vn, respectively.

$Ep(Vr, Vn) = k (Ew + E_{amp} * v2)$
$E_o(Vr, Vn) = k (Ew)$

The energy used to transmit and receive one bit is measured as Ep and E_o, respectively. The energy used for amplification is known as Eamp, and the energy used for transmitting and receiving data is known as Ew. Vr and Vn are separated by v miles.

2.1.2 Queue Model and Its Operation
Each sensor node's queue architecture is crucial to the timely delivery of packets. There is a backlog for packets that need to be transmitted on each sensor node. When a node receives more packets than it can broadcast, it overflows its queue. Congestion control procedures can improve throughput at the BS by reducing the information rate, deleting packets that are of lesser priority voluntarily, or redirection packets through an alternative path. Each queue's bounds are determined by its reduced level QRL and optimal threshold QOT. The queue can go between its three states of acceptance, filtering, and rejection more easily as a result. Whenever the queue dimension is lower than the QRL, the queue is considered as selected, and then each and every packets are queued.

2.2 Congestion Aware Clusters Building Phase
Fixed static clusters or cells based on the split of the N*N square sensing field are only produced once during the initialization stage, at the start of communication network. The amount of cluster heads created will directly depend on how the base station divides the monitoring area into k equal subspaces or cells, where k is defined as (N/u) and u is calculated as (N/4), (N/6), (N/8), etc. depending on the sensing requirement. For instance, if u is set to 25, precisely k=4 static clusters will develop.

2.2.1 Cluster Head Election using ButPCNN
In this phase, the cluster head is selected for each group. This selection procedure is based on the weight of each node. The weight of for all nodes is found by using three factors such as distance (1-hop) between particular node and its neighbors, the battery power of the node and its mobility. The cluster head can communicate better with its neighbors at close distances (1 hop) than it is in the communication area. As the sink node moves left from the cluster head, broadcasting can be hard, mainly because of the attention of the signal with increasing distance. Discover all nodes' neighbors' v (i.e., nodes in one of its 1 hop forwarding ranges) that defines the degree of deviation and is calculated using the following formula:

$$d_v = |N(v)| = \sum_{v' \in V,\, v' \neq v} \{\mathrm{dist}(v, v') < tx_{\mathrm{range}}\}. \quad (1)$$

Battery power may be utilized effectively within a certain communication range; for example, transmitting between two nodes that are close to one another will use less power. Cluster heads use more battery power than normal nodes since cluster heads have more responsibility to perform for their members. The below formula is employed for determining each node's battery power.

pv = Initial Energy - Consumed Energy (2)

Density is an important decision factor for cluster heads. To avoid frequent changes of cluster heads, it is desirable to choose cluster heads that are very close to all other sensor nodes. So, the density of each node is Dv is calculated by using below formula

$$M_v = \frac{1}{T}\sum_{t=1}^{T}\sqrt{(X_t - X_{t-1})^2 + (Y_t - Y_{t-1})^2}, \quad (3)$$

The positions of the node v at period interval and (t 1), respectively, are (Xt, Yt) and (Xt1, Yt1), including both. Finally, the weight value W of all node is measured by using the *equation (4)*

Finally, using the *formula (4)*, each node's fitness value Wv is computed as

Wv = dv + pv +Mv (4)

Algorithm 1: Cluster head selection using ButPCNN

1. Derive the fitness function by utilizing equation (4).
2. Generate an initial population of butterflies.
3. Set the stimulus intensity I to a specific value.
4. Set the switch probability, sensor modality, and power exponent to their initial values.
5. **For each** iteration (j) up to a maximum number:
6. **For each** butterfly in the population:
7. Calculate the fragrance using equation (4).
8. **End loop**.
9. Identify the population's optimal butterfly.
10. **For each** butterfly in the population:
11. Generate a random number between 0 and 1.
12. **If** the random number is less than or equal to the switch probability:
13. Move the butterfly towards the optimal solution.
14. **Otherwise**, move the butterfly randomly.
15. **End loop**.
16. Update the power exponent's value.
17. **End loop**.
18. Obtain the optimal solution and provide it as input to the PCNN.
19. Output the CH from the network, which represents the best optimal solution.

2.2.2 Cluster Formation
The novel rule is used to determine how clusters are created. Every node within the transmission range of the elected CHs receives a message from the CH. Non-CHs receive alerts from CHs, and they decide which CH to follow based on the source with the clearest signal. The node selects the CH with the lowest node ID if two sources offer the same amount of power.

