Performance Improvement of Time-Sensitive Fronthaul Networks in 5G Cloud-RANs Using Reinforcement Learning-Based Scheduling Scheme

Waqar, M., Mustafa, M. U., Jabeen, F. & Shah, S. A.

Original citation:
Waqar, M, Mustafa, MU, Jabeen, F & Shah, SA 2024, 'Performance Improvement of Time-Sensitive Fronthaul Networks in 5G Cloud-RANs Using Reinforcement Learning-Based Scheduling Scheme', IEEE Access, vol. 12, pp. 59756-59770.
https://dx.doi.org/10.1109/access.2024.3393849

DOI 10.1109/access.2024.3393849
ISSN 2169-3536

Publisher: Institute of Electrical and Electronics Engineers (IEEE)

IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 23 March 2024, accepted 17 April 2024, date of publication 25 April 2024, date of current version 3 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3393849

RESEARCH ARTICLE

Performance Improvement of Time-Sensitive Fronthaul Networks in 5G Cloud-RANs Using Reinforcement Learning-Based Scheduling Scheme

MUHAMMAD WAQAR [1], **MUHAMMAD USMAN MUSTAFA** [2], **FARHANA JABEEN** [2], **AND SYED AZIZ SHAH** [3]

[1]School of Technology, Business and Arts, University of Suffolk, IP4 1QJ Ipswich, U.K.
[2]Department of Computer Science, COMSATS University Islamabad, Islamabad 44000, Pakistan
[3]Research Centre for Intelligent Healthcare, Coventry University, CV1 5RW Coventry, U.K.

Corresponding author: Muhammad Waqar (m.waqar@uos.ac.uk)

ABSTRACT The rapid surge in internet-driven smart devices and bandwidth-hungry multimedia applications demands high-capacity internet services and low latencies during connectivity. Cloud radio access networks (C-RANs) are considered the prominent solution to meet the stringent requirements of fifth-generation (5G) and beyond networks by deploying the fronthaul transport links between baseband units (BBUs) and remote radio heads (RRHs). High-capacity optical links could be conventional mainstream technology for deploying the fronthaul in C-RANs. But densification of optical links significantly increases the cost and imposes several design challenges on fronthaul architecture which makes them impractical. Contrary, Ethernet-based fronthaul links can be lucrative solutions for connecting the BBUs and RRHs but are inadequate to meet the rigorous end-to-end delays, jitter, and bandwidth requirements of fronthaul networks. This is because of the inefficient resource allocation and congestion control schemes for the capacity constraint Ethernet-based fronthaul links. In this research, a novel reinforcement learning-based optimal resource allocation scheme has been proposed which eradicates the congestion and improves the latencies to make the capacity-constraints low-cost Ethernet a suitable solution for the fronthaul networks. The experiment results verified a notable 50% improvement in reducing delay and jitter as compared to the existing schemes. Furthermore, the proposed scheme demonstrated an enhancement of up to 70% in addressing conflicting time slots and minimizing packet loss ratios. Hence, the proposed scheme outperforms the existing state-of-the-art resource allocation techniques to satisfy the stringent performance demands of fronthaul networks.

INDEX TERMS 5G, end-to-end delays, fronthaul networks, jitter, resource allocation.

I. INTRODUCTION

The exponential growth of smart devices such as cellular phones, tablets, and the internet of things (IoT) imposes strict requirements on next-generation wireless networks (NGWNs) in terms of guaranteed high capacity, reliability, and ultra-low latency [1]. The NGWNs are expected to fulfill these demands by implementing new technologies such as heterogeneous networks (HetNets) [2] and cloud radio access networks (C-RANs) [3]. The HetNets are compromises of small cells such as Micro, Pico, and Femto cells along with the Wi-Fi, RRHs, and relays to provide high reliability and network throughput. The HetNets faced high inter-tier interference due to the deployment of a large number of small cells which degrades the overall network performance. As a result, the benefits of HetNets are limited by the effect of high interference [4].

The most effective solution to mitigate such effects is implementing coordination techniques among the base stations. The cloud radio access networks are proposed

The associate editor coordinating the review of this manuscript and approving it for publication was Stefan Schwarz.

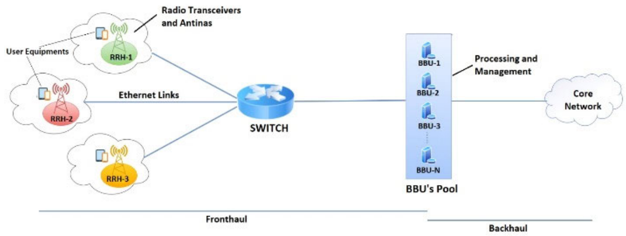

FIGURE 1. Fronthaul and backhaul in 5G C-RANs.

to minimize the impact of interference and meet the performance requirements of NGWNs without performance degradations. The C-RANs are promising solution that ensures operational, computational, and energy cost savings for the operator and improves the coordination among the multi-point units, coverage, power consumption, multiplexing gains, and the inter-cell interference as well as total cost of ownership (TCO) of mobile networks [5].

The C-RAN achieves the aforementioned objectives by separating the functionalities of the baseband unit at the cell site, distributing them between RRHs (remote radio heads) and BBU (baseband unit) pool. As illustrated in Figure 1, the RRHs are strategically positioned in target areas to serve user equipment (UE). The RRHs perform functions such as power amplification, digital signal processing, digital-to-analog conversion, filtering, and a few more. The BBU assumes a centralized control role, processing received signals and allocating resources on demand. The BBU possesses a comprehensive network view, ensuring dynamic resource allocation at high speed through centralized processing. All these facilities enable the implementation of inter-cell interference using the ICIC (inter-cell interference coordination) and CoMP (coordinated multipoint) schemes [6]. It combines the BBUs in a powerful centralized computing infrastructure termed as a BBU pool. The RRHs and BBU pool communicate with each other through a high capacity and low latency transport link named as fronthaul. The fronthaul networks are connected to the core networks via the backhaul. Despite offering high reliability and low TCO, the C-RAN architecture faces challenges in implementing fronthaul links, limiting the commercial deployment of C-RAN infrastructure.

The 3rd generation partnership project (3GPP) [7] has defined eight functional splits between the RRHs and BBU pool for deploying fronthaul links. Each split has its own set of network conditions and services for the fronthaul link. In the low-level split, known as option 1, all control functionalities are shifted to the RRHs, with the number of functionalities decreasing at each higher split level. For instance, at the highest split level, namely option 8, only the radio frequency function can be implemented at the RRHs, and all other functionalities are transferred to the centralized BBU pool. Option 8 is regarded as the optimal approach for implementing fronthaul networks [8]. However, it imposes strict requirements on the link, including high capacity, end-to-end (E2E) latency less than 250 μs, and jitter less than 60 ns [9]. This scenario also limits the multiplexing gains which represents the transmission of multiple RRHs traffic simultaneously over a fronthaul link. These requirements can only be fulfilled by deploying high-capacity optical links, that significantly increase both capital expenditure (CAPEX) and operational expenditure (OPEX), hindering C-RANs from realizing their cost-saving benefits [10]. Although wireless fronthaul may be a cost-effective and desirable alternative, its limited range and bandwidth necessitate multiple access points for larger coverage areas, consequently increasing the operational costs of the network [11]. Recently, several studies have explored this problem and suggested the Ethernet as an alternative cost-effective solution to carry the traffic over the fronthaul [12], [13]. Although Ethernet-based fronthaul could be cost-effective but unable to meet the strict fronthaul traffic demands due to limited capacity, high end-to-end latency, and jitter [14], [15]. Various techniques, such as deep reinforcement learning (RL), time division multiplexing, and multiple description coding, have been proposed for congestion control and resource allocation to enhance latency and jitter in Ethernet-based fronthaul networks [16], [17], [18].

IEEE 1914.3 [19] and IEEE P802.1CM [20] propose the mechanism and standards to carry the multiple time-sensitive traffic streams over the Ethernet fronthaul networks. Like our proposed scheme, the studies CFIT [21] and DTSA [22] implemented the fronthaul scenarios based on the 3GPP [7] and IEEE standards [19], [20]. A study [21] proposes a scheduling algorithm named CFIT to transmit multiple common public radio interface (CPRI) streams over a single switching network. Several simulations were conducted to assess jitter levels under various load conditions for functional split level 8. CPRI serves as an interface for carrying oversampled in-phase/quadrature-phase (I/Q) signals across the fronthaul interface. CPRI is independent of traffic fluctuations and operates at a constant bit rate, adversely affecting the transmission efficiency for carrying time-sensitive traffic streams in optical fronthaul networks. Table 1 represents the different CPRI line rates for the fronthaul networks [21]. This study indicates that a satisfactory level of jitter can be maintained when the load-to-Ethernet ratio (LER) is low. However, jitter increases significantly when the LER exceeds 0.35. LER values below 0.35 correspond to lower CPRI line rates and fewer input combinations. The network's performance degrades more when the incoming CPRI line rates are not multiples of each other. Multiplexing non-multiple flows in the network results in intolerable jitter fluctuations. This happens because the scheme allocates non-conflicting timeslots exclusively to flows that are perfect multiples of each other [22]. As a result, a considerable number of conflicting slots are assigned to non-multiple flows, causing increased congestion, and subsequently leading to heightened end-to-end delays and jitter.

TABLE 1. CPRI line rates.

Options	CPRI Line Rates (Mbps)
Option 1	614.4
Option 2	1228.8
Option 3	2457.6
Option 4	3072.0
Option 5	4915.2
Option 6	6144.0
Option 7	9830.4
Option 8	10137.6
Option 9	12165.1
Option 10	24330.2

In a recent study [22] another scheduling algorithm, DTSA has been proposed to prioritize and allocate network resources depending on traffic characteristics. This algorithm exhibits low jitter when high-capacity optical links are utilized with multiple wavelength calculations. However, its performance could be degraded when Ethernet links without additional wavelengths are employed. Furthermore, the algorithm prioritizes flows, but a drawback of DTSA is the potential starvation of low-priority flow (LPF) packets in the presence of continuous high-priority flow (HPF) packets in the queue. Since DTSA prioritizes HPF traffic, which already consumes a significant portion of network resources, it leaves minimal or less than the required bandwidth for LPF traffic. This prioritization of HPF traffic can result in higher queuing delays for LPF traffic, leading to increased delays, jitter, and packet losses for LPF traffic. Consequently, limiting the RRHs traffic flows in capacity-constraint Ethernet-based fronthaul, reduces the multiplexing gains.

