Grenze International Journal of Engineering and Technology, January Issue

Random Early Detection Protocol for Congestion Prediction and Avoidance in MANET under TCP

Sanjeev Mahajan[1] and Dr. Jagdeep Kaur[2]
[1]Research Scholar, SBBSU Jalandhar, India
Email: sanjeevmahajan@gmail.com
[2] Professor, SBBSU Jalandhar, India
Email: sbbsuuietcse@gmail.com

***Abstract*— Transmission Control Protocol (TCP) ensures data transfer reliability for all internet end-to-end data stream services. The information will be transferred throughout the Mobile Ad Hoc Network (MANET). Data transmission becomes unusual when there is congestion in the path of the data while being transferred. If congestion occurs packet loss got occurred and precision will also be in the slanting state. To solve this issues, Artificial Neural Network (ANN) through Random Early Detection Protocol (RED) is developed for prediction also for avoiding the congestion flow in the path of data transmission. Initially the data's will be transferred between the networks from the source to the destination. As a data is initiated from the transmission side and the other side of the receiver will be waiting for the acknowledgement. As soon as the acknowledgement got received, the function of congestion prediction will be initiated. This congestion prediction will be held by the ANN for transforming data transmission into an efficient data transmission. Furtherly, Golden Jackal Optimization (GJO) is used for selecting the optimal weights present in the ANN classifier. After the identification of the congestion is existed then it will be avoided using the REDP. The simulation analysis shows the suggested method gives improved performance of TCP in MANET.**

***Index Terms*— Artificial Neural Network, Congestion, Golden Jackal Optimization, Mobile Ad Hoc Network, Transmission Control Protocol.**

I. INTRODUCTION

Transmission Control Protocol (TCP) uses acknowledgments (ACK) to ensure reliability, for every TCP packet provided and it waits for an ACK. Based on few years preceding, the computing sector has supported a significant increase in both user numbers and traffic volume [1]. The autonomous mobile nodes that makeup Mobile Ad-hoc Networks (MANETs) have the ability to join and exit the network at random. These nodes establish wireless connections with one another to exchange data. Ad hoc makes it possible to quickly add new devices [2]. Due to the freedom of movement of network devices, a dynamic topology is produced. The design of a network involves many challenges and issues, which makes it a very tough task. Each node has the capacity to act as a router, transmitting packets from one place to another. In a network environment, multiple devices keep an eye out for imminent congestion before it arises [3]. In a conventional wired network context, the TCP is a transport layer protocol that offers dependable end-to-end data delivery between end hosts. By resending neglected packets, TCP accuracy is achieved [4]. As a result, every TCP sender keeps track of the projected return delay and the average deviation derived from the source itself.

Grenze ID: 01.GIJET.11.1.231_1

If the sender doesn't receive an acknowledgment within a predetermined timeout period or receives multiple acknowledgments, the packets will be resent [5]. Heavy traffic results from an increase in users since the communication between the network and the consumer is hampered by this excessive traffic. Internet-based applications require "Transmission Control Protocol" (TCP) to a significant extent [6]. Congestion, delay, and loss have been brought on by a sharp rise in traffic transmission. When efficiency drops owing to an increase in links and the quality of service (QOS) is supplied, for instance a network is considered to be congested. Therefore, it is important to continue to maintain a check on congestion to avoid performance decline and network failure. [7]. Furthermore, undesirable activity on the network could result in congestion. Since it emphasizes on prolonging the network's lifespan, hence the managing congestion is crucial. MANET, primarily holds the TCP and performs essential roles in network operations, is essential. [8].

Elimination of congestion techniques can be employed with competent network administration to establish a long-lasting network. A network must have such capacity in order to minimize congestion and losses. However, contemporary cloud applications, social media platforms, and other high-tech multimedia applications demand quick upload rates with low latency, requiring for networks to be improved progressively. [9]. Due to the heterogeneity of networks and data, a rapid rise in network traffic and internet users, congestion is currently becoming more and more common in both academic research and the industrial sector. [10].Artificial intelligence (AI) has been used as the primary method of regulating every aspect of the data. As stated, numerous techniques for removing negative attributes from data have been produced, some of them are also used to minimize the term congestion. [11].

Techniques like MLP, DBN, and XG Boost are among the features that have been implemented or are present in the current phase. These currently used techniques have serious shortcomings when it comes to delivering accurate data. Even if the data that is not been in a precise ratio, then there will be an enormous disparity between the precise data and the data that has been gathered from the network [12]. Along with implementing a specific technique which has been used here currently, ensures that data from the sender side of the network tends to reach the receiver side without any responsibilities [13]. Since to negotiate this major issue, an optimized neural deep learning network to transform the data into a much more accruable one in order to minimize the constraints and adverse conditions in the existing methodologies[14]. The major contributions of this proposed work are as listed in the below listings:

- Random Early Detection Protocol (REDP) with Artificial Neural Network (ANN) is developed to enhance the performance of TCP.
- Congestion which got occurred in the data transmission segment needs to be eliminated from the data transmission path using ANN based on some parameters.
- Golden Jackal Optimization (GJO) is used to optimally select the weight of the ANN classifier to enhance its performance.
- Packet Chaining Reservation Protocol (PCRP) is developed to avoid congestion during data transmission which improve the performance of TCP.