2.2.3 Scheduling
During clustering process and CH-election, the CH generates an ECFPSP schedule for its member nodes, which identifies designated specific times for every node in which they're able to transfer information. This ECFPSP scheduling prevents collisions in all intra-cluster transmission. To reduce energy usage, it has been made sure that member nodes' radios are off aside from when they are transmitting.

2.2.4 Data Communication phase
The three basic procedures in the data communication phase are information gathering, data fusion, and information routing. According to the ECFPSP schedule, each sensor node transmits the data it has collected to its CH because sensor nodes and cluster heads are so close to one another in space and because sensor nodes build clusters, these interactions need very little energy. After receiving the data from each and every member node, the CH executes packet fusion on the gathered data. As a result, the amount of raw data sent to the BS is reduced. The CH transmits the data packets in addition to the data that the BS needs for cluster data authentication using the CH-to-CH routing link that the BS offers. The primary CH sends the packets to the mobile Sink (SN). The data is sent from the mobile SN to the base station. If the data arrival from the CH is huge and has insufficient buffer capacity, the primary CHs will forward the message to every single MN to forward the packets to the secondary CHs. Formerly, the CH would reroute data transmission to the backup node. The secondary CHs transfer the data to the primary CHs Buffer when the primary CHs Buffer runs out of space.

3. EFFICIENT CONGESTION FREE PATH SELECTION PROTOCOL (ECFPSP)
As a cluster is established, the data transmission phase starts, during which each sensor node forward its acquired packets to the CH in accordance with its own ECFPSP schedule. After performing data aggregation, the CH will use mobile sink node to send the data packets across well-defined multi-hop channels to the base station or sink. On the basis of the two conditions listed below, the cluster head will decide whether to forward its packets straight to the BS or to a SN. Here, DCH To DBS, DCH To DMS, and DMS To DBS all represent the distances between the CH and the BS, the SN, and the BS, respectively.

(1) To avoid congestion, data collected from the CH will be sent to the BS via a mobile SN if DCH To DBS = > (DCH To DMS + DMS To DBS).
(2) To avoid congestion, the CH will transfer the collected data packets straight to the BS without requiring a mobile sink node if DCH To DBS = (DCH To DMS + DMS To DBS).

In ECFPSP, there are two stages of congestion computation before data transmission
1. intra cluster level
2. inter cluster level.

3.1 Intra Cluster Level Congestion Detection
At the intra cluster stage, constant input is used to identify and regulate hop-by-hop congestion. Whenever a CM "x" wanted to transfer data to CH via an intermediary CM "y," the update table of the node CM "x" would receive the present queue size QS and the remaining power PR of the node y. The status table that is shown contrasts with it. The status value is calculated using the remaining energy ER and the queue length QL as of the moment. Node X would transmit a message to Node Y depending on the current level; if not, Node X could try for another neighbor to communicate with or would just drop communications. Linear feedback is the mechanism by which nodes actively monitor the queue length and remaining energy of neighboring nodes, which information is doubled down in the header of the information packet. Binary feedback was not taken into consideration at the intra cluster level in order to decrease the control overhead associated with sending packets. The minimum queue limit MQL and maximum queue limit MAQL are bound to each node's queue and are used to determine the level of congestion.

3.2 Inter Cluster Level Congestion Detection
At the inter cluster stage, constant input is used to identify and regulate end-to-end congestion. The CH uses multi-hop connection, which can include a congested CH, to send all aggregated packets to the base station. In the ECFPSP protocol, the sensor node examines the present queue size and remaining power of CH "y" in its route cache table when CH "x" needs to interact with BS by intermediary CH "y".

Node X would deliver the packet to node Y if indeed the left's present queue size does not exceed the stated limit and had the best remaining power. If not, sensor node X alters the control packet's routing to CH or removes some packets after inserting a congestion indication. Congestion is visible when a network consistently loses packets. Queue overflow is the primary cause of packet drops. Controlling the queue exploding in a constructed network is challenging since the queue size is severely constrained. Most often, congestion is connected to each node's queuing model.

