Hindawi
Wireless Communications and Mobile Computing
Volume 2020, Article ID 7065202, 15 pages
https://doi.org/10.1155/2020/7065202

Research Article

An Intelligent SDN Framework Based on QoE Predictions for Load Balancing in C-RAN

Gleison O. Medeiros,[1,2] **João C. W. A. Costa**,[1] **Diego L. Cardoso**,[1] **and Adam D. F. Santos**[2]

[1]*Institute of Technology, Federal University of Pará (UFPA), Belém, PA, Brazil*
[2]*Institute Geosciences and Engineering, Federal University of Southern and Southeastern Pará (UNIFESSPA), Marabá, PA, Brazil*

Correspondence should be addressed to Gleison O. Medeiros; gleison@unifesspa.edu.br

Received 23 November 2019; Revised 19 February 2020; Accepted 25 May 2020; Published 18 June 2020

Academic Editor: Nathalie Mitton

The rapid growth of the Internet and technological advances are forcing mobile operators to increasingly invest in network infrastructures. C-RAN and SDN are regarded as enabling technologies that can overcome the limitations faced by operators, by reducing costs, increasing scalability, and paving the way for the next generation of 5G cellular networks. In this paper, an architectural solution based on SDN and computational intelligence is proposed for C-RAN, which can adjust BBU-RRH mapping through network load balancing rules by predicting subjective and objective QoE metrics for UHD video streaming. The simulation results achieved gains between 59% and 129%, in scenarios without activating a new BBU and scenarios that involve activating a new BBU, respectively.

1. Introduction

The world is currently witnessing an exponential evolution in wireless mobile communications. The inclusion of digital modulation techniques, the advance of wideband code division multiple access (WCDMA), orthogonal frequency-division multiplexing (OFDMA), and multiple-input and multiple-output (MIMO), have all made a significant contribution to the improvement of cellular communications. However, with the emergence of the next generation of mobile networks (5G), the rapid and constant proliferation of devices and the increase in demand for multimedia applications, such as ultrahigh definition (UHD) video and online games, are driving the mobile network operators (MNOs) to invest notably more in the infrastructure. According to [1], the amount of data traffic expected by 2022 will be up to seven times higher than the traffic that was recorded in 2017, reaching 77.0 exabytes per month. In the same study, it was found that the traffic generated by video applications is the main source for this volume of traffic. In this context, MNOs have been instructed to maximize their capital expenditure (CAPEX) and operating expenses (OPEX), to provide network services efficiently, along with a sufficiently high standard to meet the quality expectations of end-user experience (QoE), while the average incomes of users will not be sufficient to cover the rise in expenses [2].

This evolution offers new business models in which the key factor and the source of revenue are not the networks or even the content itself, but the degree of satisfaction experienced by the customers who are paying for the service [3]. Moreover, those who are driving MNOs are looking for new mechanisms to assess the degree of user satisfaction through QoE metrics, which involves many subjective factors unrelated to quality of service (QoS) evaluation metrics, which are largely based on network performance. However, they also involve assessing the mood of the user or discovering how to represent system responsiveness, and this leads to a special kind of service level agreements (SLA) which are designed to establish a common standard for the level of quality that the customer will experience from using the services and are also called experience level agreements (ELA) [4].

The emergence of software-defined networks (SDN) and the cloud radio access network (C-RAN) offers promising technological solutions to confront this challenge. These technologies allow the implementation of efficient network resources and flexible scheduling to be shared. The SDN also separates the control plane from the data (or forwarding) plane. The advantage of this separation is that it allows an SDN controller to acquire an updated global view not only of the whole network but also of all the flows competing for traffic, which can increase the flexibility and scalability of the system, by making networks programmable, adaptable, and cost effective [5, 6]. These characteristics allow, for example, the integration of innovative solutions based on artificial intelligence (IA) techniques. C-RAN has emerged as a cellular network architecture that is capable of (a) meeting the demand for high end-user traffic data, (b) optimizing the use of physical resources, and (c) reducing costs [7]. As noted in Figure 1, C-RAN can be regarded as an evolution of the distributed cellular network where the computational resources of the base stations (BSs) are centralized in a pool consisting of baseband units (BBUs) implemented with the features of network function virtualization (NFV) and a remote radio head (RRH) with radio frequency (RF) features, both connected by low-latency high-bandwidth links, called fronthaul [8].

According to [9], the adoption of this type of approach implies that less rooms and equipment are needed to cover the same areas, while it also reduces the energy consumption of air conditioning and other support equipment. Besides, the connections between BBUs and RRHs are changed dynamically, allowing a BBU to connect to one or more RRHs. This facilitates the balanced allocation of computational resources between different BBUs [8]. More details on the C-RAN architecture can be seen at [5].

However, even with the implementation of C-RAN, there will still be some recurring problems in managing network resources. The number of active users in mobile networks varies considerably, depending on the time of day. During the daytime, for example, the BSs in the commercial areas of big urban centers are the most widely used, while at nighttime the most frequently visited BSs are in residential areas or leisure centers. Currently, the processing capabilities of each BS are only used by active users on the coverage, which causes the problem of idle BSs in some areas and BS overloading in others. This nonuniform and dynamic traffic load is called the "tidal effect" flow [10]. Thus, when there is an increase in the network requirements at a given location, more RRHs are needed to improve coverage. If the BBU cannot support the network requirements, it is necessary to intensify the processing or add more BBUs to the BBU pool [11]. The increase in baseband (BBU) processing, caused by high-intensity applications of data, requires a high modulation and coding scheme index (MCS). However, random increases in MCS affect the BBUs' computational load, resulting in overload and loss of data, factors that may have a direct influence on the user's experience [12].