The upcoming parts that follow are organized as into the following categories: Section 2 includes the articles which are the current systems based on the betterment of the TCP performance based on the MANET scheme. Additionally, the recommended methodologies are characterized in Section 3. The determination of the suggested strategy is presented in Section 4 along with graphical representations. The entire determined work with the conclusion, including the reference component, is contained in Section 5. The study is divided into the following sections: Section 2 talks about a lot of research papers related to the TCP under MANET routing protocol that is currently in use. Section 3 gives a brief summary of the suggested methodology. Section 4 presents the results and performance evaluations of the proposed framework. Section 5 of the report concludes the complete research project.

II. Literature Review

In MANET, a variety of routing algorithms are employed to enhance the efficiency of the TCP traffic pattern. The analysis and review that follows looks at a few of the recent approaches.

According to Wang et al. [15] have developed congestion control approach which is used to optimize the performance of data transfer across the various supercomputing devices.

There will be a lack of issue for transferring data’s within the device to the receiver side and also the acknowledgment of the data will also be a bit hard for capturing. The agent will be built to interact with the network environment which is provided. For an ideal result, the Q-learning algorithm is too big. While the process went the oscillation issue would occasionally arise.

Su et al. [16] illustrates grid topology networks used for TCP congestion control. As early known congestion is a defect that gets interrupted while the data's gets transferred from the client to the server. While travelling like this there will be some interference on the network line to eradicate this phenomenal issue this idea has been proposed. Large RTT variations are caused by the ContikiMAC due to hidden terminal issues, and LLN endpoints may experience TCP fairness issues as a result in order to reduce congestion fields.

Rad et al. [17] have presented the SRMCC mechanism in this paper's information-centric networking strategies. This method's primary goal is to minimize multicast congestion, which develops in the data transmission field. The main drawback of this kind is that multicast does not offer congestion control, and the procedures are also rather sophisticated and necessitate frequent input.

Chatterjee et al. [18] have introduced prediction ways to spot out the congestion and to avoid those congestions are as listed. The C-TCP has a detrimental impact on fairness, which was the most significant disadvantage.

Lan et al. [19] have developed a deep reinforcement learning-based congestion control strategy for DND was described. In order to construct an effective congestion control mechanism, the newly created entirely architecture method combines numerous obstacles.

Lim [20] have stated the TCP congestion management optimization strategies for restricted Internet of Things networks. This has negative consequences that the network cannot be overloaded, and the end-to-end congestion control mechanism cannot cooperate with the networks.

The above reviews have many existing techniques which hold a lot of disadvantages and drawbacks. The review paper [15] states that the acknowledgement capturing phase seems to be a bit hard segment. The [16] has the drawback which states that transferring the data's itself seemed to be hard within the networks.[17] states the procedure of the review itself seemed to be difficult to evacuate the congestion and also the for the prediction. The accuracy fairness will not be in expected state say the [18]. A numerous obstacles are been inheriting the way of the prediction and the avoidance phase of congestion which makes the whole procedure into a difficult form which is stated in the [19] & [20] states the network won't be overloaded and also the congestion control mechanisms was not in a proper manner of functioning.

III. Proposed Methodology

The proposed work of TCP implements analytical framework for the cross-layer evaluation and performance development of TCP connections and also to condense the internal combinations of the surreal or a disturbances kind of themes that affect the whole traffic patterns. In order to eradicate these kind of major issues this MANET which includes the deep learning technology will be induced into the problem solving issue. Even though there are several existing techniques are available which has the major drawbacks like the procedure of algorithms will be too complicated, there will be large variation in the data values on account of these drawbacks the TCP performance gets a major slide of accuracy. To eliminate these kind of issues the ANN based congestion prediction is incorporated here, based on this the performance based issues will get eradicated and the further performance will be improved. Figure 1 below depicts the proposed system's architecture

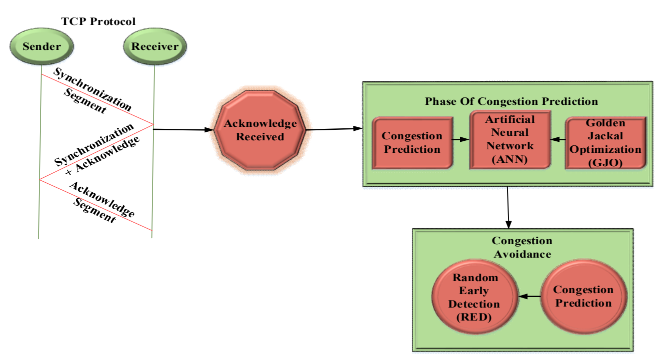

Figure 1. Architecture of the proposed system.