To address the aforementioned challenges of scheduling algorithms to carry more RRHs traffic streams over limited capacity Ethernet-based fronthaul networks, there is a need for a novel traffic scheduling and congestion control scheme capable of ensuring high multiplexing gains at acceptable latency and jitter values. This research proposes a Q-learning-based reinforcement learning algorithm that is capable of allocating non-conflicting time slots to a number of RRHs traffic flows to improve the multiplexing gains without degrading the network performance. The simulation results verify that the proposed scheme enables the Ethernet links to successfully transport the CPRI traffic in fronthaul networks. The contributions of this research work can be summarized as:

- This study proposes a reinforcement learning-based scheduling algorithm to ensure effective congestion control and resource allocation in fronthaul networks.
- The study verifies that a number of RRHs' traffic flows can be transmitted in fronthaul networks without degrading the quality of service (QoS) requirements, thereby improving multiplexing gains.
- The study confirms that cost-effective Ethernet links can carry time-sensitive traffic between the RRHs and BBU pool at tolerable latency and jitter values, resulting in reduced CAPEX and OPEX.
- Experimental results validate significant enhancements in the reduction of conflicting time slots and packet loss ratios, thereby diminishing the need for packet retransmission. This improvement leads to increased bandwidth efficiency, minimizing waste and freeing up additional bandwidth for fronthaul traffic.
- The performance of the proposed schemes is evaluated in terms of average delay, end-to-end delay, jitter, number of conflicting slots, and packet loss ratio under different traffic loads. A comparison with state-of-the-art schemes is also presented.

The subsequent sections of the paper are structured as follows: Section II provides a summary of related works on fronthaul congestion control and resource allocation. Section III presents the Q-learning based reinforcement learning scheme proposed to address the congestion control and resource allocation problem. The performance evaluation of the proposed scheme is discussed in Section IV along with a detailed comparison with existing schemes. Section V presents the conclusion of the paper.

II. RELATED WORK

The C-RAN represents a highly promising architectural framework for next-generation cellular networks. However, the practical implementation of C-RAN necessitates the provision of substantial fronthaul capacity to effectively accommodate the increased bandwidth requirements between the BBUs and RRHs. Recently, several research studies have been initiated to improve the performance of fronthaul networks in C-RAN architectures. These studies were conducted to optimize the performance of fronthaul networks in terms of bandwidth utilization, minimizing the E2E delays, and inter-packet delay variations to improve jitter and overall network performance. In a research study [23], a multi-wavelength C-RAN architecture was designed, and a packet scheduling method based on reinforcement learning was proposed for allocating the network resources in the uplink direction. This study shows that using the proposed method significant traffic could be transmitted in the uplink direction by effectively allocating the bandwidth resources. However, this study has the limitation of utilizing significant training data and high computational resources.

In a study [24], a network architecture has been proposed to fulfill the fronthaul network requirements. In this work, concepts of cloud and edge computing are utilized to address the challenges of bottlenecks in the networks and multiple transmissions. This study suggested utilizing the long short-term memory (LSTM) framework to predict the network throughput and genetic algorithm (GA) for resource allocation. Research findings indicate that this method effectively reduces baseband unit migrations, resulting in decreased power consumption. Moreover, resources are allocated based on predictive data, enhancing efficiency. However, the approach does not significantly reduce computation time and can be further improved for increased throughput prediction and accuracy. Another study [25], introduced a novel methodology for managing traffic entering 5G mobile fronthaul networks. Its primary objective was to efficiently handle multimedia services when dealing with large volumes of data, particularly when dealing with multimedia streaming services such as YouTube. The proposed solution includes placing a traffic control unit (TCU) near its boundary to monitor and analyze the incoming traffic flows. By gathering relevant information, the TCU can make informed decisions regarding resource allocation. This ensures the provision of high-quality multimedia services within the network. However, this study lacks in providing the required experimental results for the evaluation of schemes. In [26], a novel approach employing deep reinforcement learning, specifically based on policy gradients, was introduced to tackle the issue of function split in virtualized radio access networks (VRAN). This approach reduces the need to train the learning algorithm to generate a random strategy. The objective of this approach is to identify whether the base station functionalities could be implemented at the cloud unit or distributed unit. This study considers several parameters including the server, link bandwidths, and latency needs in virtualized radio access networks. If the learning algorithm does not meet these constraints it is considered as an outcome and the algorithm modifies its strategy accordingly. The simulation result verifies that the proposed scheme is efficient in terms of learning and producing optimal results and achieves the 0.4% optimal gap which surpasses the distributed radio access network (D-RAN) approach. The study claims to significantly improve the overall computation cost as compared to the D-RAN. However, this work emphasized the functional splitting in VRANs and could be more effective by including factors like network optimization, network slicing, and mobility management.

The authors in [27] proposed software-defined networking (SDN) and software-defined wireless networking (SDWN) methodologies to design flexible, reprogrammable, and robust (H-CRAN) infrastructures for next-generation cellular networks. This study aims to effectively join SDWN and H-CRAN concepts to design a sophisticated cellular network. This research discusses the case studies related to the H-CRAN using the SDWN controller for assigning the RRH channels to user equipment. However, this work relies on theoretical analysis, lacking substantial empirical data to strengthen the results. Moreover, the paper does not extensively discuss jitter and network delays, focusing more on the aspects of SDWN, H-CRAN, and their interoperability.

In a research study [28], a utility-based algorithm is introduced for the H-CRANs, combining power control and resource allocation. This algorithm anticipates dynamic loads on BBUs, RRHs, and microcell base stations (MBSs). It determines the data rate for each UE connected to RRHs and MBSs on the resource block, prioritizing the utility function. UEs consider predicted data rates and projected energy usage, opting to connect to either RRHs or MBSs. In scenarios of high-priority traffic, UEs connected to MBSs and failing to meet predefined data rate requirements have the option to request any available RRH for allocating remaining resource blocks (RBs). Comparative analysis indicates that the joint resource allocation and user association (JRAUA) algorithm delivers improved bandwidth and resource utilization. However, it exhibits lower power efficiency and may encounter higher packet loss ratios. This study increases the complexity of implementing practical fronthaul networks. Furthermore, this work assumes the exact knowledge of channel state information (CSI) and does not well consider the other parameters such as channel estimation errors that can occur in real-world networks.

The study [29] proposes a novel scheme to improve the congestion in 5G wireless networks. This scheme is based on a machine learning scheme specifically policy distillation that involves training a small-sized neural network to imitate the behavior of a large-sized neural network. Specifically, in the context of fronthaul congestion control, the authors use policy distillation to train a smaller neural network to predict the congestion status of the network based on real-time measurements of network traffic. The proposed distilled decision trees (DTs) demonstrate poorer generalization ability compared to the twin-delayed deep deterministic policy gradient (TD3) policy. Further, the distilled DTs can complicate the management of the policy life-cycle due to reduced flexibility in adapting to evolving network conditions. The study [30] introduces a methodology aimed at improving the energy efficiency of C-RANs by optimizing the performance of capacity-constrained fronthaul links. This proposed approach concentrates on the joint optimization of resource allocation and fronthaul compression to minimize energy consumption while meeting the quality-of-service requirements of users. However, a significant limitation of this research is its reliance on a simplified network model, assuming that the users are connected to a single RRH. In practical scenarios, C-RANs typically include multiple RRHs and serve users across a broader geographic area. Consequently, the proposed methodology may not be directly applicable to such complex network scenarios. In [31], the authors address the resource allocation problem by proposing the BMF and CDJM schemes. The author utilizes the slots assignment and fixed transmission time mechanism to improve the performance of Ethernet-based 5G fronthaul networks. This scheme requires high queue sizes at switches to effectively perform the successful transmissions of the CPRI packets in the networks.

The paper [32] introduces an approach for resource allocation in Ethernet-based 5G networks that utilize shared radio units with limited fronthaul capacity. In these networks, multiple base stations share a single radio unit, and the available fronthaul capacity is constrained. The primary aim of the proposed scheme is to optimize the allocation of resources, both in terms of fronthaul and radio, to achieve the best possible performance for the network. However, the primary limitation of this approach is the constrained fronthaul capacity. Fronthaul refers to the network connection between the base station and the centralized processing unit. When multiple base stations share a single radio unit, the available fronthaul capacity might not be sufficient to handle the increased traffic demands. With limited fronthaul capacity, resource contention becomes a significant concern. Multiple base stations competing for limited resources can result in inefficient resource allocation and increased latency. This limitation can affect the network's ability to handle high traffic loads and meet performance requirements. In a study [33], for an Ethernet-based fronthaul network, a scheduling algorithm is proposed with the goal of minimizing jitters and enhancing the quality of service for 5G networks. This proposed algorithm is known as time division multiplexing (TDM) slot-based scheduling (TSS), which integrates both time-division multiple access (TDMA) and flow control techniques for an effective slot allocation to multiple flows. Moreover, virtual queuing is used to overcome the negative impact of packet reordering on jitter. The algorithm consists of two components such as slot allocation, which deals with challenges in optimal slot assignment, and low delay collaboration, which ensures smooth communication between different switches to reduce delays linked to strict slot assignments. However, the simulations carried out to evaluate this algorithm are only compared to an existing algorithm and could be compared with more schemes for comprehensive assessment. Additionally implementing this algorithm will require additional execution time and resources for slot allocation, flow control, and virtual queuing. This could lead to increased latency and wasting the limited network capacity, especially in 5G Ethernet networks. Furthermore, it could be less applicable in highly congested scenarios with variable traffic patterns such as in Ethernet-based fronthaul.