Given this, in the future, the network architecture will inevitably be faced with a situation in which the various RRHs are connected to BBU pools. This will cause an increasingly serious problem about how to allocate resources and lead to a lowering of performance standards in the services provided to users and underutilization and waste of computing resources by MNOs. For this reason, a good deal of attention should be paid for providing solutions for proactive mapping techniques for load balancing between BBU-RRHs that involve heterogeneous multimedia services and user dynamics.

Although these concepts have been proposed and approach through the application of the genetic algorithm (GA) and particle swarm optimization (PSO) in [2, 13], respectively, minor research has been carried out that addresses the management of services that require a high volume of data. There has been also a failure to examine the effect of this on the QoE that these services must satisfy and highlight the need for innovative solutions that can aid the MNOs to comply with the SLA and provide users with appropriate QoE. In that regard, this paper proposes an intelligent BBU-RRH mapping architecture for C-RAN, which is capable of providing network load balancing in response to user demand and transforming a network-centric resource allocation for user-centric resource allocation. To this end, we have designed an SDN controller based on AI capable of predicting the QoE for video streaming services, to (a) adjust the BBU-RRH ratio to user demand, (b) reduce of blocked call events, and (c) reduce operational costs.

The remainder of this paper is structured as follows: Section 2 examines the recent solutions with regard to load balancing and mapping (BBU-RRH) issues involving the C-RAN architecture. In Section 3, our framework is established and there is a discussion of its relationship to the traditional components of a C-RAN architecture. In Section 4, we analyze and discuss the results obtained from the simulations. Finally, in Section 5, there are some final considerations and we make some recommendations for future work in the field.

2. Related Work

The dynamic allocation problem of BBU resources for RRH and C-RAN load balancing has been investigated extensively in many works from combining techniques to employing algorithms as AI-based optimization tools. In [9], the authors propose semistatic and adaptive switching schemes to determine the best combinations between BBUs-RRHs and ensure a continuous traffic load. The results confirmed that the number of BBUs can be reduced by 26% and 47% for semistatic and adaptive schemes, respectively, compared with conventional cell deployment. In [13], a GA and discrete particle swarm optimization (DPSO) are proposed as evolutionary algorithms to solve the problem of BBU-RRH mapping in dynamic traffic scenarios. Computational results based on three benchmark problems demonstrate that GA and DPSO deliver optimum performance for small networks, whereas close optimum is delivered for large networks. The results of both GA and DPSO are compared to exhaustive search (ES) and K-means clustering algorithms. The percentage of blocked users in a medium-sized network scenario is reduced from 10.523% to 0.421% and 0.409% by GA and

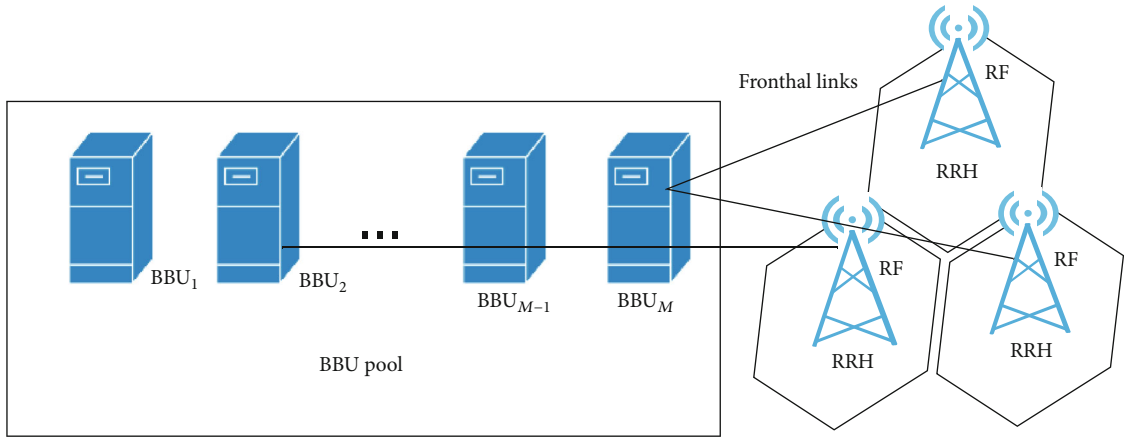


Figure 1: Traditional C-RAN architecture.

DPSO, respectively. In [14], an algorithm is proposed that considers the signal interference-noise relationship (SINR) of user equipment (UE) as a determining factor to improve BBU-RRH mapping. The results show that the proposed algorithm was able to determine the ideal number of RRHs per BBUs in dynamic traffic loads. In [15], the authors propose a joint solution to the BBU-RRH mapping and user association problem in C-RAN to minimize the system cost incurred by the power consumption of all RRHs and rentals of BBUs. Simulation results have demonstrated that the proposed algorithm performs very close to the optimal solution and accomplishes even better when the QoS requirement is not strict. In [16], the authors propose an optimization problem that models the optimal allocation of computational resources between BBUs and RRHs. Results show that the computational resource requirements and the power consumption of BBUs and the physical machines decrease as the channel quality worsens. Moreover, the developed heuristic solution can be close to the optimal performance while having lower complexity. In [17], the authors proposed a dynamic switching scheme through the resource renting approach. The purpose of this is to improve load balancing in scenarios where the problem of BBU resource scarcity is imminent. Simulation results demonstrate that the proposed mechanism in C-RAN significantly reduces the waste of resource usage and improves the throughput. In [2], the authors deploy genetic algorithms (GA) as a promising load balancing and BBU-RRH mapping solution to obtain a minimum number of blocked calls and to maximize QoS. The aim is to find the solution that is most likely to fit the adaptation to the scenario. Simulation results show a reduction of 100% in the number of blocked calls. A similar strategy is adopted in [18], although the authors address the problem by employing the particle swarm optimization (PSO), where the best solution is randomly determined on the basis of the speed and position of the particles with the best values for testing aptitude. The results obtained show improvements of up to 100% on blocked calls when QoS factors are taken into account. In [19, 20], the authors provide an enhanced dual-load balancing mechanism for connectivity through bandwidth adaptation. This system involves grouping bearers that are transmitted by the RRHs to proportionally divide network resources. Finally, in [11], the authors proposed a new C-RAN network architecture that has low transmission latency and low power consumption. The purpose of this is to configure BBU-RRH resources in situations of sudden change in traffic by combining the throughput prediction with long short-term memory (LSTM) and the optimization power of GA.