A. Data Transmission

TCP enables data interchange between software and hardware through a network, it is employed in the TCP IP paradigm. To guarantee that communication reaches its target effectively and as soon as possible, a message is designed to be divided into data packets. And continuously the data transmission will be transferred simultaneously if no other issues got occurred.

B. Congestion Prediction

The initial information is transmitted by the transmitter, who then waits for a reply. The congestion prediction algorithm will be employed if the acknowledgement is received in order to determine whether there is any congestion in the data transmission line. This prediction of congestion is eradicated through the ANN.

C. Artificial Neural Network

There have been numerous advancements in the creation of intelligent systems, some of which were influenced by biological neural networks. Artificial neural networks are computer programs whose main idea is modelled after biological brain networks. Neural Nets, connectionist systems, and concurrent distributed processing systems are various other titles for artificial neural networks. A computing system must have a labelled directed graph structure with nodes that carry out certain basic operations in order for it to go by these lovely titles. The use of the ANN would be advantageous for many applications. They offer interesting frequently parallel computing systems like SO and ANN are made up of a huge number of basic processors interconnected through several links. Artificial Neural Networks are modelled closely following the brain and therefore a great deal of terminology is borrowed from neuroscience [21]. The structural diagram of the ANN is shown in below figure 2.

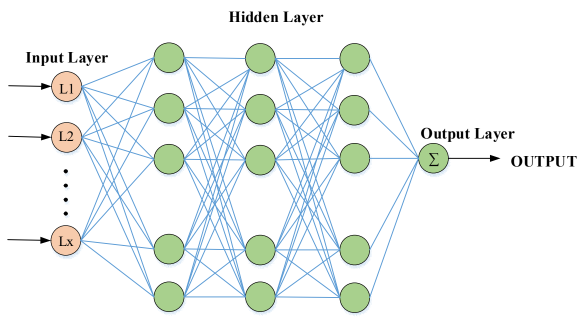

Figure 2. Architecture of Artificial Neural Network.

C. Backpropagation Algorithm

Layered feed-forward ANNs employ the backpropagation algorithm. As a result of this, the signals transmitted by the artificial neurons, which have been organized in layers, are sent forward whereas the errors are transmitted reverse. Neurons in the network's output layer provide the network's output, and those in the network's input layer give inputs. One or more hidden intermediary levels might exist. As supervised learning is used in backpropagation, the method has to be provided a sample of the inputs and outputs that the user want the network to determine before it can recognize the error. In order to assist the ANN recognize the training data, the backpropagation technique is designed to reduce this error. Starting with random weights, the training process aims to modify them to minimize error.

In artificial neurons that employ the backpropagation method, the activation function is a weighted sum (the inputs' aggregate multiplied by the appropriate weights,w_{pq}).

$$N_p(\bar{a},\bar{b}) = \sum_{j=0}^{k} a_j\, b_{pj} \tag{1}$$

The inputs and weights are the sole factors upon which the activation depends. The neuron would also be regarded as linear if the output function remained equivalent with the activation function (output=activation). However, these have significant drawbacks in the following areas; the sigmoidal function is the most common output function.

$$X_p(\bar{a},\bar{b}) = \frac{1}{1+e^{N((\bar{a},\bar{b})}} \tag{2}$$

When large positive numbers approach one, the sigmoidal function becomes 0.5 at zero, and when large negative numbers approach zero, it approaches zero. In order to create what you want given a set of inputs, a training procedure is used. In order to lower the error, the weights must be adjusted because the error is the discrepancy among the output that occurs in reality and what is anticipated. For the output of each neuron, we can define the error function.

$$E_j(\bar{a}.\bar{b},\bar{q}) = (X_p(\bar{a},\bar{b}) - q_p)^2 \tag{3}$$

Since the square of the difference between the output and the intended target is always positive and gets smaller if the difference is small, to calculate the square of the difference between the output and the target. The network's error will simply be the total of all the errors of each neuron in the output layer:

$$E(\bar{a}, \bar{b}, \bar{q}) = \sum_p (X_p(\bar{a}, \bar{b}) - q_p)^2 \tag{4}$$

The error's connection to the output, inputs, and weights is now computed via the backpropagation process. Once this has been identified, the weights can be modified with the gradient descendent approach.

$$\Delta b_{pj} = -\eta \frac{\partial D}{\partial b_{pj}} \tag{5}$$

Each weight of E with respect to b, or $pj\ (b)$, should be derived as the negative of a constant (0) multiplied by the weights previous depending on the network error. The weight's contribution to the function's inaccuracy and the value of 0 will determine the adjustment's magnitude. Furtherly the calculation of the error which depends upon the output D which is concerning X_p.

