EFFICIENT AND RELIABLE HYBRID DEEP LEARNING-ENABLED MODEL FOR CONGESTION CONTROL IN 5G/6G NETWORKS

[1]Aasha Vinnarasi A, PG Scholar,aashavinnarasi99@gmail.com,
[2]Dr.K.G. Revathi, Professor, ECE department, revathi30pranav@gmail.com

Abstract

The most recent modern communication and the next level of wireless transmission are the 5G/6G network. To achieve continuous packet transmission to a destination, a feature of Internet of Things (IoT) systems, 5G/6G in practice, uses the IoT to operate in high-traffic systems with numerous nodes/sensors. As a result, 5G/6G delivers enormous bandwidth, minimal delay, and high-speed data transfer rates. Therefore, using next-generation technologies, particularly the media access control protocol, is made possible and motivated by 5G. However, the typical congestion control actions have a cumulativeeffect on overall efficiency.A Deep Learning-based Congestion Control Algorithm (DL-CCA) is suggested in this article to enhance performance. The proposed system is generated using Non-Orthogonal Multiple Access (NOMA), Orthogonal Frequency Division Multiplexing (OFDM), and Orthogonal Quadrature Amplitude Multiplexing (OQAM). Current processes hamper the performance of 5G/6G and IoT.To forecast the best enhancement of collision control in the remote monitoring of 5G/6G IoT environments, andusinga deep learning system relying on a decision tree (DT) method is suggested in this paper. The model was applied to a training sample to find the ideal parametric configuration in a 5G/6G context. The information has been used to build deep learning and make it possible to identify the best alternatives that could improve the effectiveness of the proposed approach to network congestion. Forecasting and classification are two more tasks for which the DT technique uses. Any consumer could use charts that DT techniques produce to learn the prediction methodology. With a greater than 91.5 % recognition rate, the DT C4.5 had excellent results.

Keywords - 5G/6G, Hybrid deep learning, congestion control, Internet of Things

1. Introduction to congestion control in 5G/6G networks

Today's network is stressed due to increased internet traffic, particularly on mobile networks. Although cable and wireless link capacity have consistently risen, there is still a growing gap between user requests and what the network can provide. In addition, many new applications demand short delay, high capacity, and dependability [1]. A more plausible solution is to reconsider the protocol stack architecture to allow more effective use of the greater physical level link bandwidth, even while building wired and wirelessly connections with a larger capacity help to ameliorate the problem.

The transport layer's most crucial networking role is capacity control, ensuring user information is delivered consistently. However, creating a congested control system is quite tricky. First, the transit system is a vast, intricate network of waiting times. The kernel's multiple linked queues make up the Transmission Control Protocol (TCP) endpoint host. Whenever a TCP flow enters the Internet, it travels through severalcues at networking equipment andthe end-to-end network [2]. Each of these queues is joined by cross-traffic (such as other TCP streams and User Datagram Protocol (UDP) traffic) and serviced with some scheduling control. To build the queueing neural network model that directs the development of a designed congestion protocol, considerable effort is required to develop a solid knowledge of such a complicated network.

Secondly, an agent at end hosts must investigate the link quality and take autonomous, non-coordinated actions if the end-to-end concept is to be followed. The detected network state is typically inaccurate and postponed, and an action's impact relies on competing hosts' behaviour. Third, to achieve scalability when using routers, the method must be very straightforward (for example, stateless), as the gateway must manage many flows. Lastly, as more mobile systems are attached, lossy and capacity-variable wireless connections present significant design issues for network congestion.

A prior study suggested a new technique to improve connection utilisation in computer networks and prevent data loss whenever heavy traffic [3]. For the realistic and occult assessments of path attributes, a deep learning technique for network congestion yielded a promising outcome [4]. An innovative approach (expert foundation) was developed in a different study to calculate the end-to-end Round-Trip Time (RTT)utilising deep learning. By adjusting to variations in RTT, automated simulations of this expertise system could decrease the number of transmissions while enhancing throughput [5].

An ad-hoc system that includes three components: identifyingbottlenecks, grouping messages, and regulating data overcrowding, used grouping as part of computer vision to reduce congestion. Flow control is much improved by computer vision [6, 7]. Urban areas [8], e-Health [9], and ecological monitoring [10] are only a few of the applications and uses for the 5G/6G environment. Various strategies have been used to improve connection speeds in the 5G/6G era [11]. The computational intelligence strategy works well and has many applications. This can be deployed with the traditional routing protocol to meet the requirements of 5G/6G IoT devices (IoT) nodes/sensors systems.

Predicting the best parametric settings and routes to transmit messages is the goal of using the classification algorithm for network congestion. Finding the best node increases throughput while lowering internet traffic and packet drop. In this study, two parameters—file transfer histories and the assessment of simple path properties—are used in a deep learning strategy for assumptions based on deep learning. Relying on the computer training decision tree (DT) technique, this work proposes a new technique for enhancing the congestion control algorithm. The main contributions of the research are listed below:

- A hybrid system using Non-Orthogonal Multiple Access (NOMA) and Orthogonal Quadrature Amplitude Multiplexing (OQAM) is created in this research.
- A congestion control algorithm is created in this research to optimise the system outcome and generate higher throughput.
- A deep learning-basedcongestion control algorithm is used,and the system is developed and implemented. The software outcomes show the effectiveness of the proposed method, which reduces interference and enhances the throughput.

The remainder of the article is organised as follows: section 2 indicates the background to the congestion control algorithms. The proposed Deep Learning-based Congestion Control Algorithm (DL-CCA) is designed, and the outcomes are shown in section 3. Section 4 demonstrates the software outcomes of the congestion control and the impacts on the 5G/6G network. Section 5 indicates the conclusion and future scope of the proposed model.

2. Background to the congestion control algorithms

Over the past three decades, research and attention have focused on collision avoidance's fundamental networking issue. This section summarises the most significant recent developments in 3 groups: end-to-end, router-based, and intelligent schemes.

2.1End-to-end models

Loss-based or delay-based techniques can further categorise this type of activity. Early loss-based methods include TCP Reno [12], TCP Tahoe [13], and TCP NewReno [14]. The first delay-based technology is TCP Vegas. Currently, most computer operating systems use TCP Cubic [15] or Compound TCP [16], primarily built to handle huge capacity RTT devices.

They initially intended for the wired network of these technologies. However, recent tests in 3G and 4G systems show that channel capacity varies quickly [17]. Additionally, Liu et al. [18] demonstrate how competitive traffic and bandwidth scheduling impact RTT and capacity utilisation. Following that, the authors provide brand-new end-to-end techniques that use packet travel time to deduce the fuzzy clustering algorithms (i.e., Sprout) and use latency measurement to deal with these tasks.

The concept of data centre networks represents another new development in congestion control. Compared to ordinary networks, data centre networks typically have links with more significant and more stable (e.g., gigabits or tens of gigabits), a significantly lower latency (e.g., in s), and anAcknowledgement (ACK) that is not delivered for every message. The Data Center TCP (DCTCP) [19] takes advantage of switches' Echo Congestion Notification (ECN) response. Performance-based Crowd Control (PCC) advocates continuously tracking the relationship between activities taken and actual performance to make decisions that would improve performance [20]. Yeet al. [21] analyse and solves the Quantised Congested Notification (QCN) stability problem in the data centre Ethernet connections using tracking control theories.

2.2Router-based models

Switching or routers are used in the packet scheduling procedure in this class of activity. As a replacement for loss as a congestion indication, ECN [22] is proposed. Routers label or drop messages on impending difficulty in Active Queue Management (AQM) systems as Random Earlier Detection (RED) [23], CoDel [24], andMaximum to Average Queue(MAQ) [25], [26]. These methods call for the change of gateways and other intermediary devices that is not feasible or scalable to several TCP flows.

2.3Smart models

There is much interest in using machine learning to address networking issues, given the recent achievement of machine learning/deep teaching in machine vision, video monitoring, and computational linguistics. The findings in [27] offer intriguing new information about machine-generated bottleneck algorithms. However, these learning techniques can only be used in a few circumstances and require offline training on the network's existing knowledge. Xu et al. [28] provide a Deep Learning (DL) based approach to address the transportation planning issue at intermediary nodes in the cluster layer. Q-learning is included in the congestion control system design, and Q-based TCP (QTCP) is presented by Li et al. [29]. It is based on constrained feature sizes (discretised states) and approximates the Q-function using Kanerva Coding rather than a Deep Neural Network (DNN). The authors demonstrate gains done by QTCP over the traditional TCP newreno method.