In [34] the authors have introduced an approach to enhance 5G fronthaul networks that focuses on improving their performance. This approach set limits, on delay and variation in delay known as jitter. To achieve this goal the method uses a framework that takes into account the characteristics of network traffic. By making adjustments to transmission parameters like packet size and time intervals between packets the aim is to enhance delay efficiency. Moreover, it incorporates a model for predicting delays to anticipate and adjust for delays and variations in packets. This predictive technology enables the method to adapt proactively to changing network conditions and optimize performance. However, it may face difficulties when sudden significant changes occur, such as network congestion or unexpected spikes in traffic volume. When such a situation occurs, ensuring the proper functioning of mechanisms to control packet delay and variation is often challenging. The efficiency of the mechanism is dependent on an accurate analysis of traffic parameters and rules. The adaptive framework of the mechanism heavily relies on an accurate analysis of network traffic characteristics. Any inaccuracies or incomplete understanding of traffic patterns can lead to suboptimal adjustments of transmission parameters which could potentially compromise delay performance in fronthaul.

In [35], an effective congestion control scheme for fronthaul networks based on MDC has been proposed, which makes it possible to work well in the highest load conditions. As a result of the proposed technique, the signal source is divided into two or more distinct sub-streams, each containing partial information about the baseband signal, independently transmitted over a fronthaul network. In this manner, each of these descriptions are characterized by carrying different components of the signal source, and therefore loss or delay in one description does not affect the whole recovered at the receiver from the remaining descriptions. The proposed technique is developed to improve congestion control by adding an adaptive rate control method. The method adjusts the transmission rates of the flows based on the network resource variations so that congestion occurrence is prevented as much as possible. By controlling traffic flows with variable transmission rates, the proposed scheme aims to optimize resource allocation and prevent performance degradation. For the fronthaul networks, the use of such a technique can result in the increased average end-to-end delay from RRHs to BBU pools. The additional processing and transmission cost in switching to multiple descriptions and adaptive rate control could further increase the delays. This technique could be made more suitable by balancing the tradeoff between the congestion control mechanism and expected end-to-end delay. By improving the delays in such techniques could be effective in deploying the Ethernet-based fronthaul in the cloud radio access networks.

In [36] a mobile fronthaul network delay-aware packet scheduling scheme has been proposed. This study split a network into high-priority and low-priority segments, with time-critical traffic such as voice and video being moved towards the high-priority segment while less time-sensitive traffic such as web traffic is moved to the low-priority part. The scheme makes use of a weighted fair queuing algorithm that tunes the transmission rate according to traffic loads and packet time sensitivities. The objective of the scheme is to minimize delays to meet the performance requirements of the system. However, this scheme lacks to effectively respond to rapidly changing network conditions, such as fluctuating traffic patterns, varying network congestion levels, or dynamic resource availability as required in fronthaul networks.

Despite several attempts to improve the performance of Ethernet-based fronthaul networks, existing studies lack an efficient methodology to improve the E2E delays and jitter. There are tradeoffs between the number of traffic flows and the performance requirements of the fronthaul networks. Therefore, a novel and dynamic scheme that can fulfill the QoS requirements and improve the multiplexing gains in fronthaul networks is necessary.

III. PROPOSED METHODOLOGIES

A. PROPOSED Q-LEARNING SCHEME

Q-learning is a key reinforcement learning technique where an agent learns to maximize its rewards by iteratively exploring an environment, taking actions, and updating its action choices based on the expected future rewards associated with different states and actions. Through this process, the agent learns to make smarter choices by figuring out which actions are better in different situations. It does this by learning from its experiences and gradually getting better at picking actions that bring in more rewards over the long term, all while finding the right balance between trying new actions and sticking to what it knows. In the context of congestion control and resource allocation in Ethernet-based fronthaul networks, a Q-learning-based reinforcement learning technique can be employed to optimize the network's performance. This involves the agent learning to make decisions that minimize congestion and efficiently allocate resources. Given the absence of labeled data, the agent can only enhance its knowledge through experiential learning. In Figure 2 the Q-learning based RL algorithm has been explained in which, the first challenge is to identify the problem, in this case, the problem is to control congestion and allocate resources in the fronthaul network to maximize its performance. Then the state space is defined which denotes a collection of variables that define the present state of the environment. In this case, the state space includes variables such as the number of RRHs, the bandwidth utilization, the delay of the network, and other relevant factors. Following that, the action space is defined which comprises all the feasible actions that an agent can undertake within the environment as shown in Figure 2. In this context, the action space includes actions such as fixing the number of RRHs, adjusting the transmission rate, changing the priority of different RRHs depending upon their traffic flows, and allocation of resources such as bandwidth to different RRHs, among other possibilities. Then the reward function is identified which determines the reward for an agent based on its actions within a given environment. For this particular scenario, this function derives its value from various metrics, such as network bandwidth, end-to-end packet delay, and the number of packet losses. Once the state space, action space, and reward function are defined, then the RL agent undergoes training. During training, the agent learns to take actions that maximize its expected reward in the environment. After training the RL agent, the performance is evaluated to see how well it performs in controlling congestion and allocating resources in the fronthaul networks. The best-rewarded action is followed until the state function changes and the same process is repeated until the optimal reward is achieved again.

The proposed mechanism comprises five steps such as calculating packets transmitted by each RRH, determining RRH preference, identifying possible packet flow sequences, managing packet flows based on available bandwidth, and ensuring efficient utilization and prioritization. The mechanism begins by calculating the total number of packet flows

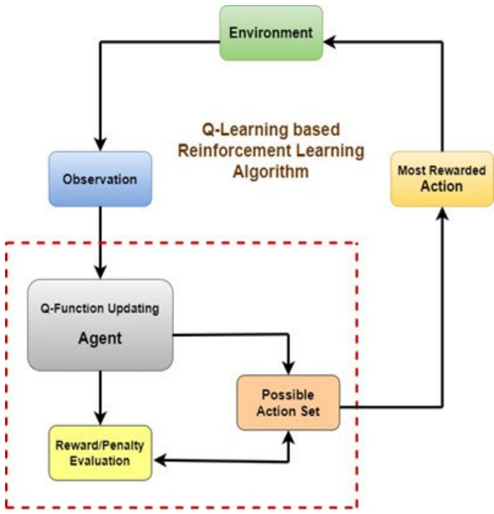

FIGURE 2. Q-Learning based reinforcement learning algorithm.

originating from multiple RRHs. The number of packets transmitted by each RRH can be determined by monitoring the packet header or using flow control mechanisms, typically over a fixed time interval. This study assumed that the CPRI line rates per RRHs are pre-determined [21]. Subsequently, all possible packet flow sequences are identified based on the number of RRHs and available bandwidth. By managing packet flows in accordance with available bandwidth, the mechanism ensures efficient utilization of network resources. This approach also facilitates the provision of sufficient bandwidth to high-priority traffic, while limiting lower-priority flows to a lower bandwidth. Additionally, the utilization of appropriate bandwidth management techniques helps prevent congestion and network performance issues that may arise when network traffic exceeds the available bandwidth. Different packet flow sequences are applied and identified to check which sequence has the jitter within the required range of 64 ns. Jitter refers to the variation in the delay between packets sent over a network. Equation 1 represents the jitter that is disparity between the highest inter-packet delay (HID) and lowest inter-packet delays (LID). The calculation of jitter in a packet stream is performed using Equation 2.

$$Jitter = HID - LID \tag{1}$$

$$Jitter = (|D(i) - D(i-1)| - Avg)/(n-1) \tag{2}$$

where D(i) is the time delay of the ith packet, D(i-1) denotes the delay to the previous consecutive packet, n is the total number of packets in the sample and Avg is the average delay of all packets in the sample. This formula calculates the variation between the delay of each packet and the average delay of all packets in the sample. The absolute value of this variation is taken, and the sum of all these absolute values is divided by n-1 to get the average deviation from the mean. This value represents the jitter in the packet stream.

The E2E delay serves as the second evaluation matrix for comparing performance with existing schemes. It is composed of five parameters, as discussed below, and illustrated

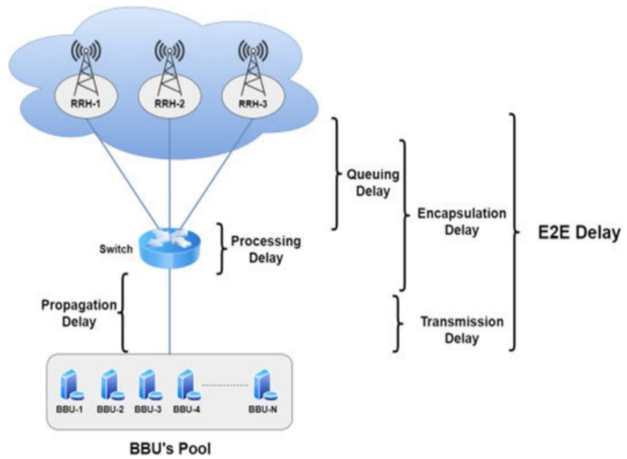

FIGURE 3. Components of delay in fronthaul network.

in Figure 3. These parameters collectively contribute to the overall delay encountered by a packet as it traverses from the RRHs to the BBU pool.

1) QUEUING DELAY

It refers to the time that a packet spends waiting in a queue at a router or switch before being transmitted to its destination. Queuing delay is influenced by several factors, including the arrival rate of packets, the queue size, the processing speed of the node, and the priority of the packets in the queue. The longer the queue, the longer the packets will wait, leading to a higher queuing delay. The first packet encounters no queuing delay as there are no preceding packets. The second packet experiences a queuing delay of L/R seconds. Subsequently, for the third packet, the queuing delay increases to 2L/R seconds due to the presence of two preceding packets. This pattern continues, with the queuing delay for the Nth packet becoming (N-1)L/R seconds. The average queuing delay (AQD) can be calculated using Equations 3 to 5.

$$AQD = \frac{\frac{L}{R} + \frac{2L}{R} + \frac{3L}{R} + \ldots \frac{(N-1)L}{R}}{N} \tag{3}$$

$$AQD = \frac{N \times \frac{(N-1)}{L}}{2.R.N} \tag{4}$$

$$AQD = \frac{\frac{N-1}{L}}{2 \times R} \tag{5}$$

where L represents the size of a packet in bits, R is the bandwidth in Gbps. N represents the total number of packets and depends on the number of flows.