Although BBU-RRH resource allocation problems have been addressed, none of the research studies mentioned above have effectively dealt with the problem of mapping and load balancing in factors related to QoE that influence or have a bearing on the question of user/customer satisfaction. Hence, it can be claimed that this paper makes two key research contributions:

(i) We establish a C-RAN architectural framework based on SDN that can be easily adapted to and administered by MNOs

(ii) Within a dynamic network environment, a load balancing and mapping algorithm (BBU-RRH) is proposed, which is based on QoE predictions that make use of Artificial Neural Networks (ANN)

3. SDN Framework for C-RAN Architecture

This section describes an architectural scheme for C-RAN that is based on the integration of C-RAN and SDN components. This feature allows the architecture in managing its resources in accordance with user demand in different traffic situations. The modular design enables MNOs to include, alter, or exclude policies, technologies, and services.

The framework of the architecture consists of the following components:

(i) An Intelligent SDN Framework that is capable of the following:

(a) Predicting the mean opinion score (MOS) of users concerning UHD videos that are transmitted within a C-RAN architecture

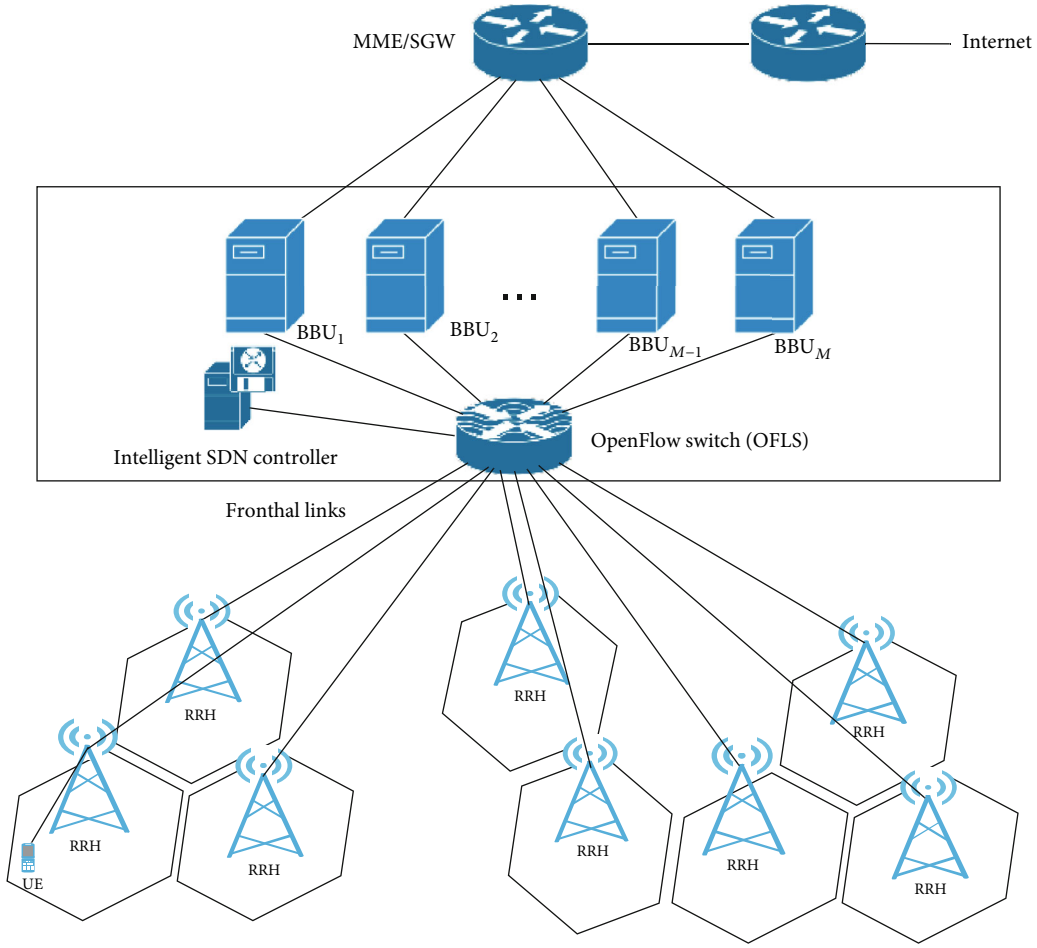


Figure 2: The C-RAN SDN architecture.