2.4 Motivations

The investigation of the present TCP congestion management algorithms reveals several unresolved problems with their design and implementation in diverse networks. (1) Model dependence: Current TCP congested control strategies are built on understanding network properties and traffic modelling. The traffic control methods for these models invariably perform poorly when implemented in reality since such models are frequently erroneous and simplified. (2) Flexibility: The traditional techniques of congestion control management rely on straightforward defined rules that cannot be adjusted to a wide range of network situations and Quality of System (QoS) goals. (3) Autonomy: Because the current TCP traffic control algorithms are reactive and passive, they cannot actively optimise resource use.

To address the problems above, this research utilises computer vision to derive new congested control parameters for TCP. A learning-based strategy can develop alternative rules for various scenarios instead of a fixed norm for window modification, respond quickly to thechanging conditions, and change the policies adaptively to maximise the connection speed.

3. Deep Learning-based Congestion Control Algorithm

The packet scheduling algorithm's primary mechanism operates via the TCP protocol. If the TCP is active throughout this layer, numerous activities, including message transmission and printing, fault detection, and information retrieval, can be completed. These functions are modified according to the network's capacity for use—the terminals-area-used development level for a precise examination of the number of terminals needed to complete specific tasks.

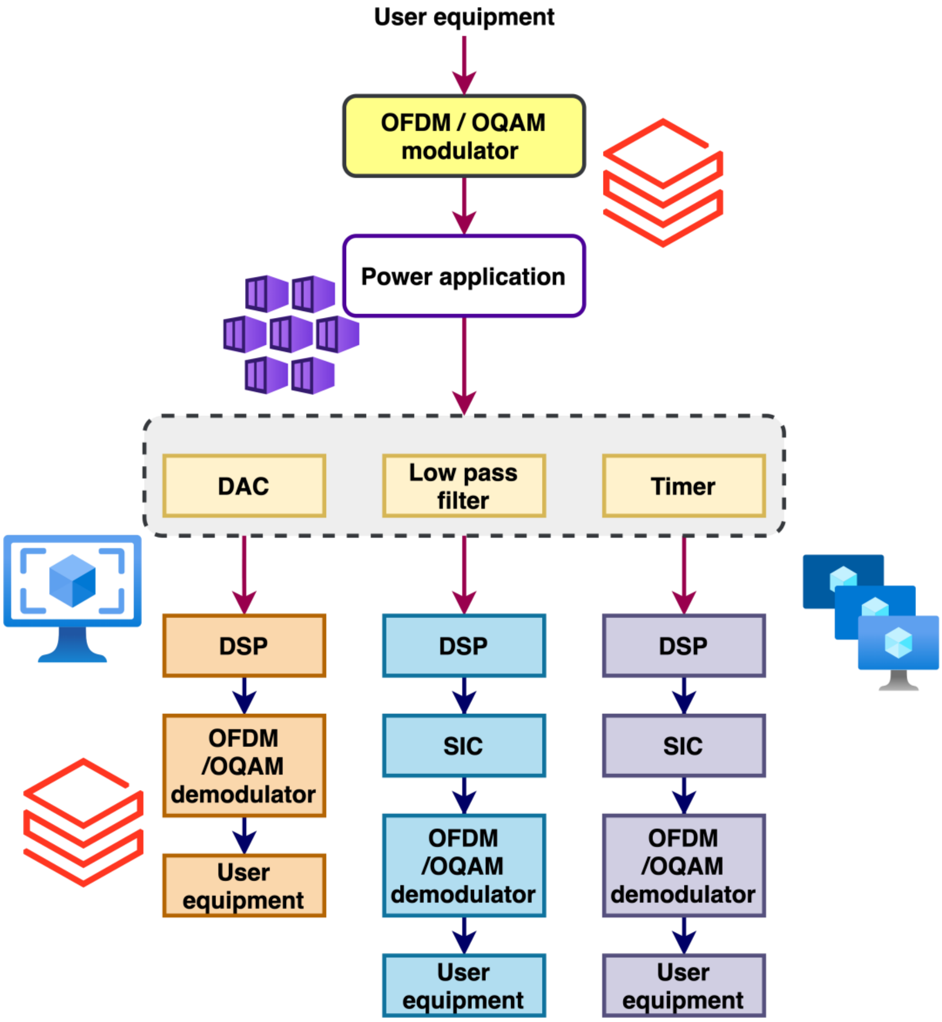

Fig. 1. SystemThe system architecture of the DL-CCA

Fig.1 illustrates the system architecture of the DL-CCA. The suggested system access the data from Frequency Division Multiple Access (FDMA) and Orthogonal Quadrature Amplitude Modulation (OQAM). The Digital to Analog Converter (DAC) is used to convert the received digital signal and convert it to ananalog signal. Digital Signal Processing (DSP) is used to process the analog spectrum and separate the spectrum based on the user's needs. System Induced Chip (SIC) is directly affected by FDMA/OQAM to receive the signal at the User Equipment (UE).

A transmitter and a recipient must be connected for the data transit technique to work. With a specific destination, the transmitter establishes a connection. To study cwnd_ and ssthresh_, it sends the original message and awaits acknowledgement (ACKed). Cwnd_ and ssthresh_ are recognised when the receiver returns ACKed to the broadcaster. As a result, congestion is controlled. The missing phase is recovered if the receiver doesn't conduct ACKed. The slow-start stage is created if cwnd_ is below or sufficient to ssthresh_. Congestion detection stages are carried out when cwnd_ exceeds ssthresh_. Until all texts are supplied, this process is repeated.

3.1 Mathematical system

The simulation of traditional or unique TCP in a 5G/6G IoT ecosystem revealed that cwnd_ grew steadily and linearly. It was believed that cwnd_ would expand each ACKed in compliance with a written interval after cwnd_ ≤ ssthresh.

3.1.1. Enhanced slow-start

The improved Slow-Start (SS) phase is started after a lengthy delay or when data transfer begins in an anonymised environment. The TCP searches the entire network. Ssthresh_ is considered to indicate the bandwidth utilisation to prevent techniques that use more resources than are available. A retransmit clock can identify the slow-start phase, which follows the recovery from transmission errors. After each acknowledgement, the cwnd_ should be raised if less than or equivalent to ssthresh_. As a result, the cwnd_ value should increase dramatically. The sender must adhere to the algorithm's permissible demands during this phase. Equation (1) demonstrates how slow-exponential start improvement works.

$$cwnd\_ = cwnd\_ + \propto \qquad (1)$$

Wherein$\propto$ can be considered the location's Path Transmission Unit (PTU) or previously acknowledged encouraging factor chunks (data ACKed). The Target PTU is referred to as TPTU, while the total size of the previously unacknowledged recognised information is ACKed.

The typical slow-start function is illustrated in Equation (2).

$$\Delta cwnd\_ = minimum\ (D_{ACK}, \propto) \qquad (2)$$

The data acknowledged is expressed as D_{ACK}, and the minimum window size is denoted as to determine the amount of channel capacity. The suggested slow-start function is expressed in Equation (3).

$$\Delta cwnd\_ = minimum\ (D_{ACK}, \propto) + \frac{cwnd\_}{R} \qquad (3)$$

$D_{ACK}$is the last acknowledged data and the minimum window size is expressed as $\propto$. The current window size is denoted *cwnd_*, and the number of acknowledge received in the last Round Trip Time (RTT) is indicated as R. Based on many performance criteria, the variable R is a number ranging from 1 to 100. These elements include network performance, congestion windows, number of packet losses, number of sent messages per time unit, and number of data packets retrieved per period. The selection process aims to identify the component that significantly impacts the routing protocol. The ideal R is determined by substituting the suggested equation for the value of R. The results of the performance indicators decide on the perfectR-value.

Maximum window size, highest queue size usage, most significant amount of accepted and sent messages, bandwidth, and most minor packet drop is the criteria for establishing the optimum result. When R is more than 9, packet loss becomes worse. This incidence has an impact on the queue size. The number of R in improved slow-start varies from 1 to 100 depending on these factors, the high-efficiency findings, and the experimental data.