$$\frac{\partial D}{\partial X_p} = 2(X_p - Q_p) \tag{6}$$

Finally, evaluate the point at which the weights from (1) and (2) affect the activation process.

$$\frac{\partial X_p}{\partial N_{pj}} = \frac{\partial X_p}{\partial N_p} \frac{\partial N_p}{\partial B_{pj}} = X_p(1 - X_p)a_j \tag{7}$$

And from (6) and (7),

$$\frac{\partial D}{\partial B_{pj}} = \frac{\partial E}{\partial X_p} \frac{\partial X_p}{\partial B_{pj}} = 2(X_p - q_p)X_p(1 - X_p)X_j \tag{8}$$

As a result, each weight will be adjusted as follows (from (5) and (8)):

$$\Delta B_{pj} = -2\eta\ (X_p - q_p)X_p(1 - X_p)A_j \tag{9}$$

The derivation of each weight for training an ANN with two layers, we can use (9) exactly as is. We now need to take various factors into account before training the network with a new layer. Make the necessary calculations first to establish how the mistake depends on the input from the preceding layer rather than the weights of a prior layer. It's simple just swap out x with w in positions (7), (8), and (9). However, also need to look at how the adjustment of affects the network error.

$$\Delta W_{jl} = -\eta \frac{\partial D}{\partial W_{jl}} = -\eta \frac{\partial D}{\partial a_j} \frac{\partial a_j}{\partial W_{jl}} \tag{10}$$

Where,

$$\frac{\partial D}{\partial W_{pj}} = 2(X_p - Q_p)\, X_p(1 - X_p)\, B_{pj} \tag{11}$$

$$\frac{\partial a_j}{\partial w_{jl}} = a_j(1 - a_j)w_{jl} \tag{12}$$

Creating a further layer while performing the same calculation makes it possible to find the errors to connect the inputs and weights of the first layer. Since the time required to train the networks grows exponentially, by implementing the backpropagation algorithm. A rapid learning process is made possible by improvements to the backpropagation algorithm. [20].

D. Golden Jackal Optimization

Chopra and Ansari proposed the GJO algorithm, an optimization method that draws inspiration from nature. GJO provides an alternate sort of optimization technique to address engineering issues that arise in the real world, much like other meta-heuristic algorithms. Golden jackal cooperative hunting habits was used as a template for GJO. To hunt in pairs, the male and female jackals together. Initially, the golden jackal pair hunts by searching for prey, approaching it, encircling and disturbing the prey until it ceases to move and then pounces on it. There are two distinct stages to the social hunting of golden jackals: There are two phases: (a) the exploration phase, which entails looking for, keeping an eye on, and pursuing the target; and (b) the exploitation phase, which entails encircling and taking the target. [22].

Step 1: Initialization

Initialize the weights based on the population of the golden jackals and Equation (13) includes the formulas required to represent the initialization procedure.

$$W = \{W1, W2, W3 \ldots Ww\} \tag{13}$$

Step 2: Fitness function

The fitness function provided for this optimization technique involves packet delivery ratio maximization. The mathematical basis for determining the packet delivery ratio is given by Equation (4).

$$fitness\ function = minimization\ (E) \tag{14}$$

$$\mathrm{E} = \frac{1}{2}\sum_{m=1}^{N}(U_n - V_n)^2 \tag{15}$$

Step 3: Updation

Whenever the value achieves an optimal value, it is adjusted in accordance with the golden jackal optimization behaviours (prey hunting, enclosing, and pouching). The phase known as the exploration phase is when the value of ($|E| >= 1$), while the phase known as the exploitation phase is when the value lies in ($|E| < 1$).

Exploration

GJO is advised to employ its exploratory methodology. The jackal can sense and pursue its prey as is their nature, but sometimes the prey flees or it will be difficult to catch. The jackals wait as a result and search for other prey. Male jackal leads the hunt. Male jackal is pursued by female jackal. Here, the exploration phase is offered by Equations (16) and (17).

$$P_1(t) = P_N(t) - E.|P_N(t) - rl.Prey(t)| \tag{16}$$

$$P_2(t) = P_{FN}(t) - E.|P_{FN}(t) - rl.Prey(t)| \tag{17}$$

In the present instance, t indicates the most recent iteration, Prey (t) is a vector that represents the prey's position, and $P_N(t)$ and$P_{FN}(t)$, respectively, are the locations where the male and female jackals are equally located. Jackal male and female positions that correspond to the prey are updated at $P_1(t)$ and $P_2(t)$. E is Contrary to Prey Energy.

Exploitation

When jackals harass the target, it loses its capacity for escape, and the pair of jackals encloses the prey that was discovered earlier. They round themselves, jump on their prey, and eat it.

$$P_1(t) = P_N(t) - E.|rl.P_N(t) - rl.Prey(t)| \tag{18}$$

$$P_2(t) = P_{FN}(t) - E.|rl.P_{FN}(t) - rl.Prey(t)| \tag{19}$$

Where the current iteration is denoted by t. Prey (t) is the prey's location vector and $P_N(t)$and $P_{FN}(t)$ indicates were the female and male jackals are located. $P_1(t)$ & $P_2(t)$ are current locations of the female and male jackals with respect to the prey. [23].