3.3 Congestion Free Energy Efficient Path Selection
This section describes the route discovery within the cluster and between the clusters. Here we employ the ECFPSP algorithm to find the congestion free energy efficient path or finest path from the given source to the destination. According to ECFPSP algorithm the residual energy value and congestion level gets updated by CM and CHs on their traversal to the destination and on their way rear to the source. When a source node want to send communication to the destination, it fist look up the IntraCluster Routing Table (IntraRT) if the destination lies inside the cluster. If it finds the destination in IntraRT, then the

FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 11, Issue 2 | Pages 315-322 | e-ISSN: 2347-470X

path discovery process was competed. IntraCluster Routing Table contains the data about the nodes inside the cluster and InterCluster Routing Table contains the information about the gateway nodes. If the destination node stays inside the cluster we always have a path because the Intra Routing Table was proactively maintained by forwarding the internal ant inside the cluster periodically. If the node lies outside the cluster we employ the InterCluster Routing Table (InterRT) for route discovery process. Initially the source wants to send data packet to the destination which lies outside the cluster first we want to look up the InterRT. If the path is already discovered by the previous communication then the route is stored in the InterRT table. Then the source node immediately sends the data packet along the path, if the link doesn't expired. If there is no route to the destination we forward the external ant to search the route between the clusters. The external forward ant initially sends by the node to meet its gateway nodes since the InterRT of the gateway node contains the data about the nodes of neighboring clusters. This process continued until the destination node is found.

4. PERFORMANCE ANALYSIS

NS2 was used to simulate the experiment in order to evaluate the suggested ECFPSP algorithm. An average of numerous experiments is used to represent the final results. The simulation's area is 100 by 100 square metres while it is being run. Then, a few wireless sensor nodes are dispersed at random around that region. The self-gathered data is then sent from the node to the sink in the simulation area's centre. We made the assumption that all nodes produced data at the same rate in order to make the simulation easier. Tab. 1 displays the exact simulation parameters. We modified a few parameter values during the simulation in accordance with our various needs.

Table 1: Simulation Parameters & its Measurements

PARAMETERS	MEASUREMENTS
Number of Nodes	50 - 300
Area Size	100m x 100m
Base Station Coordinate	50,50
Communication Range	30 m
Cache queue length	50 packets
Packet Size	1024 bits
Communication Range	30 m
Buffer Size	20 packets
Data rate	4096 bit/round
Traffic Pattern	Constant Bit Rate
Simulation Time	400 rounds

We used the following three route equation metrics in the performance analysis:

(1) The Packet Loss Rate: This indication shows how many packets are lost as a percentage of all the packets sent by the output node. Every packet's final stop is the WSN receiver node. As a result, this indication shows how trustworthy the data distribution.

(2) The Average Hops: The average number of nodes that the output node encounters when transmitting packets to the receiver is represented by this indicator, which can display the network's real-time performance.

(3) The Average Energy Consumption: The average power used per cycle across the network is shown by this indication. A circle in our simulation indicates that the sensor node forwarded the data packet to the following hop. This metric reveals how energy-efficient the data transport was.

4.1 The Packet Loss Rate

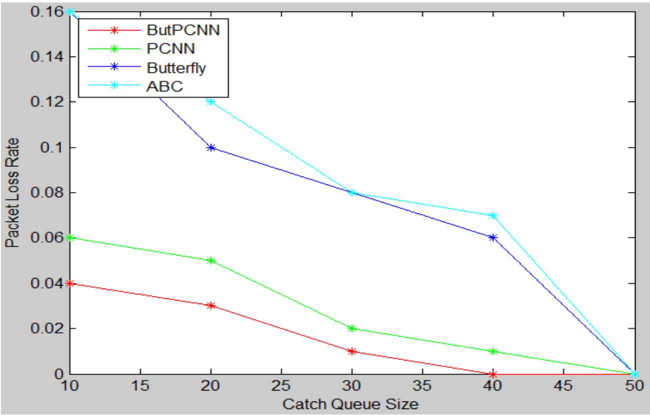

Figure 3: Evaluation Output of Packet Loss Rate

The proposed ButPCNN approach's packet loss rate is compared with existing approaches is illustrated in Fig.3. The packet loss ratio of ButPCNN has low value than existing approaches. So it is proofed that, the congestion controlling behavior of ButPCNN is highest than existing approaches. ButPCNN gives 5 percent improvement to optimize traffic on normal node. ButPCNN gives 25 percent improvement to optimize traffic on overloaded node. When the queue task is increased ButPCNN gives 1.5 percent improvement in packet loss rate.