3.2 Congestion control

The prediction data for the increments Inc_p of the queues at time p is gathered during the adaptive learning phase. The degree of data traffic is calculated using the average queuing theory for each router by the first phase's forecasted data. A more extendedNegative Acknowledgement (NACK) controller output packet follows, feeding the receiver the network congestion. The receiver modifies its Attention transmitting rate after getting the NACK message. Every router's interface gathers them at every interval in one of three typical combinations.

3.2.1 Congestion detection

System can determine the anticipated number of returned data packages by figuring out how many interest packages are in the waiting line to be sent. The network's congestion level is determined by. A period p is partitioned into n slots of similar duration without losing generality. Every slot p corresponds explicitly to the time range$[p-1, p]$. This presumption assumes that the algorithm that uses the linear growth approach. If Q_p is the size of the immediate Interest queue that was detected during a period, then B_p is the weighted value of Q_p that is greater than $B_{p-1}$of Q_{p-1}. The weighted function B_p is computed in Equations (4) and (5).

$$\sum_{p=0}^{n-1} B_p = 1 \tag{4}$$

$$B_p = \beta B_{p-1} + \propto \tag{5}$$

where β and $\propto$ are constant, $\beta > 1, \propto = 0$, the weighted function is expressed as B_p, and the weighed mean tenure of the queue over period p isB_{p-1}, given that the instant queues capacity to transmit attention messages at every time frame $T_p$is indicated in Equation (6).

$$Q_{mean} = \frac{\sum_{p=0}^{n-1}(Q_{p-1}+Inc_p-T_p)}{B_p} \quad (6)$$

The previous queue length is denoted as Q_{p-1}, the total queue size of the prior time to present time is denoted as Inc_p, and the time deviation is expressed as T_p. The weighted function is shown as B_p. $Inc_p$is the increase in the expected number of queues at time t, and Q_{p-1} is the line size at time p-1. The Quality of Service (QoS) is enhanced using the deep learning model.

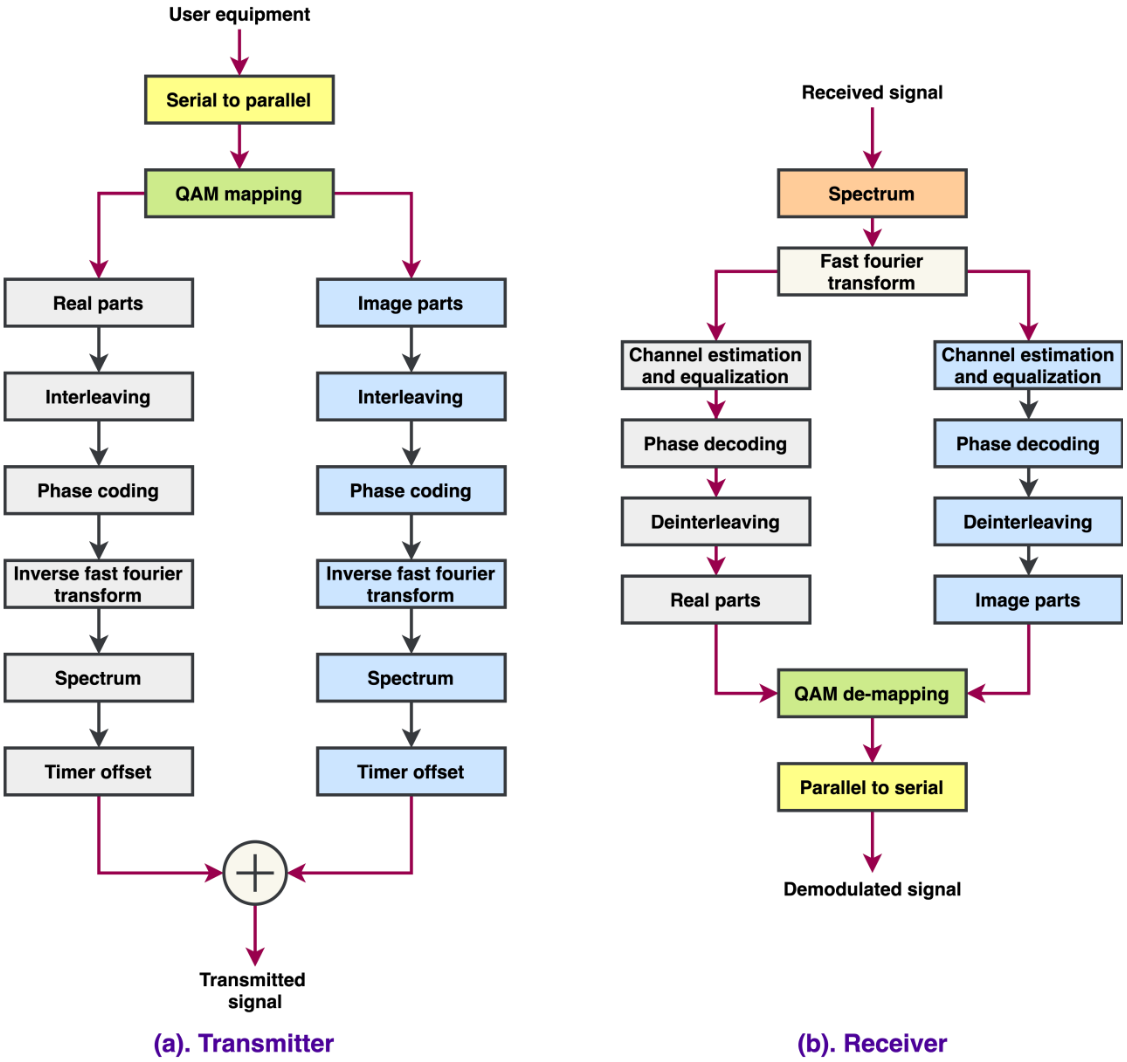

Fig. 2(a). Transmitter and 2(b). Receiver structure of the DL-CCA

The transmitter and receiver structure of the DL-CCA is shown in Fig 2(a) and 2(b). The user's digital input and serial input is converted into the parallel stream. The Quadrature Amplitude Modulation (QAM) is used to split the data into real and imaginary terms. Then unwanted data is interleaved, and then phase coding is done to incorporate multiple users for the same frequency. The Inverse Fast Fourier Transform (IFFT) is used to separate the spectrums. Then the final result is added together to get the transmitted signal. The receiver starts working from the received signal with noise. The Fast Fourier Transform (FFT) is used to separate the channels and then the channel is estimated and equalised using the weight updation. The phase decoding and de-interleaving are used to process the signal, and then real and imaginaryparts are combined to fetch the QAM signal. The QAM de-mapping is done, and the signal is converted to serial form using a parallel to the serial system.

The three phases of data traffic used are Q_I, Q_b, and Q_M. $Q_{mean}$is used to measure network problems. These three attributes go up one by one, with $Q_I$standing for an empty queue, Q_b for a busy waiting line, and $Q_M$for the queue's full capacity. Additionally, NACK packets with the load demands must be returned to the recipient. The following are the categorisation guidelines:

The threshold is set as $0 < Q_I < Q_b < Q_M$. The idle link condition is expressed as $Q_{mean} < Q_I$.The light busy link condition is denoted$Q_I < Q_{mean} < Q_b$. The heavy busy link condition is expressed as $Q_b < Q_{mean} < Q_M$, and the congested link is denoted as$Q_{mean} > Q_M$. Where Q_I, Q_b, and $Q_M$are the threshold variables which categorise the congested level. If $Q_{mean} < Q_I$, the connection is stable. If $Q_{mean} < Q_I < Q_b$, the connection is light busy. If $Q_b < Q_{mean} < Q_M$, the connection is heavy busy. If $Q_{mean} > Q_M$, the connection is crowded. The corresponding response data is captured at the receiver through the above limits.

3.2.2 Explicit congestion notification

After congestion identification, the receiver receives the congestion data. By expanding the NACK package, it uses a custom control packet that contains an extra congestion state element to represent one of the traffic volumes:

- The traffic value for the idle link is shown as "00".
- The traffic value for the busy lightlink is indicated as "01".
- The traffic value for the heavy traffic link is marked as "10".
- The delay value is shown as "11" in the congested link.