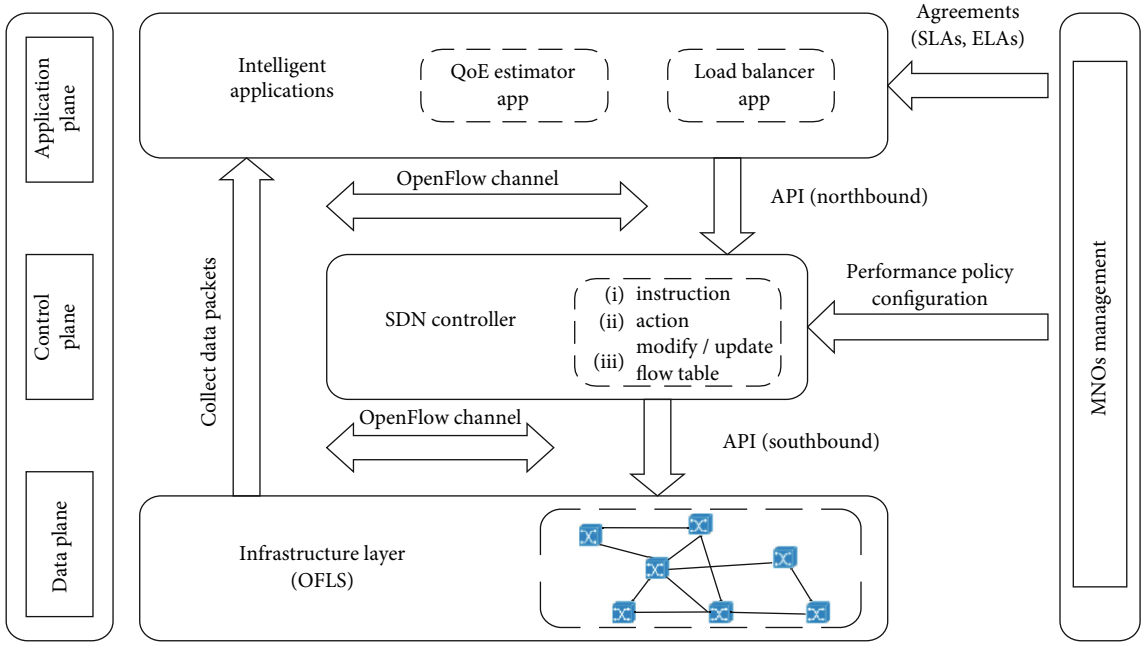


Figure 3: The SDN framework architecture.

(b) Running load balancing rules to maximize QoE in the delivery of streaming video

(ii) A BBU pool based on SDN components

3.1. General Architecture. As shown in Figure 2, the designed architecture is based on a fully centralized C-RAN, where the SDN controllers near the BBU pool make all the decisions regarding the control plane. We propose to use these controllers with extended functionality to support the software-defined BBU-RRH mapping mechanism. The controller implements the best plane for load balancing to maximize the QoE of the user and to ensure the efficient use of BBU pool resources. The calculation of the appropriate BBU-RRH ratio is based on the entire state of the network provided by the data plane. PHY and MAC layer resources are executed in the BBU pool and the RRH only runs the RF activities.

3.2. An Intelligent SDN Framework. The framework is implemented in the SDN controller following the premises of the self-organizing network (SON), which is able to simplify operational tasks through reconfiguration, optimization, and self-healing. When a video streaming service is triggered or when the control plane detects a new mobile node linked to the network, there is a need to determine whether the level of QoE with regard to physical resources complies with the defined service policies, between the MNOs and users. If the result is unsatisfactory, a load balancing action is triggered, and new features of the physical BBUs are allocated or new BBUs are instantiated to the BBU pool. On the other hand, when the available physical resources are sufficient to maintain the QoE at a satisfactory level, a resource optimization scheme is processed.

Considering the above conditions, we propose BBU-RRH switching schemes based on the predicted QoE. These schemes determine the proper combinations between BBUs and RRHs to accommodate the traffic load of BBU pool for a constant time interval in order to maximize QoE and minimize the number of allocated BBUs.

As depicted in Figure 3, our framework was established by integrating three basic components that comprise the three SDN planes: application plane, control plane, and data plane. Each component has its attributes and related methods. The particular features of each component are described below.

3.2.1. Application Plane. Consists of a QoE forecasting system for video streaming called "QoE Estimator", based on the learning provided by ANNs, namely, the Nonlinear AutoRegressive with eXogenous inputs (NARX) [21], which can analyze the data plane and manipulate the control plane, using the OpenFlow protocol.

NARX are recurring dynamic networks with feedback connections that involve many layers of the network to predict a given time series based on past value feedback input or another time series (external or exogenous) [22]. The use of these networks is justified by their ability of providing support for the automated SDN framework on account of its adaptability in different environments. However, to make the prediction, the ANNs carry out a supervised learning procedure, which allows them to progressively improve their performance as they interact with the environment, which results in a generalization of outputs.

Table 1: PSNR to MOS conversion.

PSNR (dB)	MOS
>37	5 (excellent)
31–36.9	4 (good)
25–30.9	3 (fair)
20–24.9	2 (poor)
<19.9	1 (bad)

In this sense, the following strategy was adopted for the training phase of the NARX. First, one should assume that the output of the NARX network is an estimate of the output of some nonlinear dynamic systems. This output is fed back to the input of the feedforward neural network as part of the standard NARX architecture. As the true output is available during the training phase, a series-parallel or open-loop architecture can be created, in which the true output is used instead of feeding back the estimated output. The main reasons for this strategy are as follows: (1) the input to the feedforward network is more accurate and (2) the resulting network has a pure feedforward architecture and static backpropagation can be employed for training [21].

However, for prediction, it is important to achieve the support for multistep-ahead, but the series-parallel configuration only provides one-step-ahead prediction. Thereby, the network is rearranged into the original parallel or closed-loop configuration, which can perform an iterated prediction over many time steps. Therefore, the training is carried out in an open-loop or series-parallel architecture, including the validation and testing phases. These characteristics allow the load balancing actions performed by the control plane to be proactive and accurate, based on recurring events.