4.2 The Average Hops

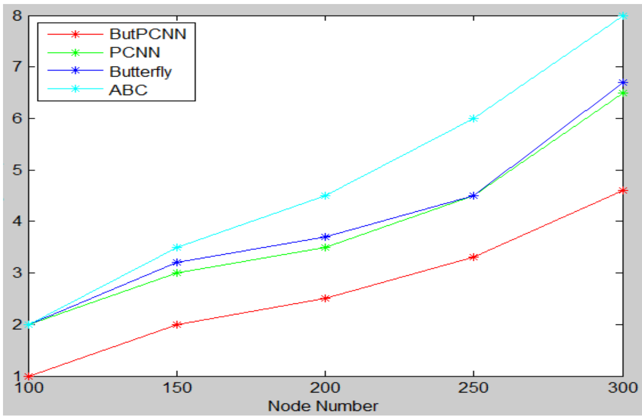

Figure 4: Evaluation of Average Hops

The proposed ButPCNN approach's average hops rate is compared with existing approaches is illustrated in Fig.4. The average hops rate of ButPCNN has low value than existing approaches. So it is proofed that, the congestion controlling behavior of ButPCNN is highest than existing approaches

4.3 The Average Energy Consumption

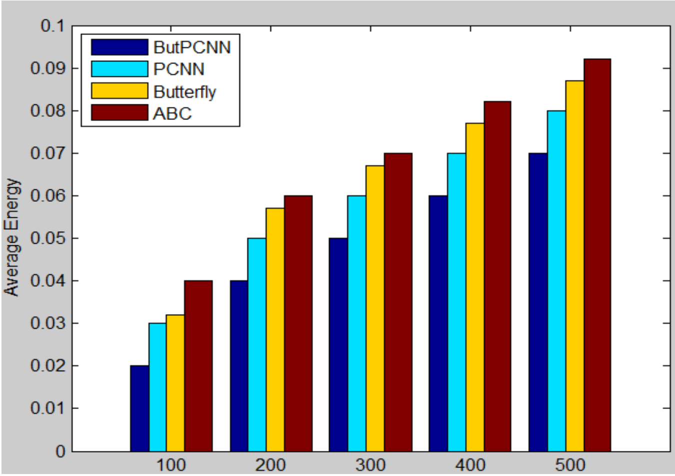

Figure 5: Assessment of Average Energy Consumption

The graph represents the average energy consumption for different algorithms, where each column corresponds to a different algorithm (ButPCNN, PCNN, Butterfly, and ABC) and each row corresponds to a different trial or data point. The average energy consumption values for all algorithms are relatively low, ranging from 0.02 to 0.092. The ButPCNN algorithm has the lowest average energy consumption values across all trials. The proposed ButPCNN approach's average energy consumption rate is compared with existing approaches is illustrated in Fig.5. The energy consumption rate of ButPCNN has low value than existing approaches. So it is proofed that, the congestion controlling behavior of ButPCNN is highest than existing approaches.

5. CONCLUSION

An important area of concern for WSNs is congestion control. Congestion in the network occurs when the number of arriving packets exceeds the network's or a node's actual capacity. Network congestion can result in decreased throughput, increased network delay, increased packet loss, and increased sensor energy consumption. As a result, this study recommends an energy-efficient routing technique to lessen network congestion. In this research, a unique technique for reducing clustering and congestion in WSNs was developed that ECFPSP protocol. It decrease end-to-end delay time and extend the network lifetime. The suggested protocol implements a routing protocol that provides security by avoiding hostile nodes and reducing data loss. It also routes the nodes. Hence, a Congestion-Free Cluster Formation is provided to increase the lifetime of the network by proposed ButPCNN approach. Experimental results show that the ECFPSP algorithm works better than other traditional methods. But BeeRoute achieves the highest life expectancy due to reduced power and packet loss by eliminating congestion. Finally, the ECFPSP algorithm is used to identify and analyze any changes in the transmission model of the actual WSN network. Furthermore demonstrated is the suggested protocol's stability as the network area expands. In future, deep learning based Congestion prediction model will be used.

REFERENCES

[1] Almalkawi I T, Guerrero Z M, Al-Karaki J N and Morillo-Pozo J 2010 Wireless multimedia sensor networks: current trends and future directions Sensors. 10(7) 6662-717.

[2] Osuo-Genseleke, M., Kabari, L. and Nathaniel, O., 2018. Performance measures for congestion control techniques in a wireless sensor network. International Journal of Scientific and Research Publications, 7, pp.1-5.

[3] Trappe W, Ning P and Perrig A 2010 Security and privacy for sensor networks Handbook on Array Processing and Sensor Networks. 855-87.

[4] Farahani, S.S.S. and Fakhimi Derakhshan, S., 2019. LMI-based congestion control algorithms for a delayed network. International Journal of Industrial Electronics Control and Optimization, 2(2), pp.91-98.

[5] Singh, K., Singh, K. and Aziz, A., 2018. Congestion control in wireless sensor networks by hybrid multi-objective optimization algorithm. Computer Networks, 138, pp.90-107.

[6] K. Singh, K. Singh, L. H. Son, and A. Aziz, ''Congestion control in wireless sensor networks by hybrid multi-objective optimization algorithm,'' Comput. Netw., vol. 138, pp. 90–107, Jun. 2018.