Flow Chart

Figure 3 below displays the suggested system's flowchart.

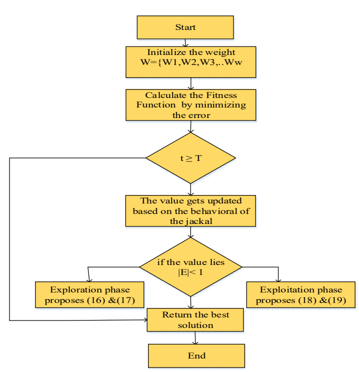

Figure 3. Flowchart of the Golden Jackal Optimization Algorithm

$$p_b \leftarrow max_p(avg - min_{th})/(max_{th} - min_{th}) \qquad (20)$$

The p_b increases slowly as the amount of data climbs from the last packet that was observed,

$$p_b \leftarrow p_b/(1 - count\,.p_b) \qquad (21)$$

The algorithm would be changed to guarantee that the likelihood of recognizing a packet is inversely related to its byte size if this decision were made. [23].

III. Result and Discussion

According to the suggested methodology, MANET will be used to improve the TCP traffic's performance, making it a worthwhile method of networking among the sender and the receiver. Using an Artificial Neural Network (ANN) deep learning technology, this should be made effective. For assistance with the testing, Python 3.8 is utilized along with an Intel Core i5, an Nvidia GeForce GTX 1650, a 16-bit operating system, and 16GB of RAM. Table I displays the GJO parameter used in the simulation.

Table I. Golden Jackal Optimization Parameters

Parameters	Range
Search Agent	10
Maximum Iterations	100
Upper Limit	1
Lower Limit	0

The above table represents the parameter of GJO optimization which has the search agents of 10, the maximum iteration is 100. In addition to that both of the upper and the lower limits are 1 and 0 which is also depicted in the above tabulation. In the section below, a comparison analysis of RBFNN techniques based on deep learning with other existing methodologies is provided. Initially the data transmission will be took part if the sender side data got acknowledged. As soon as the data got received from the sender side then the congestion prediction technique will be initiated this prediction form of detection will be performed by using the Artificial Neural Network(ANN) to predict and congestion is occurring or occurred in the path of transmission.

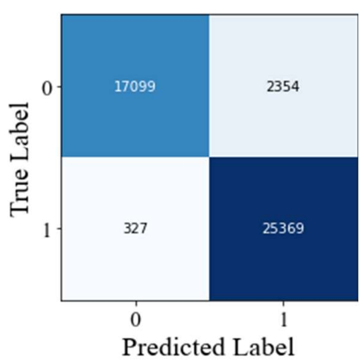

Figure 4. Confusion Matrix.

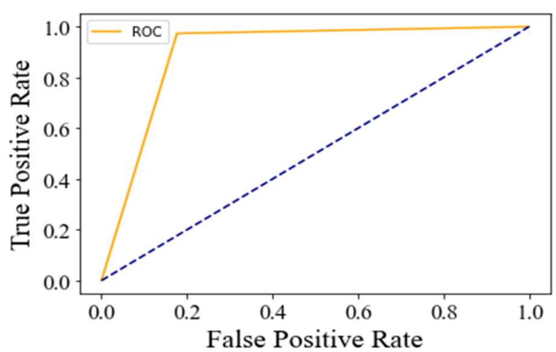

Figure 5. ROC Plot.

Figure 4 displays the confusion matrix plot for the proposed system. A confusion matrix compares the estimated target values from the machine learning model with the actual target values to evaluate the way a classification model is working. The graph above illustrates the expected values for classes 0 and 1 that are 17099 and 25369, respectively.

A. Comparison Analysis of Congestion Prediction

The proposed work of the congestion prediction is compared to the other several other existing techniques. A comparison was done between a number of parameters, including Naïve Bayes (NB), Decision Tree (DT), Probabilistic Neural Network (PNN), Golden Jackal Optimization – Artificial Neural Network (GJO-ANN), and Support Vector Machine (SVM). Various measures are used in this particular area of congestion prediction, including FNR, FDR, accuracy, precision, recall, error, specificity, and F1-Score.

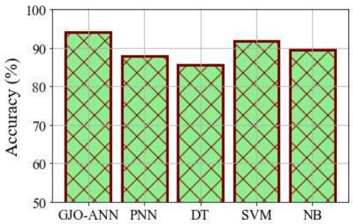

Figure 6. Accuracy comparison.