In our proposal, the applied ANN works with multiple external inputs, such as video objective parameters (the percentage of the lost frame) and the data network conditions (traffic load, total packets received, and total packets lost), and predict an output series of peak signal noise ratio (PSNR), an objective video evaluation metric that compares the quality of the video received by the user against the original video, expressed in dB (decibels). Then, a MOS is obtained by subjectively classifying the PSNR on a five-point scale, as noted in Table 1. We chose this system because it is based on traditional metrics for QoE evaluation. These metrics provide a subjective/objective evaluation system and a video quality indicator that corresponds to the closest possible approximation of human perception. Moreover, it is capable of estimating video quality in realistic network conditions without any interaction with real-world viewers.

3.2.2. Control plane. To perform the mapping reset and load balancing processes, the control plane interacts directly with the BBU pool and the OpenFlow logic switches (OFLS). This allows that the controller is aware of the changes that occurred in physical resources in the network. Thus, the control plane performs basic parsing functions on the package

```
1.  BBUi: ith BBU (1<=i<=I);
2.  RRHj: jth RRH (1<=j<=J);
3.  tBBU; //BBU traffic load
4.  tRRH; //RRH traffic load
5.  Function star_BBU () {  //starts a new BBU
6.  star new BBU in BBU pool;
7.  }
8.  Function search_BBU (RRH,k){  //search for active BBUs to assign RRHs
9.    for (i=k+1; i<=I; i++){
10.     if ((QoE_Satisfactory) and (tBBUi <= Capacity_Limit(BBUi))) {
11.       assign (BBUi, RRH); //reset the BBU-RRH mapping
12.       tBBUi = tBBUi+tRRH  //update traffic load of BBUi
13.       return true;
14.     } else if (i=I){
15.       return false;
16.       }
17.   }
18. }
19.  While ((!QoE_Satisfactory) {  //output of the QoE Estimator
20.   for (i=1; i<=I; i++){
21.    for (j=1; k<=J; j++) {
22.    search RRH with lower tRRH; //search for rrh with lowest tRRH in BBUi
23.    }
24.    if (search_BBU (tRRHj,i)) {
25.     unassign(RRHj, BBUi); //unassign RRHj in BBUi
26.     add_newFlowTables (OFLS); //adds new flow tables to OFLS
27.     tBBUi = tBBUi-tRRHj; //update traffic load of BBUi
28.    else
29.    star_BBU ();
30.    add_newFlowTables (OFLS); //adds new flow tables to OFLS
31.   }
32.  }
33. }
```

Figure 4: The pseudocode of the load balancing algorithm (Case 1 and Case 2).

```
1.  BBUi: ith BBU (1<=i<=I);
2.  RRHj: jth RRH (1<=j<=J);
3.  tBBU; //BBU traffic load
4.  tRRH; //RRH traffic load
5.  k=1;
6.  While (QoE_Satisfactory) { //output of the QoE Estimator
7.     for (i=1; i<=I; i++){
8.       while (tBBUi <= Capacity_Limit (BBUi)){
9.        for (j=k; j<=J; j++){
10.        assign (BBUi, RRHj); //assign RRHs to BBU up to capacity limit
11.        add_newFlowTables (OFLS); //adds new flow tables to OFLS
12.        tBBUi = tBBUi+tRRHj; //update traffic load of BBUi
13.        k=j;
14.       }
15.      }
16.      if(i<I){
17.      shutdown (BBUi+1); //shutdown idle BBU
18.     }
19    }
20. }
```

Figure 5: The pseudocode of the optimize feature algorithm of BBU (Case 3).

headers that pass through the OFLS interfaces. OFLS interfaces are logical ports where packages in and out of the OpenFlow pipeline, used to transmit packets between OpenFlow processing and the rest of the network. In this manner, the controller can add, update, and delete flow entries in flow tables, both reactively (in response to packets) and proactively.

The controller monitors the load on each BBU and chooses the proper BBU-RRH configuration. Each BBU can handle multiple RRHs at a time and each RRH belongs to only one BBU at a period. The traffic load on each BBU is defined as the load sum generated by each RRH assigned and the traffic load of each RRH is attributed exclusively to the sum of loads generated by each associated UE. However, there is a hardware or software limitation on the number RRHs in each BBU, usually known as the hard capacity, as defined in [2]; the bandwidth, delays, and lost packets are used as parameters about the current state of physical resources. This approach makes the whole policy of load balancing and BBU-RRH mapping reset acts proactively.

Thus, two important aspects of C-RAN architecture are taken into account. The first is to distribute the loads between the BBUs, so that all the users are served equally, in accordance with their preferences. The second is to allow traffic loads generated in RRHs to be dynamically shared between BBUs, so that the provisioned network resources from the BBU pool are not underutilized. In all the processes, rules for packet streaming are added, updated, or modified in the OFLS tables.

To perform efficient BBU pooling functions, the framework is able to take into account three possible cases as follows.

Case 1. When the predicted QoE is considered to be unsatisfactory. In this case, we propose a BBU-RRH load balancing scheme, which determines the new BBU-RRH configurations to accommodate the traffic load of all RRHs without triggering new BBUs. As shown in Figure 4, firstly, when one considers the output of the QoE estimator as unsatisfactory (line 19), the algorithm selects RRHs with a low traffic load and that are assigned to the originating BBU (line 22), then it searches for target BBUs with minimum conditions to assign new RRHs (line 24 and line 10). In this manner, the originating BBU can cancel the RRH assignment (line 25) and updates its traffic load (line 26). This process is done for all RRHs until all traffic is accommodated and the predicted QoE is satisfactory.