[7] V. Bibin Christopher and J. Jasper, "Jellyfish dynamic routing protocol with mobile sink for location privacy and congestion avoidance in wireless sensor networks", Journal of Systems Architecture, Elsevier, Volume 112, January 2021, Pages 1-14

[8] Amit Grover, R. Mohan Kumar, Mohit Angurala, Mehtab Singh, Anu Sheetal and R. Maheswar, "Rate aware congestion control mechanism for wireless sensor networks", Alexandria Engineering Journal, Elsevier, Volume 61, Issue 6, June 2022, Pages 4765-4777

[9] Vaibhav Narawade and Uttam D. Kolekar, "ACSRO: Adaptive cuckoo search based rate adjustment for optimized congestion avoidance and control in wireless sensor networks", Alexandria Engineering Journal, Elsevier, Volume 57, 2018, Pages 131–145

[10] Karishma Singh, Karan Singh, Le Hoang Son and Ahmed Aziz, "Congestion Control in Wireless Sensor Networks by Hybrid Multi-Objective Optimization Algorithm", Computer Networks, Elsevier, Volume 138, 19 June 2018, Pages 90-107

[11] Yawar Abbas Bangash, Ling-Fang Zeng and Dan Feng, "MimiBS: Mimicking Base-Station to Provide Location Privacy Protection in Wireless Sensor Networks", Journal of Computer Science and Technology, Springer, Volume 32, Issue 5, September 2017, Pages 991–1007

[12] Prasenjit Chanak and Indrajit Banerjee, "Congestion Free Routing Mechanism for IoT- Enabled Wireless Sensor Networks for Smart Healthcare Applications", IEEE Transactions on Consumer Electronics, Volume 66, Issue 3, August 2020, Pages 223 – 232

[13] Daniyal Alghazzawi, Omaima Bamasaq, Surbhi Bhatia, Ankit Kumar, Pankaj Dadheech and Aiiad Albeshri, "Congestion Control in Cognitive IoT-Based WSN Network for Smart Agriculture", IEEE Access, Volume 9, November 2021, Pages 151401 - 151420

[14] Srijit Chowdhury, Abderrahim Benslimane and Chandan Giri, "Noncooperative Gaming for EnergyEfficient Congestion Control in 6LoWPAN", IEEE Internet of Things Journal, Volume 7, Issue 6, June 2020, Pages 4777 – 4788

[15] P. Kuppusamy, R. Kalpana and P. Venkateswara Rao, "Optimized traffic control and data processing using IoT", Cluster Computing, vol. 22, no. 1, pp. 2169-2178, 2018.

[16] S. Qu, L. Zhao and Z. Xiong, "Cross-layer congestion control of wireless sensor networks based on fuzzy sliding mode control", Neural Computing and Applications, vol. 32, no. 17, pp. 13505-13520, 2020.

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 11, Issue 2 | Pages 315-322 | e-ISSN: 2347-470X

[17] A. Hussain, S. Manikanthan, T. Padmapriya and M. Nagalingam, "Genetic algorithm based adaptive offloading for improving IoT device communication efficiency", Wireless Networks, vol. 26, no. 4, pp. 2329-2338, 2019.

[18] M. Swarna and T. Godhavari, "Enhancement of CoAP based congestion control in IoT network - a novel approach", Materials Today: Proceedings, vol. 37, pp. 775-784, 2021.

[19] S. Gheisari and E. Tahavori, "CCCLA: A cognitive approach for congestion control in Internet of Things using a game of learning automata", Computer Communications, vol. 147, pp. 40-49, 2019.

[20] F. Naeem, G. Srivastava and M. Tariq, "A Software Defined Network Based Fuzzy Normalized Neural Adaptive Multipath Congestion Control for the Internet of Things", IEEE Transactions on Network Science and Engineering, vol. 7, no. 4, pp. 2155-2164, 2020.

[21] T. Akhtar, N. G. Haider, and S. M. Khan, "A Comparative Study of the Application of Glowworm Swarm Optimization Algorithm with other Nature-Inspired Algorithms in the Network Load Balancing Problem", Eng. Technol. Appl. Sci. Res., vol. 12, no. 4, pp. 8777–8784, Aug. 2022.

[22] M. A. Mahdi, T. C. Wan, A. Mahdi, M. A. G. Hazber, and B. A. Mohammed, "A Multipath Cluster-Based Routing Protocol For Mobile Ad Hoc Networks", Eng. Technol. Appl. Sci. Res., vol. 11, no. 5, pp. 7635–7640, Oct. 2021.