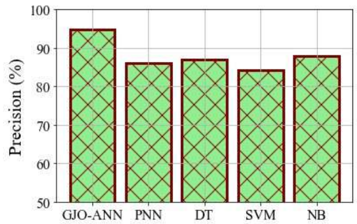

Figure 7. Precision comparison.

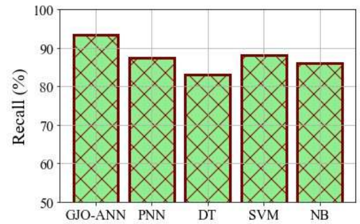

Figure 8. Recall Representation

The accuracy and precision representations are displayed in Figures 6 and 7. Though the accuracy values of the GJO-ANN, PNN, and DT are 94%, 85.7 %, 91.9%, and 89.5%, respectively, the SVM and NB have higher accuracy values. The accuracy comparison is depicted in the image, where the GJO-ANN is explicitly shown to have a precision value of 94.8%, the PNN as 86%, the DT as 87%, the SVM as 84.2%, and the NB as 88%.

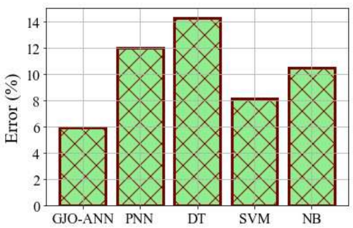

Figure 9. Error comparison.

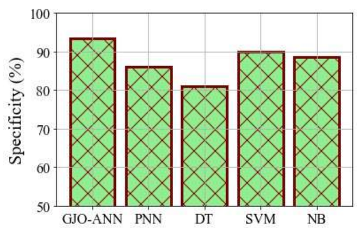

Figure 10. Specificity comparison.

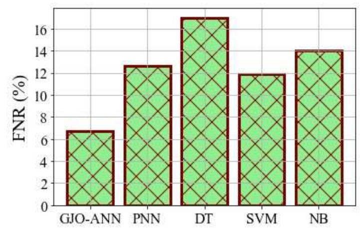

Figure 11. FNR comparison

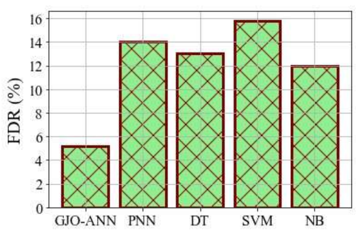

Figure 12. FDR comparison.

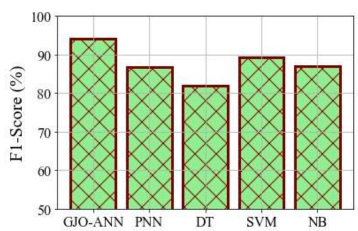

Figure 13. F1-Score comparison

Figures 8 and 9 demonstrates the recall and error comparison representations. The recall representation for GJO-ANN has a score of 93.31%, whereas the PNN, DT, SVM, and NB scores are 87.34%, 83%, 88.1%, and 86%, respectively. The implicit GJO-ANN has a value of 94.81%, the PNN has an 86% value, the DT has an 87% value, the SVM has an 84.2% value, and the NB has an 88% value. This is where the error comparison is shown in the figure.

Figures 10 and 11 illustrate specificity, and the FNR representations show specificity. As a result, the specificity representation's GJO-ANN value is 93.31%, while the PNN, DT, and SVM have values of 86%, 81%, 90%, and 88.6%, respectively. The FNR comparison is displayed in the image, explicitly showing the 6.69% value which is equivalent to the GJO-ANN value of 93.3%, the PNN value of 87.34%, the DT value of 83%, the SVM value of 88.1%, and the NB value of 86%.

The FDR and the F1 representations are shown in Figures 12 and 13. While the FDR value for the proposed GJO-ANN is 94.06%, the PNN has 88%, the DT has 85.7%, the SVM has an 91.9%, and the NB will have an value of 89.5%. The F1-Score value for the GJO-ANN is 94.05%, the PNN value is 86.8%, the DT value is 81.89%, the SVM value is 89.23%, and the NB value is 87%.

B. Comparison Analysis of Congestion Avoidance

The proposed work of the congestion avoidance is compared to the other several other existing techniques. The comparison was made in between some proposed and current protocols like Random Early Detection Protocol (REDP), Dynamic Source Routing Protocol (DSRP). Optimized Link State Routing Protocol (OLSRP), Ad hoc On-Demand Distance Vector (AODV), and Constrained Application Protocol (CAP). In this specific field of congestion avoidance, some of the metrics that are utilized are as follows: throughput, packet delivery ratio (PDR), delay, packet loss (PLR), and energy consumption (EC).

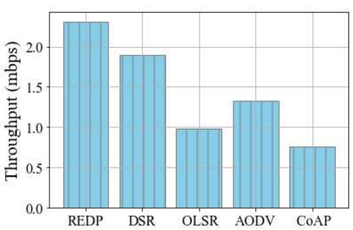

Figure 14 Throughput comparison.