2) ENCAPSULATION DELAY

The delay occurs when a packet is processed and prepared for transmission by adding headers and trailers to the data. This operation is known as encapsulation, and the time taken to perform it is referred to as encapsulation delay. This can be calculated using Equation 6.

$$Encapsulation\ Delay(ED) = \frac{L}{CLR} \tag{6}$$

where L is the size of the packet in bits and CLR is the CPRI line rate in Gbps which is selected from options 1 to 9 from Table 1.

3) PROCESSING DELAY

Packet processing time refers to the duration required for a network device, such as a router or a switch, to handle a packet before forwarding it to the next node in the network. This processing time is influenced by the processor's speed, including operations such as bit-level error detection during transmission. The processing delay (PrD) depends on the speed of the device and is considered as 300 ns [37].

4) TRANSMISSION DELAY

Transmission delay refers to the time required for a packet to travel through a physical medium, which can be either a wired or wireless network. The transmission delay depends on the packet's size and the bandwidth of the link. The transmission delay has been computed using Equation 7.

$$Tranmission\ Delay(TD) = \frac{L}{R} \tag{7}$$

where L is the size of a packet in bits, R is the available bandwidth in Gbps. In this study packet size of 1500 bytes and the bandwidth of 10 Gbps are considered.

5) PROPAGATION DELAY

Propagation delay (PD) is the time it takes for a data packet, to travel through a transmission medium from its source to its destination. It represents the delay introduced by the physical properties of the medium, including its length and the speed at which signals propagate within it. Propagation delay has been calculated by using Equation 8.

$$Propagation\ Delay(PD) = \frac{D}{S} \tag{8}$$

where D represents the distance between the RRH and BBU pool, while S denotes the propagation speed. This study considers the distance of 10 Km between RRHs and BBU.

Once the above delay components are calculated then the total E2E delay using Equation 9 is calculated which is the sum of all the above parameters.

$$E2E\ Delay = AQD + ED + PrD + TD + PD \tag{9}$$

B. PROPOSED CONGESTION CONTROL AND RESOURCE ALLOCATION (CCRA) SCHEME

The proposed CCRA scheme is illustrated in Figure 4 and outlined in Algorithm 1 initiates by calculating the aggregate number of flows originating from multiple RRHs within the fronthaul network. Each flow is associated with a specific CPRI line rate, which is predetermined and falls within the range of options 1 to 9, as illustrated in Table 1. Subsequently, the RRHs are prioritized based on their respective line rates. An optimal window size is then computed, considering the ratio between the total number

TABLE 2. **RRHs configurations for CCRA scheme.**

Total Number of RRHs	Maximum Preferred Number of RRHs	Ratio between Preferred and Non-preferred RRHs	Number of packet flows having jitter=0
4	2	4:2	16
5	3	5:3	32
6	4	6:4	64
7	5	7:5	128
8	6	8:6	256
9	7	9:7	512
10	8	10:8	1024

of packets and the preferred number of packets, as outlined in Table 2. The calculated window size is taken into consideration when permitting a defined number of packets to be transmitted towards the BBUs pool, ensuring a consistent allocation across a fixed number of time slots. Each packet within a flow is assigned a specific time slot based on the preference of the corresponding RRH traffic flow at its origin. This allocation aims to prevent conflicts and congestion, ultimately minimizing inter-packet delays. Subsequently, packet flow sequences are chosen depending on whether their jitter falls within the fronthaul requirements. Jitter, denoting the variation in packet arrival times, is calculated using Equations 1 and 2. Only packet flows exhibiting jitter values less than 64 ns are considered, as a stringent fronthaul requirement that the jitter must not exceed this threshold. After computing the jitter, using Equation 9, E2E delays are determined for all flows with jitter less than 64 ns. Reward values are then assigned to the flows based on their respective E2E delays, with priority given to the highest to lowest delay order and decremented as delays increase. This iterative process continues until the sequence of packet flow with the highest reward is achieved. The highest rewarded packet flow sequence is maintained until the traffic flow is changed. Once the traffic flow changes, the aforementioned procedure is repeated until the best-rewarded sequence is achieved again.

Following the scheduling process, packets can be transmitted within predetermined and conflict-free time slots [21]. This approach ensures consistent inter-packet delays, effectively addressing and resolving jitter issues. This significantly minimizes packet loss resulting from congestion and slot unavailability, thereby reducing the necessity for packet retransmission. Consequently, the proposed scheme markedly enhances the quality of service in Ethernet networks, rendering them well-suited for efficiently transporting CPRI traffic between RRHs and BBU pools in fronthaul networks by meeting the stringent delay and jitter requirements. Table 2 provides the relationship between the total number of RRHs and the preferred number of RRHs in terms of ratio. It also calculates and presents the maximum number of packets in the packet flow sequence that satisfies the jitter requirements for different numbers of RRHs. The

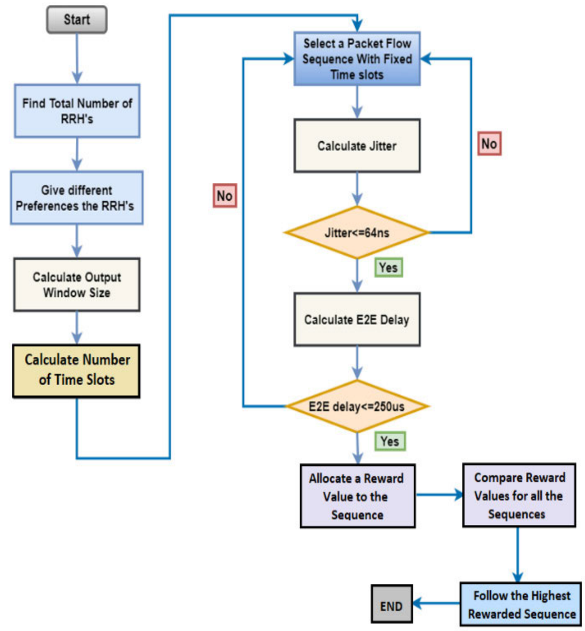

FIGURE 4. **Flow chart of CCRA scheme.**

Algorithm 1 Congestion Control and Resource Allocation (CCRA) Scheme

Input: No of Packets (N), PacketSize (L), DataRate (R), Distance (D) and LinkSpeed (S).
Output: Best Rewarded Packet Flow Sequence

```
Select TotalNoOfRRHs()
Set PreferenceOfRRHs
Select PrefferedNoOfRRHs()
Compute OutputWindowSize()
Calculate NoOfTimeSlots
Print AllPacketsFlowSequences()
Select PacketFlowSequence()
Procedure BestPacketFlow()
  Calculate Jitter()
  if (Jitter <= 64 ns) then
        Save PacketFlowSequence()
        Calculate E2Edelay()
  else Repeat Procedure
  end if
  For (E2Edelay < 250 μs) do
     Set RewardValue()
     Compare RewardValues()
     If (RewardValue > PrevRewardValues) then
        Update PacketFlowSequence()
        Use UpdatedPacketFlowSequence()
      else Retain PrevPacketFlowSequence() and
      Repeat Procedure()
        end if
  end For
end Procedure()
```

output window size varies accordingly, depending on the total number of RRHs and their preferences. As the total number of RRHs increases, the output window size also increases to prevent congestion and packet loss.

For instance, consider scenario 1 in Table 2 with four RRHs (R1, R2, R3, and R4). In this scenario, R1 and R2 are given

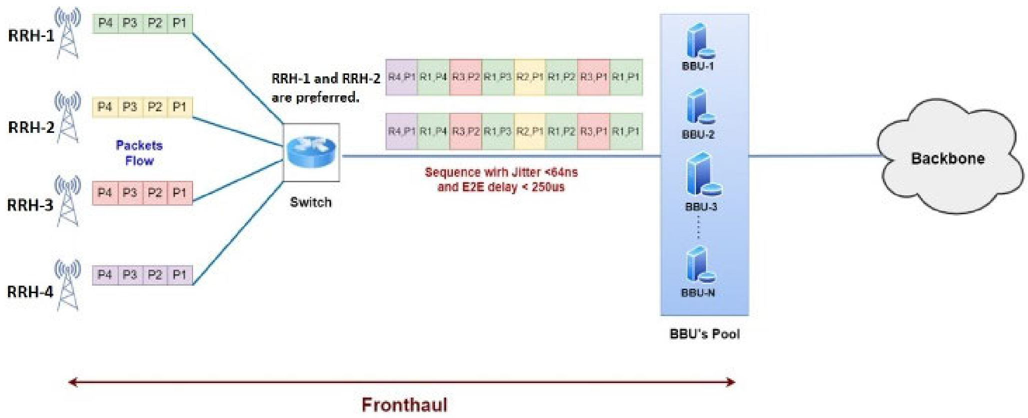

FIGURE 5. Packets flow when the total number RRHs are 4 and preferred number of RRHs are 2.