The receipt of the attention packet initiates the NACK packet. Additionally, the NACK package defines a field to identify it from the interested packets and wraps the data about the load demands into an interested packet. The recipient doesn't need to modify if the congested level it receives is "00." If it is someone else, the receiver adjusts the Interest packet's data transmission by the load demands included in the NACK message. Additionally, the NACK packet replicates the identity into the attention package before sending it farther down the packet transmission chain to the receiver. The QoS is directly related to ECN; the congestion control method enhances the QoS by reducing the packet drop.

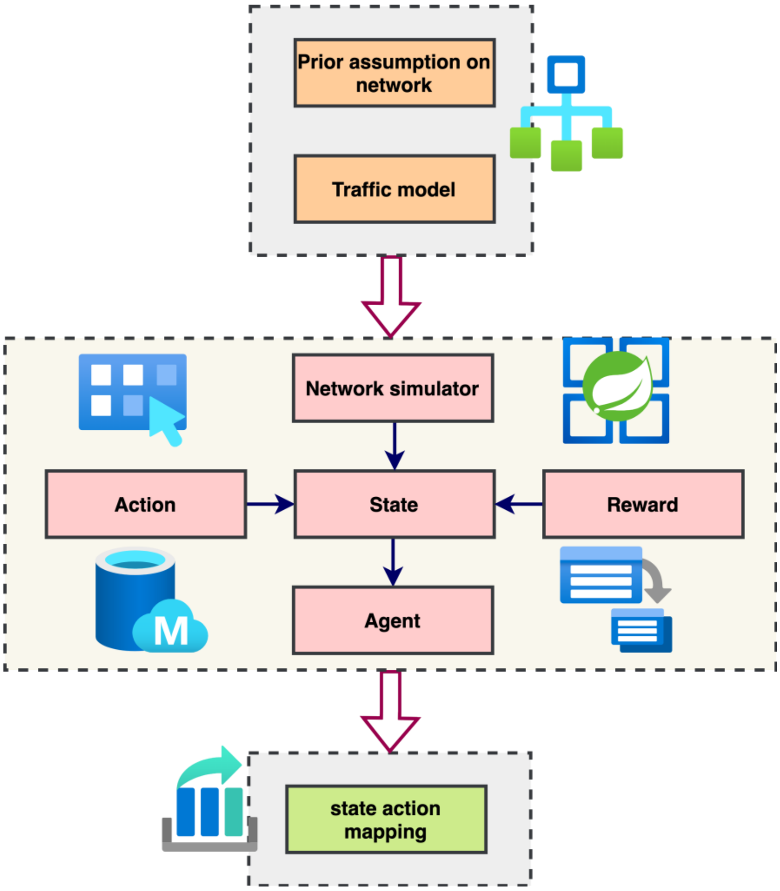

Fig. 3. The network monitoring algorithm of DL-CCA

The network monitoring algorithm of the DL-CCA system is shown in Fig. 3. The network model is assumed, and the traffic model is assigned initially. The network consists of an action module, network simulator module, state module, reward module, and agent module—the state action mapping controls and monitors the network intermediate node's queue size. Between the load demands identified in this network and the initial traffic degree in the NACK message when travelling over each intermediary node, the blocked data in NACK message is modified with the stronger one. As a result, when the NACK product has reached the recipient, it contains the link's busiest condition.

The window-based speed regulation is also used, which is identical to TCP. The maximum total current interested packets permitted at the receiver is indicated by the congestion control variable (W). The transmit power of the incentive message at the recipient is adjusted using the Exponentially Increment Addition Increment Multiplication Decrease (EIAIMD) algorithm by the observed congestion situation.

In particular, if a "00" is obtained, the Exponential Increment (EI) method is used to fully utilise the available capacity. If the number "01" is obtained, the connection is only lightly congested but has additional capacity. Thus, the Additive Increase (AI) algorithm reduces congestion. If the number "10" is obtained, the system is already extremely busy, and increasing the transmitting window would probably lead to congestion issues. Therefore, the transmit window size is kept constant. If the signal "11" is received, it swiftly shrinks the window size using the Multiplicative Decrease (MD) method. The deep learning model enhances the Signal to Noise Ratio (SNR). The EIAIMD method in further depth is as follows:

The exponential growth method is shown in Equation (7).

$$B_{p+RTT} = B_p(1+\beta) \tag{7}$$

The current window size is denotedB_p, and the incremental factor is expressed as β. The additional increment method is expressed in Equation (8).

$$B_{p+RTT} = B_p + \gamma \tag{8}$$

The current window level is expressed as B_p, and the additive incremental factor is expressed as γ. The multiplicative decrease method is expressed in Equation (9).

$$B_{p+RTT} = \frac{B_p}{\alpha} \tag{9}$$

whereB_p denotes the window's width at time p. The βstands for round travel time. In the EI method, $\beta$is an exponentially rising factor. In the AI method, $\alpha$is an additive risk factor. In the MD method, $\beta > 0, \gamma > 0, and\ 0 < \alpha < 1$is a multiplication decrease factor. In a multi-source scenario, it is demonstrated that the congestion method forecasts congestion problems more precisely than a sole RTT calculation. To prevent congestion issues and preserve the network's superior efficiency, it actively notifies the recipient before it happens.

3.3 System architecture

The fundamental Non-Orthogonal Multiple Access (NOMA) utilising SIC for D recipients in the downlink when assuming a multi-path fading network with one BaseStation (BS) and D (the endpoints). After the OFDM/NOMA modulation, the information signal of $D_i(i = 0,1,2,\cdots,n)$ could be represented in Equation (10).

$$M_i(t) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \frac{\propto_{x,y}}{f\left(t - \frac{x(N-1)}{2}\right)} \exp\left(\frac{2jyt}{N-1}\right) \tag{10}$$

$\propto_{x,y}$is the real-valued information of the xth blocks on the yth subcarrier, wherein N represents the size of subbands in an OFDM/NOMA blocks. The input and the output dimension are denoted as $x\ and\ y$. The symbol $f(t)$in the time-varying domain indicates the prototype filter's input signal. While applying less energy to D and having a strong channel quality indication, all D in the Network coding can use the whole spatial-frequency choices concurrently. In contrast to OFDM, NOMA allocates different amounts of electricity to every D at the same TF choice. Each D uses the entire power judgement with a distinct Time-Frequency (TF) region to modify transmitted signals. The data transmission process is expressed in Equation (11).

$$M(t) = \sum_{i=0}^{N-1} \frac{\sqrt{T_p W_p}}{M_i(t)} \qquad (11)$$

$T_p$is the transmitting power, and W_p represents each D's interference. The transmitted power of the data is expressed as $M_i(t)$. The signal $\beta_p$that was detected at D is indicated in Equation (12).

$$\beta_p = g_p \propto + b_p \qquad (12)$$

Where b_p is the baseweight and g_p is the fibre impulse response sensitivity. The scaling factor is indicated as $\propto$. Every D in NOMA can utilise the whole bandwidth (BW) continuously, resulting in a considerable increase in the sum speed. The SIC method is put into practiceon the recipient side. The CQI, or the channels image file to noise ratios, also determines the best order for data decoding. For example, the decoder sequence for $OS_x > OS_y$is $O_x > O_y$.It is considering a 2-D situation and $OS_x > OS_y$, additional power should be given to D_2 in the NOMA downstream to ensure the signal quality against interfering. D_2can read j_2 from k_2 because it occurs first in the decoder order, hence it does not require SIC speed. In contrast, D_1 can decrypt j_1 without SIC's influence fromj_2.