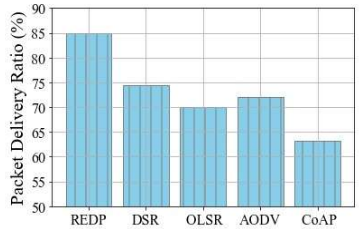

Figure 15. PDR comparison.

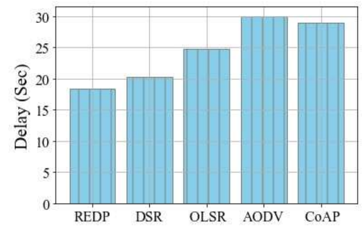

Figure 16. Delay comparison

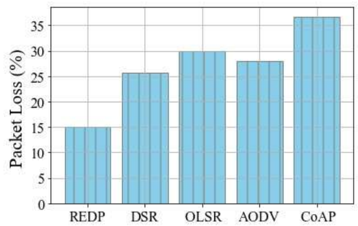

Figure 16. Packet loss comparison.

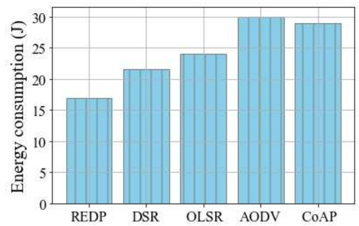

Figure 17. Energy Consumption.

Figures 14 and 15 depict the throughput and the packet delivery representation. The throughout value for the proposed REDP is 2.31 mbps, DSR has 1.89 mbps, OLSR has 0.98 mbps, AODV has 1.32 mbps and, COAP holds the value of 0.755 mbps. Whereas the Packet Delivery Ratio (PDR) holds the value of REDP as 84.89%, DSR as 74.34%, OLSR as 70.1%, AODV as 72.12% and COAP holds the value of 63.3%.

The delay, packet loss, and energy consumption ratio are depicted in Figures 16, 17 and 18 along with their respective representations. The delay values of REDP is18.4 sec, DSR has 20.2, OLSR has 24 sec, AODV has 30 sec, and CoAP has 28.9 sec respectively. For packet loss the REDP is valued at 15.11%, DSR is valued at 25.66 %, OLSR is valued at 29.9%, AODV is valued at 27.88%, and COAP is valued at 36.7 correspondingly. The energy consumption values of REDP is 18.4 J, DSR is 20.2 J, OLSR is 24 J, AODV is 30 J, and COAP is 28.9 J respectively. Compared to other current approaches, the proposed REDP performs better.

IV. Conclusion

The performance of Transmission Control Protocol (TCP) is enhanced by developing Random Early Detection Protocol (REDP) with deep learning based optimized neural network. By implementing these techniques packet loss of a network will decrease which will has a great impact in the performance of TCP.

High bit error rates, frequent route changes, and partitions are just a few of the issues that beset transport links established in wireless ad hoc networks. When utilizing transmission control protocol (TCP) over these connections, to observe extremely low throughput while TCP interprets absent or delayed acknowledgments as congestion.

As to eradicate the traffic patterns between the user sender and the receiver this type of mitigating the congestion is involved in the exhibition of such commotion.

The transmitter sends a message to the receiver and then the receiver waits for an acknowledgment. If the acknowledgement arrives delayed means then the congestion prediction algorithm is executed by utilising Artificial Neural Network (ANN), whether the data transmission path has any traffic or not. If the path has any traffic means then the congestion avoidance phase is carried out based on REDP.

In addition to this there are various evaluations have been proposed for both the congestion prediction and the congestion avoidance. Proposed GNN-ANN was evaluated over other current approaches such as PNN, DT, SVM, NB. The evaluated parameters of predicting the congestion are accuracy, recall, precision, specificity, FNR, F1-Score, error and the FDR correspondingly.

Proposed system's accuracy is 95%, precision is also 95%, and recall will be 93% Likewise for the avoidance of the congestion avoidance performance for the proposed REDP is evaluated over the other current approaches such as DSR, OLSR, AODV, COAP respectively.

The suggested approach has an average residual energy of 14.4J, an 84.89% packet delivery ratio, 15.1% packet loss, an 18.4 second delay, 2.31 Mbps of throughput, and an 18.4J use of energy. The results indicated the ability of the suggested approach to outperform other approaches in terms of results.

REFERENCES

[1] H. Huang, Z. Sun, and X. Wang, “End-to-end TCP congestion control for mobile applications,” IEEE Access, vol. 8, pp. 171628-171642, 2020.

[2] H. D. Le, C. T. Nguyen, V. V. Mai, and A. T. Pham, “On the throughput performance of TCP cubic in millimeter-wave cellular networks,” IEEE Access, vol. 7, pp. 178618-178630, 2019.