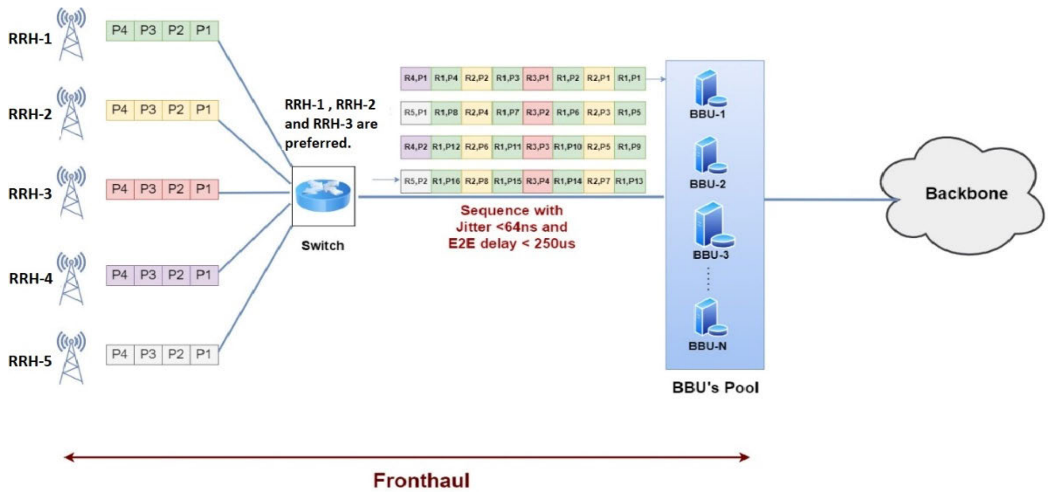

FIGURE 6. Packets flow when the total number RRHs are 5 and preferred number of RRHs are 3.

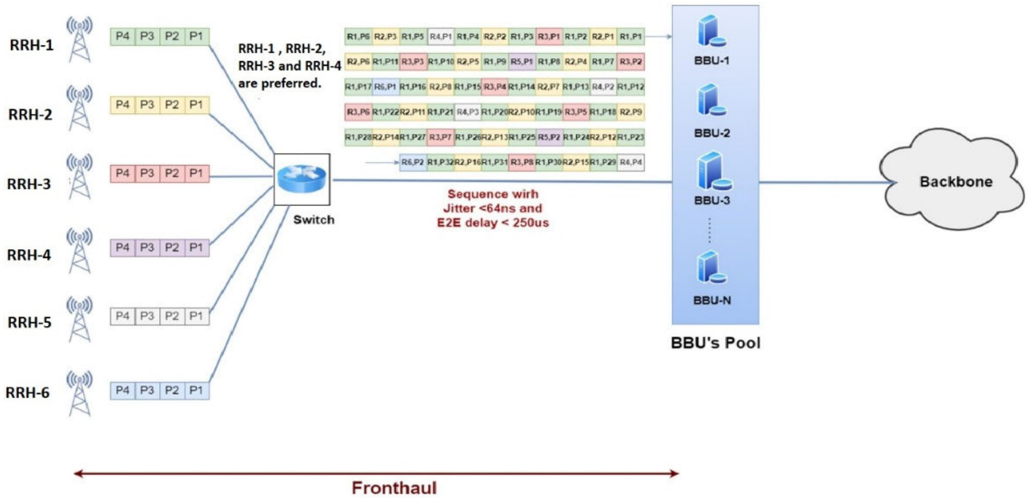

FIGURE 7. Packets flow when the total number RRHs are 6 and the preferred number of RRHs are 4.

the highest priority, while R3 and R4 share the lower priority, with preference represented as 4:2. The preference is decided by considering that R1 is transmitting packets by following a high CPRI line rate, followed by R2 and so on. Consequently, the sequence comprises the highest number of packets from R1 (8), followed by R2 with half of R1 (4), and R3 and R4 with half the number of R2 (2). The total number of packets in this sequence is 16. This sequence of four RRHs packets will

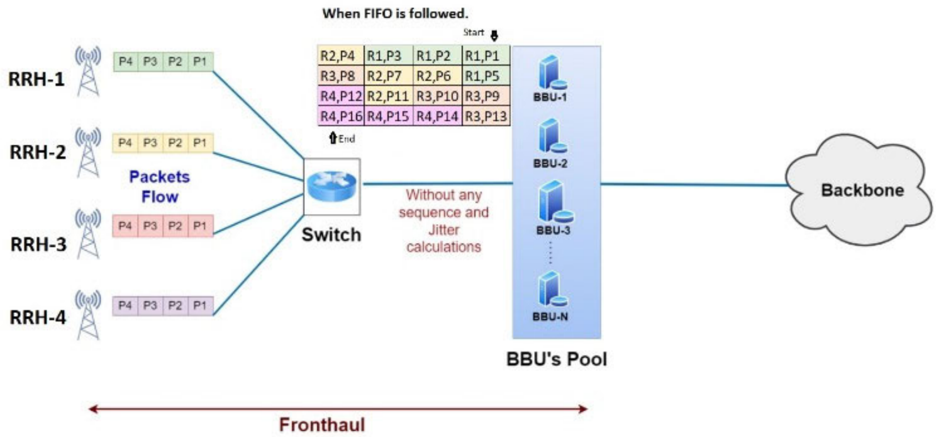

FIGURE 8. FIFO Output and high inter packet delay variations.

be repeated throughout the transmission, ensuring constant inter-packet delay resulting in jitter equal to zero. Scenario 1 is depicted in Figure 5, providing a clear representation of scheduling outcomes for various traffic flows. Four flows emanate from four RRHs, each allocated a designated fixed window size to accommodate the RRH flows. RRH-1 holds the highest preference, followed by RRH-3 as the second-highest priority, while RRH-2 and RRH-4 share the lowest priority. The primary goal is to maintain a consistent inter-packet delay for all traffic flows, ensuring tolerable jitter and preventing conflicting slots to minimize packet loss. At the ingress port of the switch, packets from different RRHs arrive at varying times, depending on the RRHs' CPRI line rates. The CCRA reorganizes and schedules these packets before transmitting them on the egress port of the switch. In the given scenario, eight packets from RRH-1, four packets from RRH-3, and two packets each from RRH-2 and RRH-4 are transmitted in each window. To mitigate congestion and packet loss, fixed timeslots are assigned to all packets. This window of conflict-free packets, with constant inter-packet delay variations, repeats throughout communication until there are changes in RRHs' traffic flows or network scenarios. Figures 6 and 7 illustrate scenarios for accommodating the increasing traffic from RRHs and adjusting the window size to ensure acceptable jitter and end-to-end delays. In Figure 6, the transmission of traffic from five RRHs to the BBU pool is depicted, with a preference ratio of 5:3 as shown in Table 2. The first three RRHs have prioritized order, while the last two share equal lowest priority. This sequence comprises a total of 32 packets, and zero jitter is maintained throughout the transmission in the case of five RRHs. In Figure 7, the transmission of traffic from six RRHs is demonstrated, and it is evident that the inter-packet delay remains constant among all packets, ensuring zero jitter in the flows. The proposed scheduling approach allows for scheduling packets for up to 10 RRHs, thereby increasing the multiplexing gains as demanded in fronthaul networks.

The FIFO benchmark scheme proved inadequate in meeting the delay and jitter requirements of CPRI traffic. This was due to its consistent processing and forwarding of packets to egress ports solely based on their arrival at the ingress ports [21]. The scheme resulted in the random forwarding of packets from the switch, lacking any mechanism to ensure consistent inter-packet delay variations for each CPRI flow. Consequently, this led to high jitter, rendering the scheme unsuitable for effective packet transportation. A comparison between FIFO and CCRA algorithms for scheduling time-sensitive fronthaul traffic is illustrated in Figures 8 and 9. Figure 8 depicts the ingress interface of the switch, which receives packets from four RRHs. Without a scheduling algorithm in place, the switch defaults to the FIFO scheme, forwarding packets based on their arrival times. This default mechanism results in high interpacket delay variations, reaching a jitter of 3 μs [22]. This value falls significantly short of meeting the fronthaul requirements of 64 ns. Figure 9 illustrates the switch's output after implementing the CCRA for the same input scenario. Regardless of the packet arrival rates, the proposed scheme consistently transmits them in a predefined order. This ensures a steady inter-packet delay, crucial for mitigating jitter and resolving conflicting time slot issues. A noticeable distinction is evident between the switch's behavior with FIFO, where packets are forwarded based on arrival times, resulting in elevated jitter, and CCRA, which maintains a uniform delay between packets. By utilizing the proposed CCRA algorithm, better scalability can be achieved. Despite increasing the number of RRHs producing high traffic load, the algorithm will always allocate the packets in a predefined and well-managed order to meet performance requirements. In terms of efficiency, FIFO is not always the most suitable for all types of data. For example, in a scenario where a high-priority packet arrives just after a large batch of lower-priority packets, the high-priority packet must wait for all preceding packets to be processed. This situation can potentially lead to intolerable E2E delays and variations in inter-packet delays. However, the CCRA prioritizes traffic flows based on their CPRI line rates. This ensures a constant inter-packet delay between all packets of a similar flow,

16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
R4	R1	R2	R1	R3	R1	R2	R1	R4	R1	R2	R1	R3	R1	R2	R1

FIGURE 9. **CCRA scheduling output and constant inter-packet delay variations.**

keeping the jitter value consistently at zero. In contrast to FIFO, which processes data packets in the order they arrive without considering their content, priority, or current network conditions, CCRA provides a more dynamic approach. FIFO operates on a first come first sending principle and is not intelligent enough to adapt to changes in network traffic patterns, link capacities, and latency needs [31]. Fronthaul networks demand dynamic and self-learning-based approaches to efficiently handle diverse and changing traffic patterns. The proposed CCRA scheduling algorithm is more suitable for fronthaul networks as it adjusts the traffic flow sequence based on changing network conditions due to the utilization of the Q-learning mechanism, ensures that the inter-packet delay and E2E delay for maximum number of RRHs traffic flows would be in the acceptable limits of fronthaul.