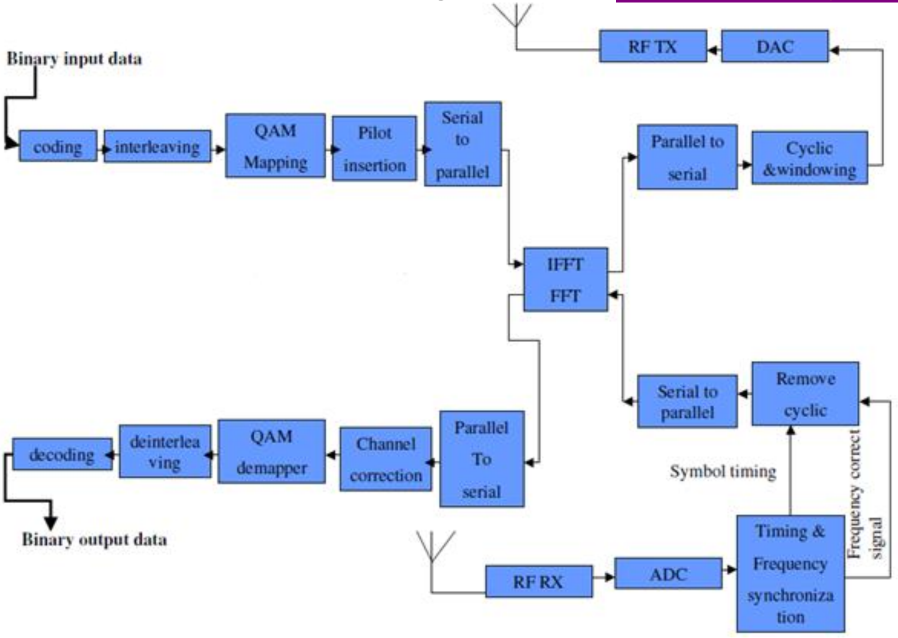

Fig. 4. The end-to-end architecture of the DL-CCA

The end-to-end architecture of the DL-CCA is shown in Fig. 4. The input binary signal is fetched, processed and modulated. The received signal is decoded in the receiver end and binary data is decoded.Considering ideal SIC capacity, the throughput Th_x of D_x for NOMA is denoted in Equation (13).

$$Th_x = W \log_2(1 + \beta * OS_x) \qquad (13)$$

The binary data is denoted as W, the output sequence is denoted as OS_x, and the window scaling function is expressed as β. The throughput Th_o of D_o for OFDM is expressed as Equation (14).

$$Th_o = \frac{W}{(\gamma_0 + \beta_0)} \log_2 \frac{1 + \beta * OS_x}{\beta_0} \qquad (14)$$

The binary data is expressed as W, the window scaling factor is denoted as β. The initial window size is expressed as β_0, the output sequence is described as OS_x, and the biasing factor is defined as γ_0.

3.4 Development model

The enhanced TCP traffic control technique of devices in the 5G/6G IoT ecosystem is discussed in this study. Better slow-start and traffic shaping methods must be used to accomplish the third goal. The limitations of traditional collision avoidance in the 5G/6G IoT ecosystem were resolved using upgraded methodologies. The final form of DL-CCA was developed using various development techniques based on an enhanced load-balancing device that takes advantage of Decision Tree (DT) capabilities. To create DL-CCA, two stages of testing with multiple scopes were used. DL-CCA congestion management offered the most delicate I-selection-based window management in the initial stage.DT was in charge of I-selection in the second phase. Queue length, performance, packet drop, and cwnd_ were used as the DT methods' assessment criteria.

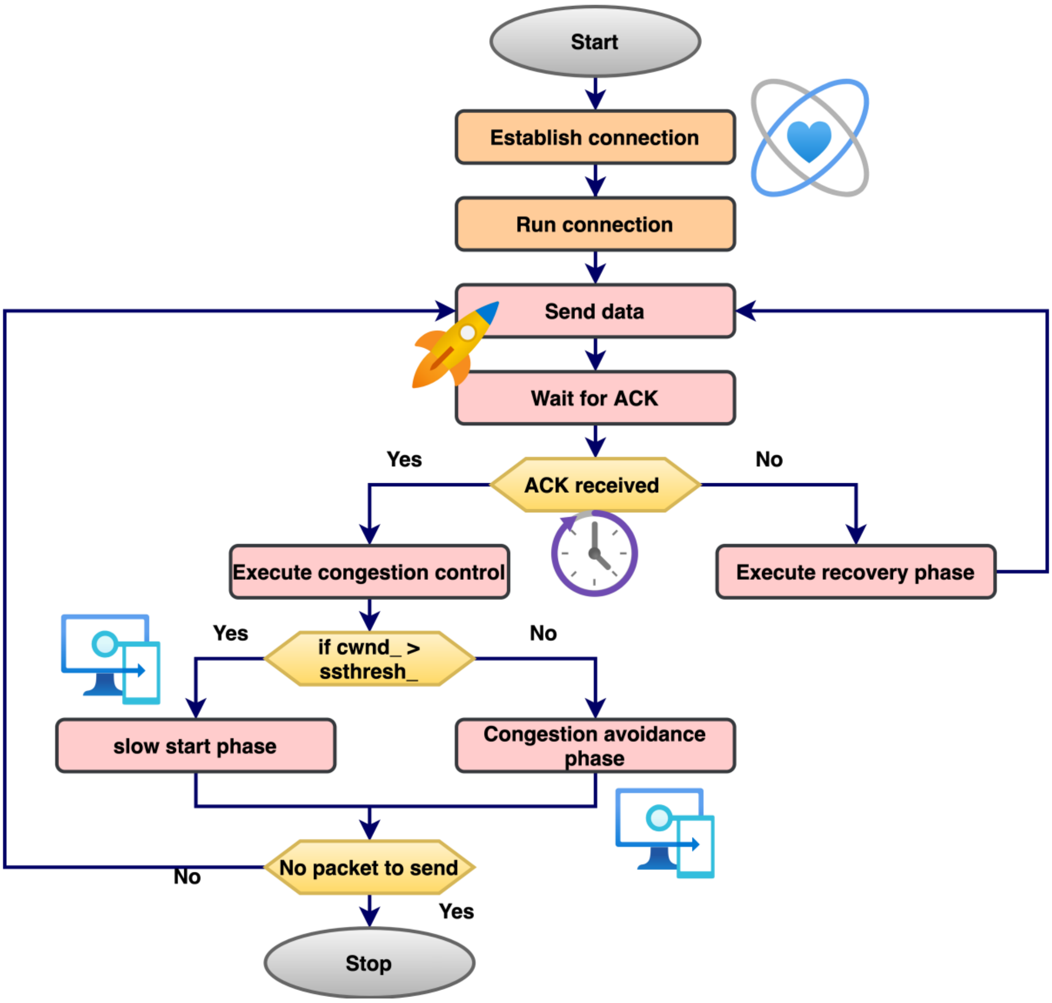

Fig. 5. Congestion control workflow of the DL-CCA

The congestion control workflow of the DL-CCA is depicted in Fig. 5. The connection is established and then waited for the reception of ACK. If ACK is received, the further packet is transmitted; if not, the congestion control process is established. After sending all the data, the connection is released. All potential trials, including associated situations, have been addressed and provided. Experimental investigations were created depending on these phrases. The following section goes into the specifics of these tests. The enhanced capacity control techniques use modified slow-start and traffic shaping methods.

The C4.5 methodis used with deep learning to arrive at the ideal Regression (R) value. The C4.5 method is implemented in J48, an open-source programme. As a result, using the training tables, which would be a composite of many performance variables, J48 finds the optimal option and forecasts the new ideal alternative. J48 uses the following stages to produce a DT.

3.4.1 Applying the deep learning

J48 is produced during this stage of the modelling process. Numerous fields use deep learning techniques, such as grouping, for applications like assessing students' academic achievement and enhancing the effectiveness of network elements. The DT approach has more roles than other mining techniques. According to earlier studies that used the approach in various fields, such as medicine, e-government, Wireless Sensor Network (WSN) connectivity, mobile multifactor authentication, and penetration testing, the DT method, as a component of computer vision, exhibits good results. The DT algorithm creates an orientated tree network based on the learning data for classifiers. The. Dynamic QoS is helped to enhance the throughput and reduce the end-to-end latency in 5G/6G.

The root node of a tree graph branches into a subtree and terminates with a binary tree. The nodes that portray the dataset's characteristics are encountered on the route from the real cause to the binary tree. This route might be considered for the forecast of upcoming cases. The resulting tree is constructed and translated using IF condition expressions. Information retrieval can use reliable and practical DT deep learning algorithms to uncover underlying patterns in small and large databases. A tree is built in three phases: construction, data gain calculation and trimming. The following fundamental actions are a part of the project stage:

- Check to see if all concerns fall under the same category. As a result, a leaf with a classifier is a tree.
- Determine the gain ratio and the data for one attribute's component.
- Find the best splitting attribute based on the current screening method.