Case 2. When the predicted QoE is deemed to be unsatisfactory and the conditions to apply load balancing of case 1 not

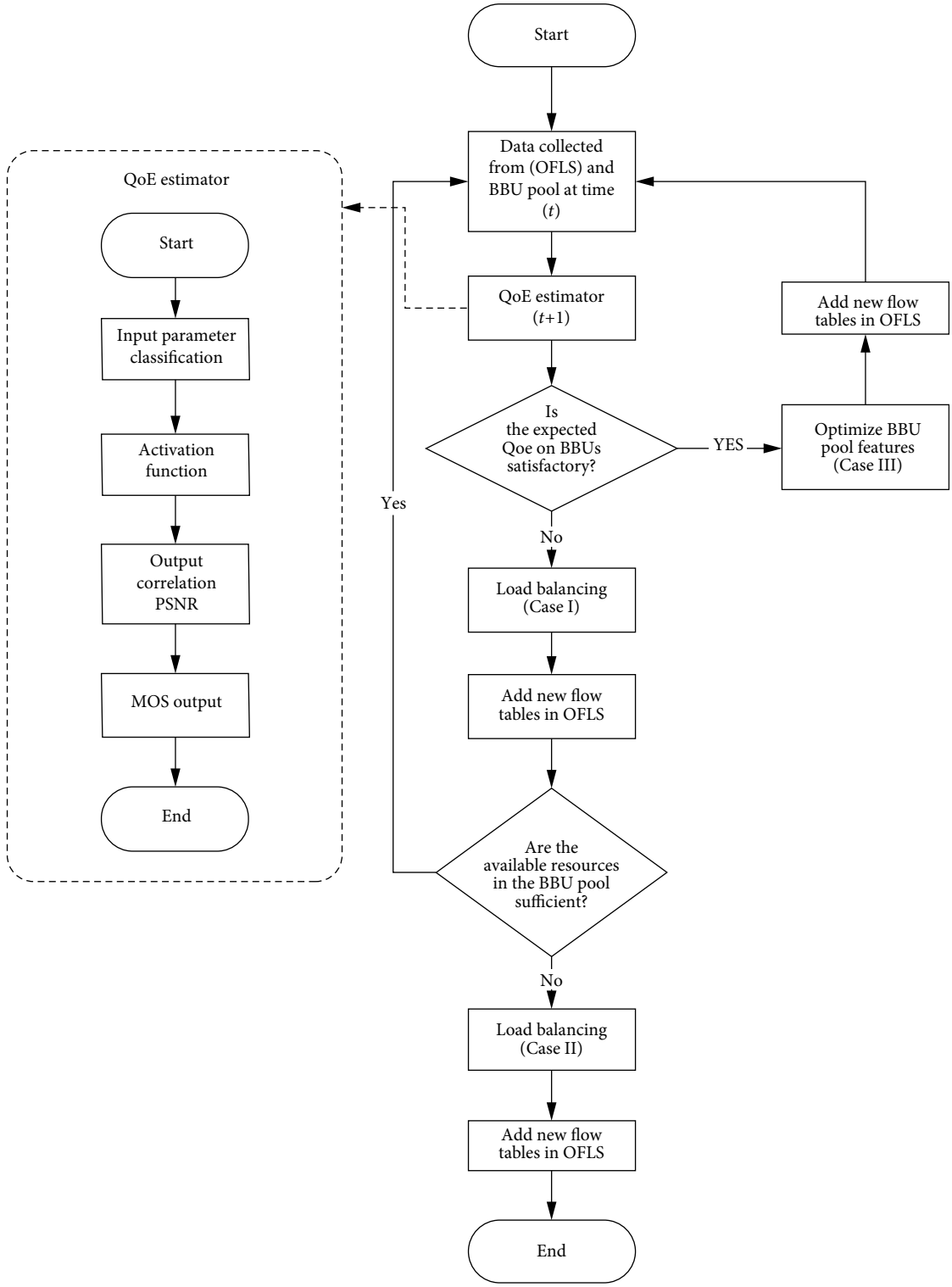


Figure 6: The flow chart of integrating framework components.

being met. In this case, we propose a load balancing scheme that triggers a new BBU to the BBU pool. As shown in Figure 4, the execution steps for this case include triggering a new BBU (line 29 and line 6), updating resources of the BBU pool, resetting the new mapping BBU-RRH (line 11), and updating OFLS flow tables (line 30). These steps end when the traffic from all RRHs are accommodated and the expected QoE is satisfactory.

Case 3. When the predicted QoE is considered to be satisfactory and the traffic load of the BBU is below the limit capacity. In this case, as shown in Figure 5, all RRHs in the BBU are

Table 2: Simulation parameters.

Elements	Attributes
BBU	Number of BBUs: min = 1, max = 3.
	Fronthaul: link 10 Gbps full duplex
	Fronthaul delay = 100 μs
RRH	Number of RRHs: 18
	Coverage area: 1000 m × 1000 m
	Distance between RRHs: 200 m
	Power: 46 dBm
	Bandwidth: 20 MHz
	System: FDD
	Allocation: GridPositionAllocator
	Pathloss: Coast-231
UE	Number of UEs: 200
	Power: 18 dBm
	Allocation: RandomBoxPositionAllocator
	Mobility: RandomWalk2D
	Velocity: 30% 30 km/h, 70% 3 km/h
	Simulation time: 30000 s
	Applications: RTP streaming (UHD video)

switched to other neighboring BBUs whose resource usages are lower than the capacity limit (line 8 and 10), which results in the deactivation the idle BBUs (line 17). This process ends when the entire traffic load of the RRHs is reallocated with the minimum possible BBUs.