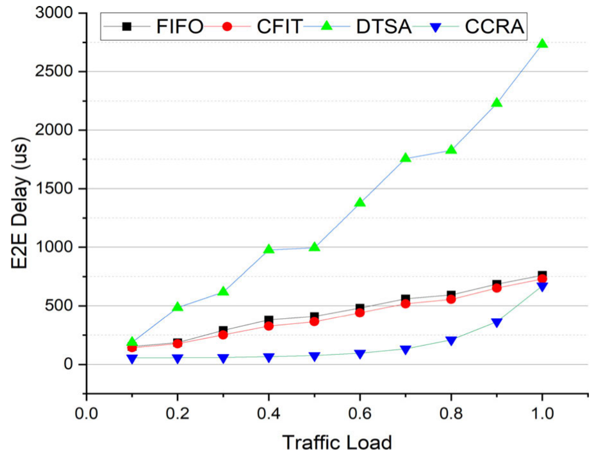

FIGURE 10. **E2E delay vs traffic load.**

IV. PERFORMANCE EVALUATION

This section provides an evaluation of the performance of the proposed CCRA scheme, focusing on E2E delays and jitter. These parameters are computed using Equations 1 to 9 and based on the studies [21] and [22]. The metrics are implemented using the Java programs. The study conducts a comparative analysis with existing scheduling schemes for fronthaul networks, including benchmark FIFO, CFIT, and DTSA. To assess performance, over 1000 packet flow sequences are generated, randomly selecting line rates based on Table 1. Up to ten flows per switch are multiplexed and transmitted over a 10 km Ethernet link at 10 Gbps, employing a packet size of 1500 bytes and a constant inter-packet delay of 1.2 μs. Table 2 is utilized to set the preferences and select the parameters such as priorities, number of RRHs, and traffic flows. The different networking schemes are compared based on the traffic load which is defined as a ratio between the sum of aggregated CPRI line rates, channel data rate, and estimated delay and jitter [21]. These considered matrices and parameter values are consistent with the current state-of-the-art schemes such as CFIT and DTSA for cross-comparison purposes and comply with fronthaul and Ethernet standards including 3GPP [7], IEEE P802.1CM [19], and IEEE P1914.3 [20].

In Figure 10, the proposed CCRA algorithm is compared with existing schemes including FIFO, DTSA, and CFIT algorithms in terms of E2E delays. The existing FIFO, DTSA, and CFIT algorithms exhibit a direct proportionality between traffic load and average E2E delay. As the traffic load increases, so does the average delay. This behavior arises because the CFIT algorithm performs effectively only when the traffic load is low. As the load increases, the number of conflicting slots also increases, leading to congestion and consequently higher E2E delays. Likewise, when the DTSA algorithm is employed, the E2E delay also increases with higher load values. The increased number of conflicting timeslots on a capacity-constraint Ethernet link causes congestion and results in elevated E2E delays. This relationship can be depicted by a curve illustrating an ascending trend in the E2E value as the traffic load rises. In contrast, when utilizing an RL-based CCRA scheme, the E2E delay increases with traffic load. However, due to traffic flow optimization, the E2E delay remains below the maximum limit of 250 μs in most of the cases. The RL-based CCRA scheme can learn to prioritize RRHs and packets that have a greater impact on the reward signal, such as RRHs with higher data rates and packets with stringent delay requirements are given more priority, by adjusting the timeslots. When the CCRA scheme is used the maximum E2E delay at traffic load 1.0 is 520 μs, while the E2E delay for CFIT, FIFO, and DTSA is 600 μs,700 μs, and 2700 μs respectively. Consequently, this approach enables more efficient resource utilization and reduced delays, even under high traffic loads.

Likewise, in Figure 11 when examining the average delay across all four algorithms including FIFO, CFIT, DTSA, and proposed CCRA, similar trends can be observed as in the case of E2E delays. The average delay graph also exhibited an increase with higher load values. This behavior arises because, as the load increases, the average queuing delay and other delay components also increase, resulting in an overall increase in the average delay. In the case of the DTSA algorithm, the average delay increases significantly, because DTSA is primarily designed for high-capacity links. However, when used with limited-capacity Ethernet links, its performance deteriorates. When the CCRA scheme is used the maximum average delay at traffic load 1.0 is 250 μs, while the E2E delay for CFIT, FIFO, and DTSA is 300 μs, 400 μs,

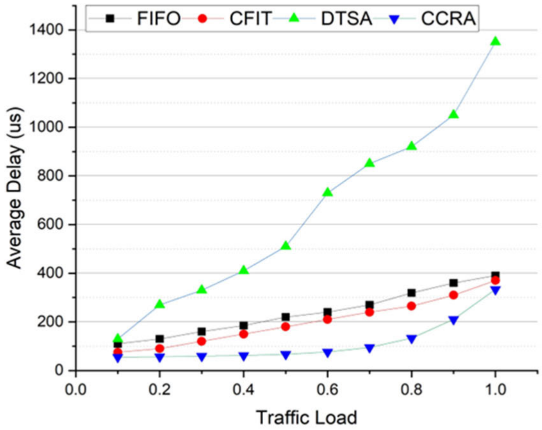

FIGURE 11. Average delay vs traffic load.

and 1300 μs respectively. Figure 11 demonstrates that the proposed CCRA scheme exhibits improved performance in terms of average delays when compared to existing schemes.

Figure 12 illustrates and compares the jitter of four algorithms including the FIFO, CFIT, DTSA, and proposed CCRA algorithm. Observing the figure, it is evident that when the load value is low, both CFIT and DTSA algorithms are capable of achieving low jitter. However, as the load increases, the jitter value also increases. This occurs due to the rise in conflicting slots, which leads to congestion and delays, consequently resulting in intolerable jitter. For CFIT algorithms, the jitter depends on the CPRI line rates. When the CPRI line rates are multiples of one another, the jitter is minimal or near zero. However, if the line rates are non-multiples of each other, the jitter value increases [22]. Likewise, the jitter value of the DTSA algorithm relies on the priority of flow packets. It operates effectively as long as there are available time slots in proximity. However, due to limited Ethernet bandwidth, when there are no free timeslots, high-priority flow packets are forced to wait, causing delays in packet flows and subsequently increasing the jitter value. In contrast, when employing the proposed CCRA algorithm, the jitter value is nearer to zero regardless of the load. This is achieved by allocating fixed time slots for all packets, resulting in a constant inter-packet delay variation. Consequently, tolerable jitter can be achieved for high load values. It is observed that the FIFO algorithm consistently exhibits high jitter values, which rapidly increase with higher load values. This is attributed to the absence of any scheduling technique, leading to highly conflicting slots, congestion, and delays. It can be seen in Figure 12 that when the CCRA scheme is used the jitter is within the threshold even at high traffic loads such as at 1.0, while the jitter for CFIT, FIFO, and DTSA has been increased.

In Figure 13, a comparison of the number of conflicting time slots has been made between the proposed CCRA scheme and the state-of-the-art scheduling schemes such as FIFO, CFIT, and DTSA algorithms. Fronthaul networks have a finite bandwidth to transmit data. When multiple packet requests attempt to use the network simultaneously, there

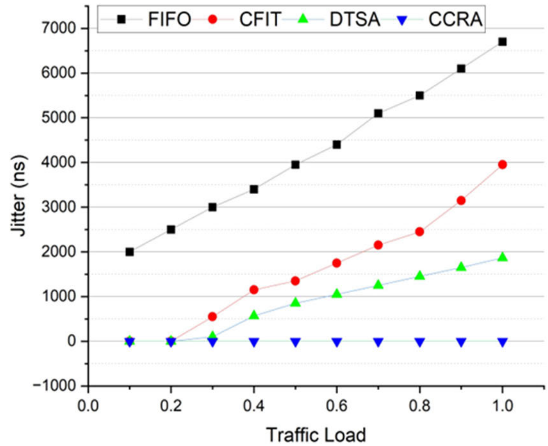

FIGURE 12. Jitter vs traffic load.

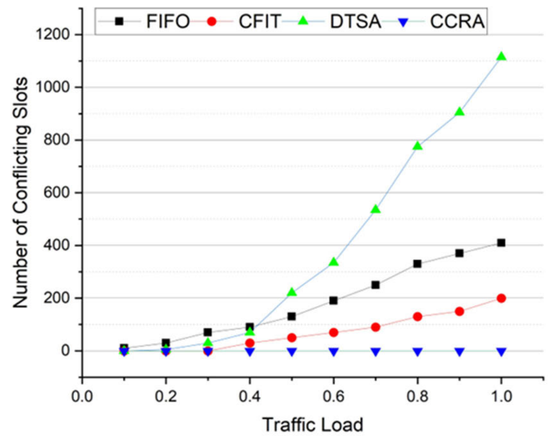

FIGURE 13. Number of conflicting slots vs traffic load.

may not be enough available bandwidth to accommodate all the requests, leading to conflicting time slots. This study utilizes capacity-constraints Ethernet links, when the network experiences congestion from a high volume of data traffic, resource contention occurs, leading to conflicting time slots. As depicted in Figure 13, when employing the FIFO, CFIT, and DTSA algorithms, the number of conflicting time slots increases with higher load values. In CFIT, since there is less effective scheduling for the high number of packets, the occurrence of conflicting time slots depends on the line rate values. If the line rates are multiples of each other, no conflicting time slots arise. However, when the line rates are non-multiples of each other, conflicting time slots occur, resulting in high congestion. In the DTSA algorithm, conflicts arise when different priority flows or the same priority flows seek available time slots across multiple channels. On the other hand, when the proposed CCRA scheme is employed, a fixed time slot adjustment is applied to all packets, ensuring that no conflicting time slots occur even with increased load. This fixed scheduling mechanism prevents conflicts and contributes to the enhanced utilization of network resources. On the other hand, when the other state-of-the-art scheduling algorithms are implemented a large

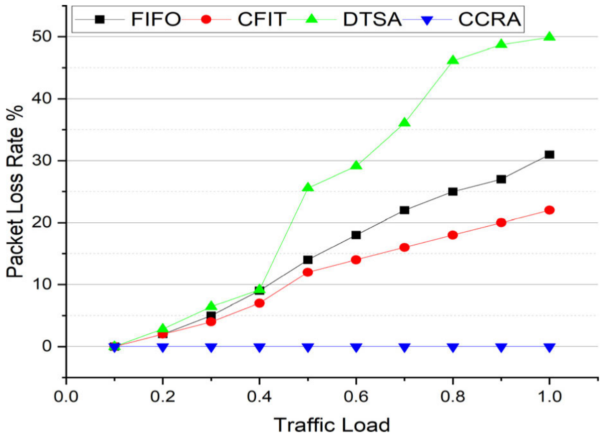

FIGURE 14. **Packet loss rate vs traffic load.**

number of conflicting slots occur which results in increased delay and jitter.