Entropy's computed information gain is employed. The measurement of a randomised variable's unpredictability is called entropy. As follows, the entropy of Q is calculated in Equation (15).

$$E(x) = -\sum_{k=0}^{N-1} \left\| {}^{x_k}/_{x} \right\| \times \log_{10}\left(\left\| {}^{x_k}/_{x} \right\| \right) \qquad (15)$$

The input is denoted as x_k and the sample size is denoted as x. The conditional entropy isrepresented in Equation (16).

$$E(k|x) = -\sum_{k=0}^{N-1} \left\| {}^{x_k}/_{x} \right\| \times \log_{10}\left(\left\| {}^{x_k}/_{x} \right\| \right) \qquad (16)$$

The conditional input is denoted as x, and the input of the DT is expressed as $\square_{\square}$. The data gain is computedusing Equation (17).

$$\square(\square,\square) = \square(\square - \square(\square|\square) \qquad (17)$$

The entropy and the conditional entropy are expressed as □(□)□□□ □(□|□). The resultant tree is compact and effective if the break is predicated on power amplifier. A pruning strategy is used in the concluding stage to deal with overfitting and outliers. Datasets with ill-defined examples can be classified through pruning. There are two categories of pruning. When a tree is created, online trimming is done, and post-pruning is done after the tree is built. The separate-and-conquer rule structure provides the foundation for pruning implementations.

The DL-CCA system is designed in this section with NOMA/OQAM. The deep learning model further enhances the system outcomes,reducing congestion control and increasing the D. The software outcomes are shown in the next section.

4. Software evaluations and the system impacts

This section discusses the tests using Matlabmillimetre-wave (mmWave) analysis to evaluate how well the suggested DT predicted the ideal parametric values for the transmission schemes for sensing over the 5G/6G IoT. Additionally, it reached the results of predictions of the best sensors' collision control techniques over 5G/6G IoT and state-of-the-art DTs. It used a paired-sample t-test to demonstrate the notable discrepancy between the outcomes of standard mechanisms and those of enhanced traffic control strategies.

Table 1. Simulation parameters

Parameters	Value
Bandwidth – uplink downlink	500 Mbps 1 Gbps
Latency	1 msec
Maximum Transmission Unit	1500 B
Data size	1468 B
Carrier frequency	28 GHz

The simulation parameters are shown in Table 1. The uplink and downlink bandwidth are dented as 500 Mbps and 1Gbps. The latency is 1msec, the maximum transmission unit is denoted as 1500B, and the data size is expressed as 1468B. The carrier frequency for the 5G/6G network is considered 28 GHz. The outcomes are evaluated using the Matlab simulation tool.

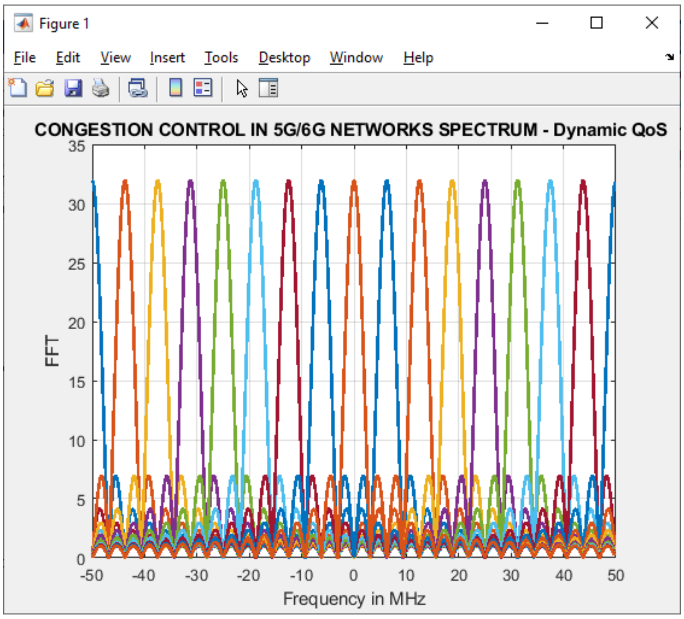
Figure 1
File Edit View Insert Tools Desktop Window Help
CONGESTION CONTROL IN 5G/6G NETWORKS SPECTRUM - Dynamic QoS
FFT
0
5
10
15
20
25
30
35
-50
-40
-30
-20
-10
0
10
20
30
40
50
Frequency in MHz

Fig.6 Congestion control analysis of the DL-CCA system

The congestion control analysis of the DL-CCA system is represented in Fig. 6. The DL-CCA system follows dynamic Quality of Service, which uses a deep learning model. The DL-CCA system with NOMA and OQAM produces higher throughput than the standard Multiple Input and Multiple Output (MIMO). NOMA subcarrier production for 5G/6G and 6G is shown in different spectrums. In secondary systems using non-orthogonal channel access, the efficiency of 5G/6G is evaluated. The DL-CCA system with multiple users at the same frequency enhances spectrum utilisation and reduces interference.

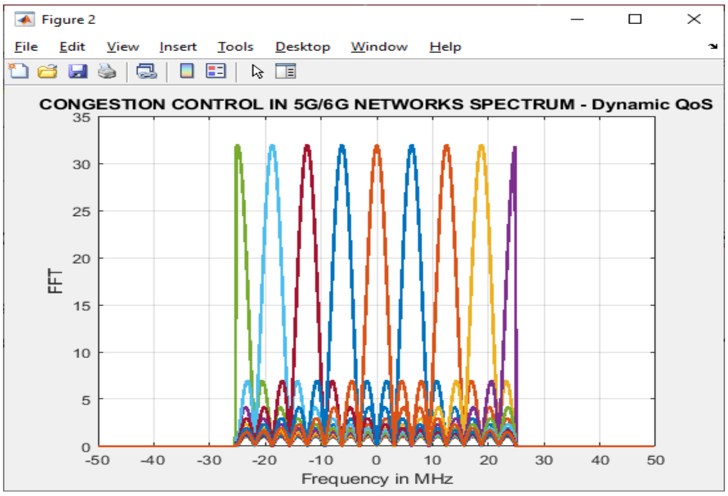
Figure 2
File Edit View Insert Tools Desktop Window Help
CONGESTION CONTROL IN 5G/6G NETWORKS SPECTRUM - Dynamic QoS
FFT
0
5
10
15
20
25
30
35
-50
-40
-30
-20
-10
0
10
20
30
40
50
Frequency in MHz

Fig.7. Bandwidth analysis of the DL-CCA system

The bandwidth analysis of the DL-CCA system is shown in Fig. 7. The deep learning-basedeffective spectrum utilisationis directly related to the number of users. The NOMA/OQAM model enhances the spectrum utilisation, which reduces the bandwidth and hence improvesthe system performance with lower interference with the successive chancels.In secondary systems using non-orthogonal multiplexed access (NOMA), the efficiency of 5G/6G is evaluated. The resulting NOMA subcarrier creation uses much bandwidth.

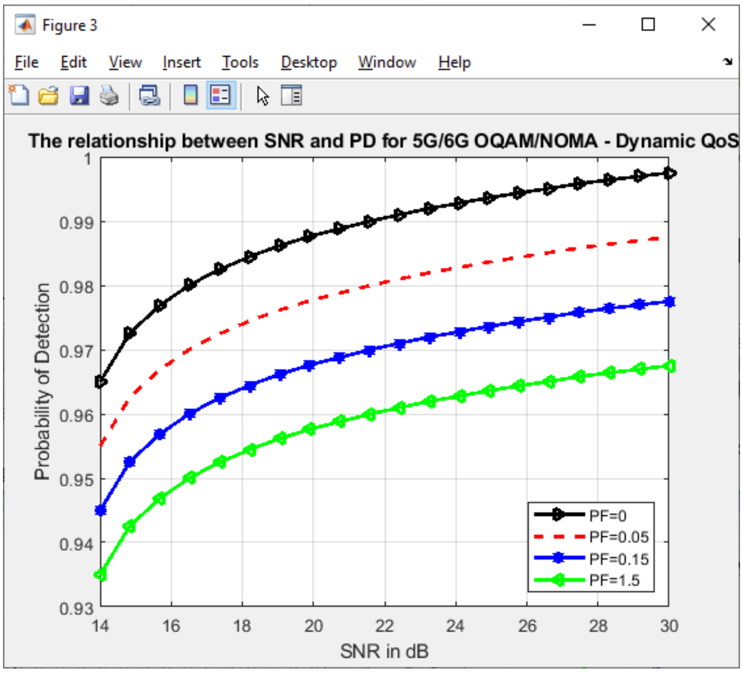

Fig. 8. SNR vsProbability of detection analysis

The SNR and probability of detection evaluation are represented in Fig. 8. The SNR for the simulation analysis is varied from 14 to 30dB with an incremental level of 2dB. As the SNR increases, the detection probability also increases. Compared to other control schemes, the Bit Error Rate (BER) of OFDM/NOMA is significantly lowered in the suggested DL-CCA system.The simulation results are evaluated by varying the Probability Factor (PF) from 0, 0.05, 0.15 and 1.5. The higher probability of detection has reduced the interference and hence increases the system throughput.