3.2.3. Data Plane. It forwards packets through the OFLS devices. Its main function is to add or remove rules in the flow tables the ports of the OpenFlow appliances that connect the entire infrastructure of C-RAN in accordance to the rules established by the controller. Each flow table in the OFLS contains a set of flow entries; each stream entry consists of corresponding fields, counters, and a set of specific instructions to be applied to the packages if a corresponding entry is found. These features allow the entire traffic load (uplink and downlink) of the BBU pool, processed by the OpenFlow pipeline, to be balanced between BBUs.

Basic data plane actions include the following:

(i) Sending packet streams to the control plane

(ii) Updating or making modifications to the OFLS flow tables

Figure 6 presents a structured view of the functions for each component that integrates the architecture of the proposed framework.

4. Performance Evaluation

This section is devoted to the performance evaluation of the solution proposed in this article. The main objective here is to evaluate the gains in QoE performance that can be obtained with the deploy of the framework in a traditional C-RAN architecture. In Section 4.1, we define the underlying assumptions of the simulation, while in Section 4.2, we show and discuss the results of the experiments.

Table 3: ANN performance.

ANN stage	Performance (MSE)
Training	0.0541
Validation	0.0490
Test	0.0649
Closed-loop prediction	0.0757
Step-ahead prediction	0.0549

4.1. The Simulation Assumptions. The network simulator version 3 (NS3) was used to implement the C-RAN architecture and this involved adding and modifying some modules already consolidated in the scientific community. NS3 is a discrete-event network simulator based on open-source software and registered under the GNU license GPLv2 that is mainly employed for research and educational purposes; it is available to the public at https://www.nsnam.org/. The SDN controller was implemented with the aid of the OFSwitch13 module, which implements the OpenFlow version 1.3 [23], and simulates all activities of the control framework. The basic structure of a cellular network is obtained through the adaptation of some features of the LTE module for NS3, made available by the LTE-EPC network simulator (LENA) [24], which is a simulator of a cellular modular architecture widely used in the academic world. The scenario defined for the simulation was designed in accordance with the guidelines of the 3rd Generation Partnership Project (3GPP) [25], which propose the design principles for the next generation of mobile networks. The machine used for the experiments consists of Intel (R) Xeon (R) Silver 4114 processor; 2.20GHz frequency; 32 GB of RAM; Ubuntu 16.04.4 Lts Operating System. Besides, most of the configurations set for the simulations adopt the parameters already covered in other works, for example in [26].

The BBU pool initially consists of a maximum of 3 BBUs that are activated or deactivated according to the requirements of association and disassociation with regard to the UEs. RRHs are allocated statically in an area of 1000m^2 at 30 m from the surface, making as total of 18 BSs. UEs are inserted 1.5 m from the surface and allocated randomly one by one. Each UE consumes UHD video streaming applications and moves randomly, which allow handovers between the BSs whenever conditions are favorable.

The videos used in this simulation were "crowd run," "ducks take off," and "park joy," all in y4m format and MPEG-4 encoded UHD resolution in a publicly accessible repository, which can be accessed at https://media.xiph.org/video/derf/. In each simulation, the number of connected UEs varies between 1 and 200 every round, which yields 200 examples per round. Each video was streaming simultaneously to all users for a period of five rounds, which yields 1000 examples per simulation. In all experiments, we use a random number seed. This factor is important when

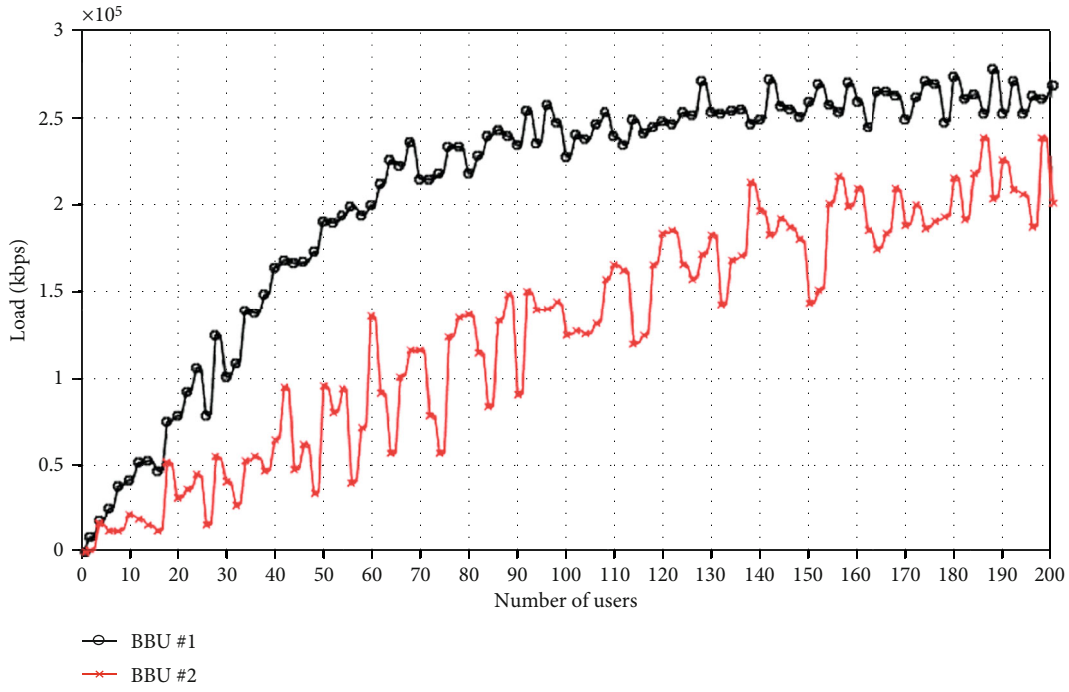


Figure 7: Total workload on each BBU (traditional C-RAN).