Figure 14 depicts a comparative analysis of the CCRA algorithm with respect to packet loss ratio (PLR), comparing it with algorithms, namely FIFO, DTSA, and CFIT. Packet loss occurs when data packets transmitted in a network fail to reach their intended destination. The primary cause of packet loss is network congestion. When multiple packet requests concurrently seek network access, insufficient available bandwidth may hinder the accommodation of all requests, leading to conflicting time slots and subsequent network congestion. The PLR can be computed using Equation 10.

$$Packet\ Loss\ Rate\ (PLR) = \frac{LP}{TP} \times 100 \tag{10}$$

where LP is the number of packets lost while TP is the total number of packets in the flow. From Figure 14, it becomes evident that state-of-the-art scheduling algorithms like FIFO, CFIT, and DTSA exhibit packet loss ratios as the traffic load increases, indicating a significant loss of data packets during transmission due to limited bandwidth consideration in this study. This loss can have detrimental effects on the quality and reliability of network services. In contrast, the proposed CCRA scheme yields nearly zero PLR. This favorable outcome is attributed to the absence of conflicting time slots when CCRA is applied, resulting in a congestion-efficient fronthaul network and, consequently, zero packet losses. The proposed CCRA scheme shows improved performance in terms of packet loss and congestion control as compared to the other state-of-the-art scheduling schemes including FIFO, DTSA, and CFIT.

The DTSA algorithm could be suitable in a scenario of high-capacity links which are very expensive in deploying Ethernet-based fronthaul networks and eliminate the most important benefit of fronthaul which is the cost-efficiency. Another problem is that this algorithm works on prioritizing the flows that may lead to starvation of LPF traffic when there is a continuous presence of HPF traffic. The prioritization of HPF traffic in the algorithm results in the consumption of a substantial portion of network resources, leaving little to no available bandwidth for LPF traffic. Consequently, this prioritization leads to higher queuing delays for LPF traffic, causing an increase in end-to-end delay and jitter. Therefore, it can efficiently manage and route traffic, reducing congestion and latency when there is less traffic load, and can optimize the allocation of network resources during less traffic load. This algorithm is suitable for networks having high bandwidth and unlike Ethernet-based fronthaul networks where factors such as limited bandwidth and cost are not a challenge. On the other hand, the CFIT algorithm gives tolerable jitter values when the CPRI line rates are perfect multiple of each other, if the CPRI line rates are not multiple of each other it does not handle the traffic load efficiently and results in intolerable jitter and packet loss. This is suitable for network scenarios where transmission of traffic is limited to a few numbers of RRHs or limited traffic such as less than 0.35 LER. FIFO is easy to implement for networks and treats all data packets equally without any prioritization. Due to its simplicity, FIFO has a minimum computational cost. FIFO has a lack of optimization as it does not account for different priorities or types of traffic, which can be a major drawback in a fronthaul network where time-sensitive traffic is prevalent and necessary to prioritize different RRHs traffic flows depending upon the line rates to efficiently allocate resources. Moreover, during high traffic periods, FIFO can lead to increased inter-packet delays which causes intolerable delays in fronthaul networks. In the proposed CCRA scheme, jitter and latencies are computed using the reinforcement learning algorithm for different traffic flows and then packets are scheduled in such a way they get the fixed time slots, and no conflicting slots are assigned resulting in fixed inter-packet delays which require to achieve tolerable jitter in the fronthaul networks. The fixed-sized and no conflicting slot mechanism eliminates the need for retransmissions and chances of packet loss. The CCRA is suitable for network scenarios where time-sensitive traffic from multiple RRHs have to transmit in low bandwidth links and retaining the jitter and end-to-end delays within the tolerable limits of fronthaul networks is mandatory. The proposed scheme is designed in such a way that it first takes the requirements of the network and depending upon the available resources it dynamically allocates the time slots and transmission times to each packet of different flows which makes it scalable and highly useful for scenarios like fronthaul.

In the proposed CCRA algorithm, efficient scheduling is performed by the traffic flows in a proper sequence by allocating them a fixed time slot based on their CPRI line rates and preference as there are reinforcement learning techniques are used so it learns itself and follows the best traffic flow sequence according to the intensity of the traffic load. The proposed CCRA algorithm is designed to be scalable and capable of handling traffic loads of several RRHs without degrading the performance requirements of fronthaul. The CFIT algorithm can efficiently handle traffic at low load conditions. The DTSA could handle a large number of RRHs if high-capacity links are deployed but in case of limited bandwidth such as in the case of Ethernet, DTSA performance is limited. When the traffic load surpasses a specific threshold, as indicated in the graphs illustrating E2E delay and jitter, it becomes evident that the CFIT and DTSA algorithms experience an increase in both jitter and E2E delays. This escalation renders the existing scheduling schemes less suitable for low-capacity Ethernet-based networks and more suitable for high-capacity links like optical fiber. The CFIT algorithm performs well when the CPRI line rates are perfect multiple of each other, at high load, it assigns conflicting time slots to traffic flows which causes congestion and increases the jitter and E2E delays. In DTSA, the existence of high-priority traffic flows leads to extended waiting times for low-priority flows. This imbalance in inter-packet delays contributes to heightened jitter and delays. The experimental results show that when the proposed algorithm is used then the overall performance of the network improves. This scheme could be compatible with other networking topologies required to transmit traffic by ensuring the QoS, especially in Ethernet-based fronthaul networks. By increasing the bandwidth, the proposed scheme can be improved to multiplex and transmit traffic of more RRHs over the Ethernet-based fronthaul links.

V. CONCLUSION

In this study, a novel congestion control and resource allocation (CCRA) scheme based on Q-learning has been proposed where E2E delay, jitter, and packet loss rate are computed under different traffic load conditions for the Ethernet-based fronthaul networks. The scheme aims to improve multiplexing gains by successfully transmitting the traffic of multiple RRHs on capacity-constraint Ethernet links of 10 Gbps. The scheme emphasizes minimizing inter-packet delays, E2E delays, and packet loss ratio to minimize the need for packet retransmissions and wastage of bandwidth. These are achieved by performing the efficient scheduling of time-sensitive packet flows by employing the Q-learning and dynamically slots allocation procedure which enhances the adaptability and self-learning features in the scheme. As a result, conflict-free and predefined slots are assigned which improves the performance of the network despite high traffic load conditions. The simulation results verify that the proposed scheme outperforms the existing scheduling schemes proposed for the fronthaul networks. In conclusion, the proposed Q-learning, congestion control, and resource allocation scheme have the potential to significantly improve the performance of Ethernet-based 5G fronthaul networks. By employing it at intermediate switches without upgrading the hardware resources, traffic between multiple RRHs and the BBU pool can be transmitted while satisfying the QoS requirements.

To assess the effectiveness of proposed congestion control and resource allocation mechanisms, future research would focus on experimental validation through field trials. Conducting real-world experiments in fronthaul network environments can provide valuable insights into the practical implications, limitations, and performance of proposed solutions. These experiments would also assist in identifying further challenges and potential improvements in the scheme.

REFERENCES

[1] C.-H. Fang, L.-H. Shen, T.-P. Huang, and K.-T. Feng, "Delay-aware admission control and beam allocation for 5G functional split enhanced millimeter wave wireless fronthaul networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2430–2444, Apr. 2022.

[2] Y. Azimi, S. Yousefi, H. Kalbkhani, and T. Kunz, "Applications of machine learning in resource management for RAN-slicing in 5G and beyond networks: A survey," *IEEE Access*, vol. 10, pp. 106581–106612, 2022.

[3] E. C. Awasume, S. Musyoki, and V. K. Oduol, "Cloud radio access network fronthaul solution using optimized dynamic bandwidth allocation algorithm," *Int. J. Electr. Comput. Eng.*, vol. 11, no. 2, pp. 1395–1404, Apr. 2021.

[4] Y. Xu, G. Gui, H. Gacanin, and F. Adachi, "A survey on resource allocation for 5G heterogeneous networks: Current research, future trends, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 668–695, 2nd Quart., 2021.

[5] M. F. Hossain, A. U. Mahin, T. Debnath, F. B. Mosharrof, and K. Z. Islam, "Recent research in cloud radio access network (C-RAN) for 5G cellular systems—A survey," *J. Netw. Comput. Appl.*, vol. 139, pp. 31–48, Aug. 2019.

[6] D. R. Du, J. Li, Z. Y. Ding, L. Q. Wu, L. Li, and S. J. Huang, "A novel ICIC scheme combining 3D ML-SFR and CoMP," *Radioengineering*, vol. 31, no. 1, pp. 85–93, Apr. 2022.

[7] *Study on New Radio Access Technology-Radio Access Architecture and Interfaces*, document TR 38.801, Version 14.0.0, 3GPP, 2017.

[8] X. Lin, "An overview of 5G advanced evolution in 3GPP release 18," *IEEE Commun. Standards Mag.*, vol. 6, no. 3, pp. 77–83, Sep. 2022.

[9] A. Lometti and V. Sestito, "Fronthaul in 5G transport networks: IEEE1914.1 architecture and requirements," in *Proc. 22nd Int. Conf. Transparent Opt. Netw. (ICTON)*, Bari, Italy, Jul. 2020, pp. 1–4.

[10] A. Fayad and T. Cinkler, "Cost-effective delay-constrained optical fronthaul design for 5G and beyond," *Infocommunications J.*, vol. 14, no. 2, pp. 19–27, 2022.

[11] Y. Huang and A. Ikhlef, "Joint optimization of wireless fronthaul and access links in CRAN with a massive MIMO central unit," in *Proc. IEEE Int. Conf. Commun.*, Seoul, South Korea, May 2022, pp. 1906–1911.