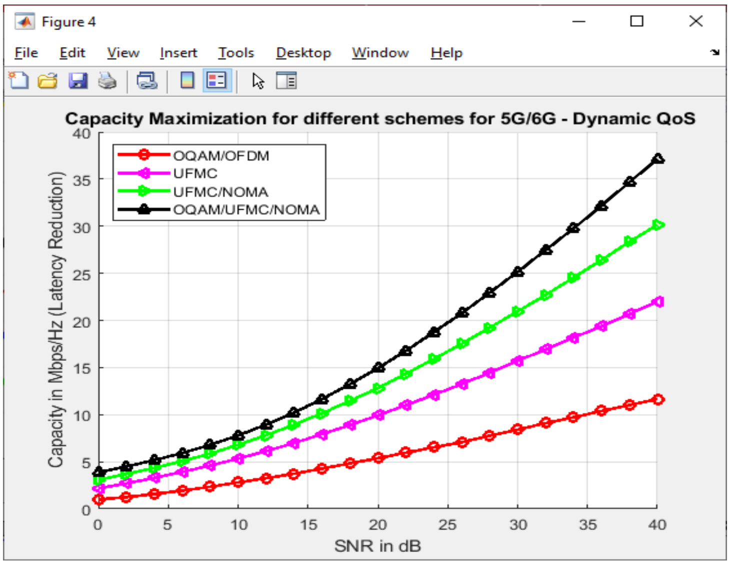

Fig. 9. Capacity maximisationanalysis of the DL-CCA system

The maximum capacity analysis for theDL-CCA system is shown in Fig. 9. For 5G/6G and 6G radio transmission, capacity maximisation is used for various modulation techniques. The simulated system is established to evaluate the categorisation approach's performance and power usage, interference from the backup plan with the primary network, and user capability. Two links with a 10 m gap are established between the transceivers. An antenna for synchronisation is anticipated on every node in the wireless connection. The hybrid system of NOMA, OQAM, Orthogonal Frequency Division Multiplexing (OFDM), and Universal Filtered Multi-Carrier (UFMC) are considered for the analysis. The proposed DL-CCA system exhibits a higher capacity than the deep learning models.

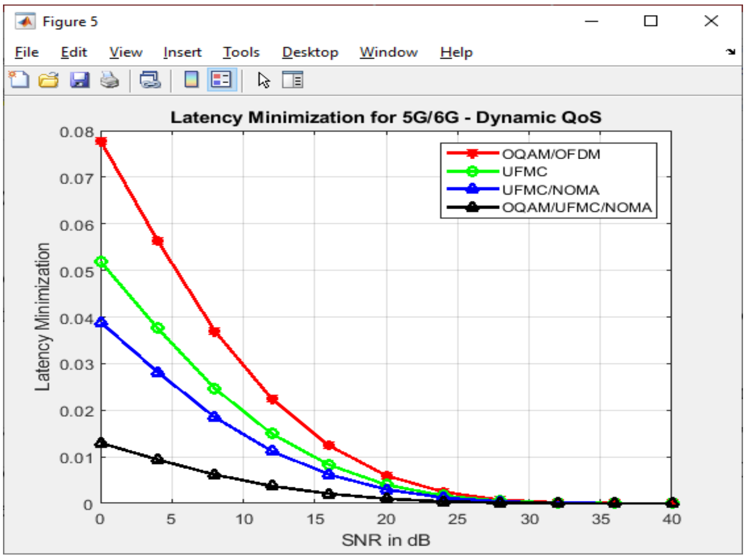

Fig. 10. Latency minimisation analysis

The different latency analysis of the DL-CCA system is depicted in Fig 10. Comparing the method to the other forms of demonstrating latency elimination, the proposed DL-CCA system has the minimum network latency. Consequently, every intermediate node uses less power, and there are a decreased transmission latency thanks to the proposed strategy. The simulation outcomes are evaluated with different SNRs, and the latency minimisation results are plotted for other hybrid systems. As the SNR increases, the DL-CCA system reduces the latency and increases the throughput.

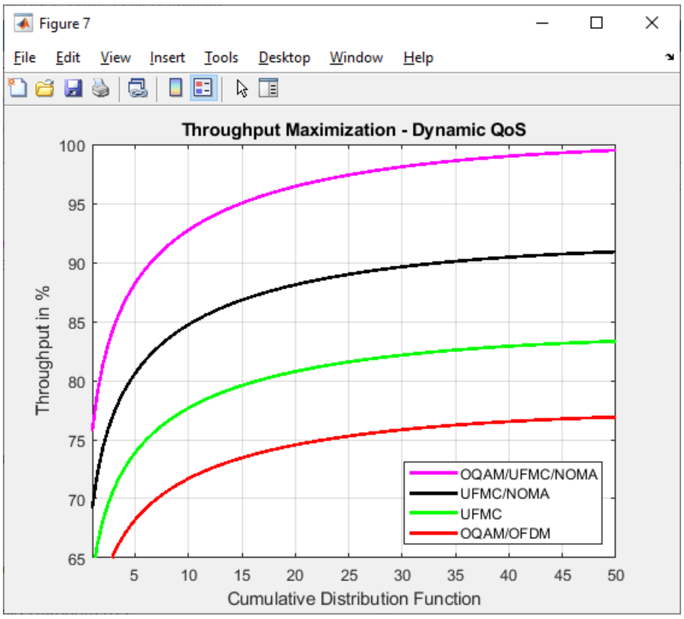

Fig. 11. Throughput evaluation of the DL-CCA system

The throughput analysis and evaluation of the DL-CCA system are shown in Fig. 11. Based on OQAM/UFMC/NOMA, the proposed technique offers a higher percentage efficiency than other existing methods. The cumulative distribution function of the data is increased from 0 to 50 for the carrier spectrum. The distribution function and throughput are directly related to each other. The throughput is computed as the ratio of the number of bits transmitted to the simulation time—the DL-CCA system with a deep learning model and congestion control methodologies.

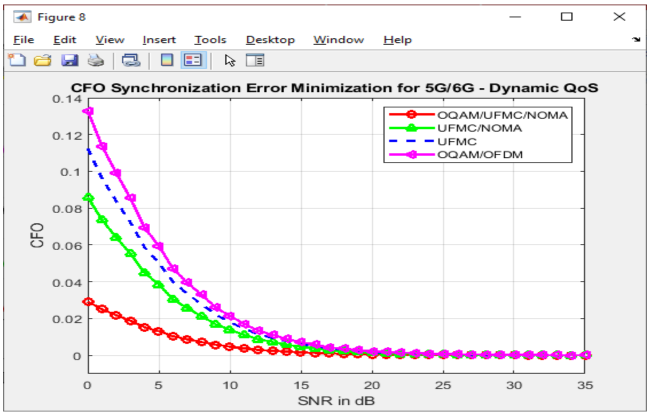

Fig. 12. Carrier frequency offset analysis of the DL-CCA system

The Carrier Frequency Offset (CFO) analysis of the DL-CCA systemis shown in Fig. 12. The signal-to-noise ratio of the modulating signal is increased from a minimum of 0SNR to a maximum of 35SNR with a step count of 1 SNR. The DL-CCA system exhibits higher simulation outcomes than the hybrid NOMA and OFDM systems. The deep learning model and the congestion control method reduce the packet drop,increasing the throughput and reducing the end-to-end latency.