[3] T. Ghanem, W. Elkilani, and M. Hadhood, “Improving TCP Performance over Mobile Ad Hoc Networks with an Adaptive Backoff and Response Approach,” MEJ-Mansoura Engineering Journal, vol. 33(1), pp. 19-27, 2020.

[4] X. Xie, J. Zhao, C. Chen, and L. Wang, “AF-TCP: Traffic congestion prediction at arbitrary road segment and flexible future time,” In International Conference on Algorithms and Architectures for Parallel Processing (pp. 167-181). Cham: Springer International Publishing, 2021.

[5] M. H. Hanin, M. Amnai, and Y. Fakhri, “Improved TCP Prediction Congestion in Mobile Ad Hoc Network Based on Cross-Layer and Fuzzy Logic,” Int. J. Interact. Mob. Technol., vol. 15(14), pp. 125, 2021.

[6] S. R. Fahim, Y. Sarker, and S. K. Sarker, “Modeling and development of a five dof vision based remote operated robotic arm with transmission control protocol,” SN Applied Sciences, vol. 2, 1-10, 2020.

[7] M. Alaei, P. Sabbagh, and F. Yazdanpanah, “A QoS-aware congestion control mechanism for wireless multimedia sensor networks,” Wireless Networks, vol. 25(7), pp. 4173-4192, 2019.

[8] R. Agrawal, N. Faujdar, C. A. T. Romero, O. Sharma, G. M. Abdulsahib, O. I. Khalaf, and O. A. Ghoneim, “Classification and comparison of ad hoc networks: A review,” Egyptian Informatics Journal, vol. 24(1), pp. 1-25, 2023.

[9] S. R. Samal, P. G. Kumar, J. C. Santhosh, and M. Santhakumar, “Analysis of traffic congestion impacts of urban road network under Indian condition,” In IOP conference series: materials science and engineering (Vol. 1006, No. 1, p. 012002), 2020. IOP Publishing.

[10] D. Pandey, and V. Kushwaha, “An exploratory study of congestion control techniques in wireless sensor networks,” Computer Communications, vol. 157, pp. 257-283, 2020.

[11] A. Yahyaoui, A. Jamil, J. Rasheed, and M. Yesiltepe, “A decision support system for diabetes prediction using machine learning and deep learning techniques,” In 2019 1st International informatics and software engineering conference (UBMYK) (pp. 1-4), 2019. IEEE.

[12] N. Kumar, and M. Raubal, “Applications of deep learning in congestion detection,” prediction and alleviation: A survey. Transportation Research Part C: Emerging Technologies, vol. 133, pp. 103432, 2021.

[13] N. Thrimoorthy, R. Chandramma, S. Unnikrishnan, and K. Vanitha, “Ornstein Uhlenbeck Cache Obliviousness Neural Congestion Control in Wireless Network for IOT Data Transmission,” 2023.

[14] F. Muchtar, A. H. Abdullah, M. Al-Adhaileh, and K. Z. Zamli, “Energy conservation strategies in Named Data Networking based MANET using congestion control: A review,” Journal of Network and Computer Applications, vol. 152, pp. 102511, 2020.

[15] Y. Wang, L. Wang, and X. Dong, “An intelligent TCP congestion control method based on deep Q network,” Future Internet, vol. 13(10), pp. 261, 2021.

[16] Y. Su, P. Jiang, H. Chen, and X. Deng, “A qos-guaranteed and congestion-controlled sdn routing strategy for smart grid,” Applied Sciences, vol. 12(15), pp. 7629, 2022.

[17] F. Rad, M. Reshadi, and A. Khademzadeh, “A survey and taxonomy of congestion control mechanisms in wireless network on chip,” Journal of Systems Architecture, vol. 108, pp. 101807, 2020.

[18] T. Chatterjee, R. Karmakar, A. Mitra, A. Ghosh, and Bhattacharya, “ICon: How Can We Intelligently Predict TCP Congestion?,” In 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1-7), 2021. IEEE.

[19] D. Lan, X. Tan, J. Lv, Y. Jin, and J. Yang, “A deep reinforcement learning based congestion control mechanism for NDN,” In ICC 2019-2019 IEEE International Conference on Communications (ICC) (pp. 1-7), 2019. IEEE.

[20] C. Lim, “Improving congestion control of TCP for constrained IoT networks. Sensors,” vol. pp. 20(17), 4774.

[21] A. K. Jain, J.Mao, and K. M. Mohiuddin, “Artificial neural networks: A tutorial,” Computer, vol. 29(3), pp. 31-44, 2020.

[22] N. Chopra, and M. M. Ansari, “Golden jackal optimization: A novel nature-inspired optimizer for engineering applications,” Expert Systems with Applications, vol. 198, pp. 116924, 2022.

[23] S. Misra, B. J. Oommen, S. Yanamandra, and M. S. Obaidat, “Random early detection for congestion avoidance in wired networks: a discretized pursuit learning-automata-like solution,” IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 40(1), pp. 66-76, 2009.