[12] M. K. Al-Hares, P. Assimakopoulos, and N. J. Gomes, "Ethernet fronthaul transport of an upper-physical layer functional split using time-aware shaping," *Opt. Switching Netw.*, vol. 41, Sep. 2021, Art. no. 100605.

[13] I. Freire, I. Almeida, E. Medeiros, M. Berg, C. Lu, E. Trojer, and A. Klautau, "Testbed evaluation of distributed radio timing alignment over Ethernet fronthaul networks," *IEEE Access*, vol. 8, pp. 87960–87977, 2020.

[14] M. Waqar and A. Kim, "Performance improvement of Ethernet-based fronthaul bridged networks in 5G cloud radio access networks," *Appl. Sci.*, vol. 9, no. 14, p. 2823, Jul. 2019.

[15] J. Kim, S. H. Chang, and J. K. Lee, "Comparative study for evaluating the cost efficiency of 5G Ethernet mobile fronthaul networks," *J. Opt. Commun. Netw.*, vol. 14, no. 12, pp. 960–969, Dec. 2022.

[16] O. A. Nwogu, G. Diaz, and M. Abdennebi, "Differential traffic QoS scheduling for 5G/6G fronthaul networks," in *Proc. 31st Int. Telecommun. Netw. Appl. Conf. (ITNAC)*, Nov. 2021, pp. 113–120.

[17] S. Bhattacharjee, R. Schmidt, K. Katsalis, C.-Y. Chang, T. Bauschert, and N. Nikaein, "Time-sensitive networking for 5G fronthaul networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Dublin, Ireland, Jun. 2020, pp. 1–7.

[18] Y. Liu, Y. Zhou, J. Yuan, and L. Liu, "Delay aware flow scheduling for time sensitive fronthaul networks in centralized radio access network," *IEEE Trans. Commun.*, vol. 68, no. 5, pp. 2992–3009, May 2020.

[19] *IEEE Draft Standard for Radio Over Ethernet Encapsulations and Mappings*, IEEE Standard P1914.3, 2017.

[20] *Standard for Local and Metropolitan Area Networks-Time-Sensitive Networking for Fronthaul*, IEEE Standard P802.1CM, 2017.

[21] D. Chitimalla, K. Kondepu, L. Valcarenghi, M. Tornatore, and B. Mukherjee, "5G fronthaul-latency and jitter studies of CPRI over Ethernet," *J. Opt. Commun. Netw.*, vol. 9, no. 2, pp. 172–182, Feb. 2017.

[22] L. Wu, C. Gan, Z. Xu, and J. Hui, "Zero-jitter differentiated traffic scheduling algorithm in 5G fronthaul hybrid network," *Optik*, vol. 265, Sep. 2022, Art. no. 169558.

[23] A. Mohammed Mikaeil, W. Hu, and L. Li, ''Joint allocation of radio and fronthaul resources in multi-wavelength-enabled C-RAN based on reinforcement learning,'' *J. Lightw. Technol.*, vol. 37, no. 23, pp. 5780–5789, Sep. 3, 2019.

[24] W.-C. Chien, C.-F. Lai, and H.-C. Chao, ''Dynamic resource prediction and allocation in C-RAN with edge artificial intelligence,'' *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 4306–4314, Jul. 2019.

[25] D.-Y. Kim and S. Kim, ''Incoming traffic control of fronthaul in 5G mobile network for massive multimedia services,'' *Multimedia Tools Appl.*, vol. 80, nos. 26–27, pp. 34443–34458, Nov. 2021.

[26] F. W. Murti, S. Ali, and M. Latva-Aho, ''Deep reinforcement based optimization of function splitting in virtualized radio access networks,'' in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Montreal, QC, Canada, Jun. 2021, pp. 1–6.

[27] H. Shaheen, M. S. Bhuvaneswari, N. Balaganesh, B. K. Rani, P. J. Paul, S. Deepajothi, and A. A. Berhanu, ''Utility-based joint power control and resource allocation algorithm for heterogeneous cloud radio access network (H-CRAN),'' *Wireless Commun. Mobile Comput.*, vol. 2022, pp. 1–7, Aug. 2022.

[28] J. P. Martins, I. Almeida, R. Souza, and S. Lins, ''Policy distillation for real-time inference in fronthaul congestion control,'' *IEEE Access*, vol. 9, pp. 154471–154483, 2021.

[29] A. Younis, T. X. Tran, and D. Pompili, ''Energy-efficient resource allocation in C-RANs with capacity-limited fronthaul,'' *IEEE Trans. Mobile Comput.*, vol. 20, no. 2, pp. 473–487, Feb. 2021.

[30] Z. Gao, M. Eisen, and A. Ribeiro, ''Resource allocation via graph neural networks in free space optical fronthaul networks,'' in *Proc. IEEE Global Commun. Conf.*, Taipei, Taiwan, Dec. 2020, pp. 1–6.

[31] M. Waqar, A. Kim, and P. K. Cho, ''A transport scheme for reducing delays and jitter in Ethernet-based 5G fronthaul networks,'' *IEEE Access*, vol. 6, pp. 46110–46121, 2018.

[32] L. Wang, X. Que, X. Gong, Y. Tian, T. Wang, and X. Wang, ''A scheduling algorithm for low jitter in Ethernet-based fronthaul,'' in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Barcelona, Spain, Jun. 2019, pp. 1–6.

[33] F. Tonini, C. Raffaelli, S. Bjørnstad, D. Chen, and R. Veisllari, ''A traffic pattern adaptive mechanism to bound packet delay and delay variation in 5G fronthaul,'' in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Valencia, Spain, Jun. 2019, pp. 416–420.

[34] S.-H. Park, O. Simeone, and S. Shamai, ''Robust baseband compression against congestion in packet-based fronthaul networks using multiple description coding,'' *Entropy*, vol. 21, no. 4, p. 433, Apr. 2019.

[35] I. Nascimento, R. Souza, S. Lins, A. Silva, and A. Klautau, ''Deep reinforcement learning applied to congestion control in fronthaul networks,'' in *Proc. IEEE Latin-American Conf. Commun. (LATINCOM)*, Nov. 2019, pp. 1–6.

[36] D. Yu, Y. Liu, Z. Li, and H. Zhang, ''Energy-efficient beamforming design for user-centric networks with full-duplex wireless fronthaul,'' *IEEE Trans. Commun.*, vol. 71, no. 3, pp. 1521–1535, Mar. 2023.

[37] T. Leyrer, P. Varis, W. Wallace, P. Gangadar, M. Mandhana, P. Jayarajan, and S. Karaiyan, ''Analysis and implementation of multi-protocol gigabit Ethernet switch for real-time control systems,'' in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Montreal, QC, Canada, Jun. 2021, pp. 1–6.

MUHAMMAD WAQAR received the B.S. degree in computer engineering and the M.S. degree in computer networks from COMSATS University Islamabad (CUI), Pakistan, in 2011 and 2015, respectively, and the Ph.D. degree in 5G cloud radio access networks from Sejong University, Seoul, South Korea, in 2020. He was an Assistant Professor with CUI. He is currently with the University of Suffolk (UoS), Ipswich, U.K. He has published a number of research papers in peer-reviewed journals and international conferences. His research interests include 5G cloud networks, cybersecurity, and deep learning. He serves as a member of steering and technical program committees for numerous international conferences. He is a reviewer of several esteemed journals and a member of editorial boards.

MUHAMMAD USMAN MUSTAFA received the bachelor's and M.S. degrees in software engineering from COMSATS University Islamabad, Islamabad, Pakistan, in 2021. His research interests include computer networks, next-generation technologies, and QoS in ethernet.

FARHANA JABEEN received the Ph.D. degree from The University of Manchester, U.K., in 2011. She is currently an Associate Professor (Tenured) with COMSATS University Islamabad, Islamabad, Pakistan. Her research interests include different aspects of computer vision, cyber security, and distributed networks, particularly the development of solutions (architectures, algorithms, and protocols) for emerging paradigms specifically named data networking, smart grids, software-defined networking, the Internet of Things (IoT), VANETs, edge cloud computing, and 5th generation communication. Her work has appeared in several journals and conferences. Moreover, she serves as a regular reviewer for numerous journals and various international conferences. She is also working as the program chair and an associate editor.

SYED AZIZ SHAH is currently an Associate Professor with the Research Centre for Intelligent Healthcare (CIH), Coventry University. Since 2020, he has been leading the Healthcare Sensing Technology Research Group, Coventry University, overseeing two Assistant Professors, two Research Fellows, and eight Ph.D. students, and having obtained extensive administrative experience. With academic qualifications spanning across the U.K., China, Sweden, and Pakistan, he brings over 12 years of experience in teaching, research, and innovation. He is an interdisciplinary researcher focusing on advanced radio frequency (RF) sensor design and signal processing using RF and THz sensing, specifically for physiological measurements. He is highly motivated to utilize and develop advanced technologies that address the unmet healthcare challenges of unobtrusive monitoring for health and well-being purposes. With a proven track record of securing substantial research funding; notably, he secured research grants totaling £1 million (eight grants, six as PI, and two as Co-I) and was honored with the U.K.'s prestigious Engineering and Physical Sciences Research Council (EPSRC) New Investigator Award for the project ''Radar Sensing for Human Activity Monitoring of Daily Living Simultaneously in Multiple Subjects'' (EP/W037076/1, £410,000 fEC). He has also (co)authored more than 110 top-rank peer-reviewed journals and conferences, including three transactions, indicating his dedication to advancing knowledge in the field of RF sensing. He provides strategic leadership, oversees postgraduate research activities, and offers mentorship to M.Sc. and Ph.D. students. He is actively involved in supporting PGR collaborations, maintaining ethical standards, managing PGR funds, and enhancing research impact within the research center and faculty. His research interests include RF sensing, telecommunications, 5G and beyond networks, and the use of technology in the healthcare industry.