The DL-CCA system is analysed in this section,and the results are compared with the other hybrid systems. The DL-CCA system with NOMA/OFDM enhances the overall performance and reduces interference. The congestion control and deep learning model ensures a higher throughput and lower latency.

5. Conclusion and findings

Increasing network speed and quality methods or combinatorial techniques is necessary due to the capacity control issue many models confront across all networking generations. This paper proposed a new model built on a classification algorithm DT technique to forecast the ideal nodes in 5G/6G IoT settings, addressing the traffic control scheme in 5G/6G IoT surroundings. The suggested method aid in better congestion management and boostsnetwork efficiency. The best congestion control management is based on circumstances like maximum bandwidth, considerable queue length, high congested window, and low network loss.The optimised parameter readings were chosen using a database of node measurements, and the deep learning method used the generated database to estimate future ideal nodes.

Future research compares three deep learning techniques for reducing traffic in 5G/6G IoT scenarios for refinement and improvement. Additionally, it makes an effort to put the best deep learning methods based into practice on equipment and test them in a genuine test-bed setting. Promising recent developments in the creation of predictions and improvement of network protocol stack include 5G/6G settings and computer vision.Other factors are also taken into account for automated systems in order to increase network usage, where merging creates a multi-objective field and results in multidisciplinary realms like artificial intelligence (AI) and personal control, resulting in the best connection speeds.

References

[1]. Kato, T., Bandai, M., & Yamamoto, M. (2018). A congestion control method for named data networking with hop-by-hop window-based approach. IEICE Transactions on Communications.

[2]. Xu, Z., Tang, J., Yin, C., Wang, Y., &Xue, G. (2019). Experience-driven congestion control: When multi-path TCP meets deep reinforcement learning. IEEE Journal on Selected Areas in Communications, 37(6), 1325-1336.

[3]. Raca, D., Zahran, A. H., Sreenan, C. J., Sinha, R. K., Halepovic, E., Jana, R., & Gopalakrishnan, V. (2020). On leveraging machine and deep learning for throughput prediction in cellular networks: Design, performance, and challenges. IEEE Communications Magazine, 58(3), 11-17.

[4]. Zhang, T., & Mao, S. (2020). Machine learning for end-to-end congestion control. IEEE Communications Magazine, 58(6), 52-57.

[5]. Giannakou, A., Dwivedi, D., &Peisert, S. (2020). A machine learning approach for packet loss prediction in science flows. Future Generation Computer Systems, 102, 190-197.

[6]. Regin, R., &Menakadevi, T. (2021). A novel clustering technique to stop congestion occur vehicular ad-hoc networks using node density based on received signal strength. Peer-to-Peer Networking and Applications, 14(4), 2361-2371.

[7]. Zhang, S., & Zhu, D. (2020). Towards artificial intelligence enabled 6G: State of the art, challenges, and opportunities. Computer Networks, 183, 107556.

[8]. González-Landero, F., García-Magariño, I., Lacuesta, R., &Lloret, J. (2018). PriorityNet App: A mobile application for establishing priorities in the context of 5G/6G ultra-dense networks. IEEE Access, 6, 14141-14150.

[9]. AnlineLizie, R., & Gomathy, C. (2021). A Review of Design and Protocol for Smart Continuous Monitoring E-Health Systems in 5G. Advances in Automation, Signal Processing, Instrumentation, and Control, 467-475.

[10]. Huang, Y., Liu, S., Zhang, C., You, X., & Wu, H. (2021). True-data testbed for 5G/B5G/6G intelligent network. Intelligent and Converged Networks, 2(2), 133-149.

[11]. Khan, I., Zafar, M. H., Jan, M. T., Lloret, J., Basheri, M., & Singh, D. (2018). Spectral and energy efficient low-overhead uplink and downlink channel estimation for 5G/6G massive MIMO systems. Entropy, 20(2), 92.

[12]. Chowdhury, T., &Alam, M. J. (2019). Performance Evaluation of TCP Vegas over TCP Reno and TCP NewReno over TCP Reno. JOIV: International Journal on Informatics Visualization, 3(3), 275-282.

[13]. Mishra, A. (2018). Performance analysis of TCP Tahoe, Reno and New Reno for scalable IoT network clusters in QualNet® network simulator. Internl Journal of Computer Sciences and Engineering, 6(8), 347-355.

[14]. Liu, J., Han, Z., & Li, W. (2019). Performance analysis of TCP new Reno over satellite DVB-RCS2 random access links. IEEE Transactions on Wireless Communications, 19(1), 435-446.

[15]. Ahmad, M., Ahmad, A., Jabbar, S., Asif, M., Habib, M. A., Ahmed, S. H., & Shah, S. C. (2018, May). TCP CUBIC: a transport protocol for improving the performance of TCP in long distance high bandwidth cyber-physical systems. In 2018 IEEE International Conference on Communications Workshops (ICC Workshops) (pp. 1-6). IEEE.

[16]. Zhuang, R., Xing, Y., Wei, W., Zhang, Y., Yang, J., &Xue, K. (2021, June). wCompound: Enhancing Performance of Multipath Transmission in High-speed and Long Distance Networks. In 2021 IEEE/ACM 29th International Symposium on Quality of Service (IWQOS) (pp. 1-10). IEEE.

[17]. Pan, Q., Wen, X., Lu, Z., Jing, W., & Zhang, H. (2019). Autonomous and adaptive congestion control for machine-type communication in cellular network. International Journal of Distributed Sensor Networks, 15(4), 1550147719841869.

[18]. Liu, D., Chuai, G., &Gao, W. (2022, January). Dynamic Adaptive Marking Strategy Based on DCTCP in Data Center Networks. In 2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA) (pp. 387-392). IEEE.

[19]. Tsiknas, K. G., Aidinidis, P. I., &Zoiros, K. E. (2021, July). On the Fairness of DCTCP and CUBIC in Cloud Data Center Networks. In 2021 10th International

Conference on Modern Circuits and Systems Technologies (MOCAST) (pp. 1-4). IEEE.

[20]. Jiang, W., Li, H., Peng, L., Wu, J., Ruan, C., & Wang, J. (2021). Analysis and improvement of the latency-based congestion control algorithm DX. Future Generation Computer Systems, 123, 206-218.

[21]. Ye, J., Feng, L., Xie, Z., Huang, J., & Li, X. (2019). Fine-grained congestion control for multipath TCP in data center networks. IEEE Access, 7, 31782-31790.

[22]. Sharma, S., Jindal, D., & Agarwal, R. (2018). Analysing mobile random early detection for congestion control in mobile ad-hoc network. International Journal of Electrical and Computer Engineering, 8(3), 1305.

[23]. Carlucci, G., De Cicco, L., &Mascolo, S. (2018). Controlling queuing delays for real-time communication: the interplay of E2E and AQM algorithms. ACM SIGCOMM Computer Communication Review, 46(3), 1-7.

[24]. Pashmforoush, S., &EmraniZarandi, A. A. (2021). A modified continuous-time Markov chain for the prioritised spectrum access over cognitive radio ad-hoc networks. International Journal of Electronics, 108(9), 1519-1533.

[25]. Majumder, T., Mishra, R. K., Singh, S. S., &Sahu, P. K. (2020). Robust congestion control in cognitive radio network using event-triggered sliding mode based on reaching laws. Journal of the Franklin Institute, 357(11), 7399-7422.

[26]. Sun, G., Li, C., Ma, Y., Li, S., &Qiu, J. (2022). End-to-End TCP Congestion Control as a Classification Problem. IEEE Transactions on Reliability.

[27]. Li, W., Zhang, H., Gao, S., Xue, C., Wang, X., & Lu, S. (2019). SmartCC: A reinforcement learning approach for multipath TCP congestion control in heterogeneous networks. IEEE Journal on Selected Areas in Communications, 37(11), 2621-2633.

[28]. Xu, Z., Tang, J., Yin, C., Wang, Y., &Xue, G. (2019). Experience-driven congestion control: When multi-path TCP meets deep reinforcement learning. IEEE Journal on Selected Areas in Communications, 37(6), 1325-1336.

[29]. Li, W., Zhou, F., Chowdhury, K. R., &Meleis, W. (2018). QTCP: Adaptive congestion control with reinforcement learning. IEEE Transactions on Network Science and Engineering, 6(3), 445-458.