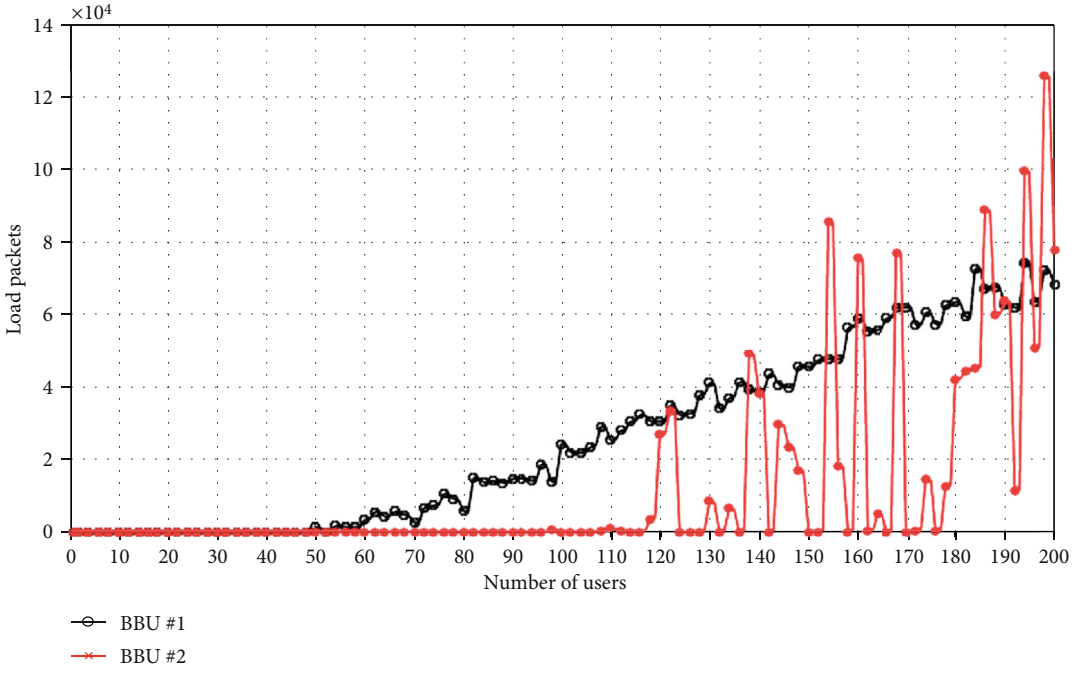


Figure 8: Total packets lost in each BBU (traditional C-RAN).

assessing the benefits of predict derived from the different experimental scenarios.

When the performance of the framework is evaluated, we define user's QoE satisfaction policies, as being experience level agreements in which the PSNR and MOS values are equal to, or higher than, 25 dB and 3-Fair, respectively. These values are explained by the intermediate levels between the minimum and maximum of each metric, as noted in Table 1. The configuration parameters of the simulated environment were established according to [19, 20], and the complete list of parameters is shown in Table 2.

The ANN that acts on the QoE estimator was trained by division of random indices, together with the Levenberg Marquardt optimized backpropagation algorithm [27].

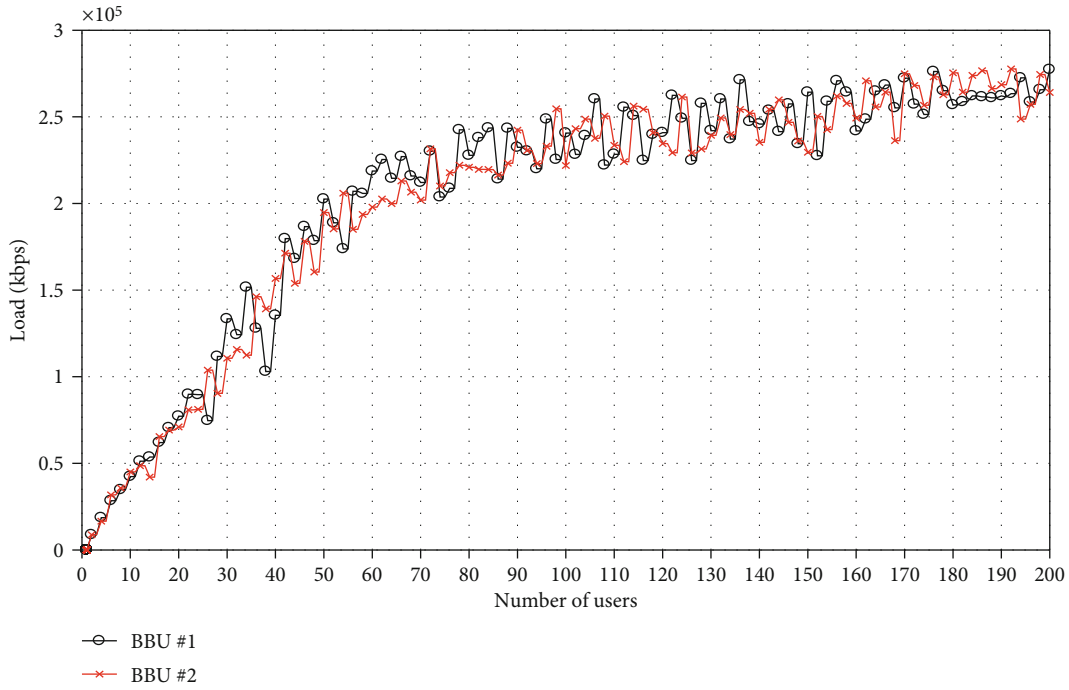


Figure 9: Load balancing (Case 1).

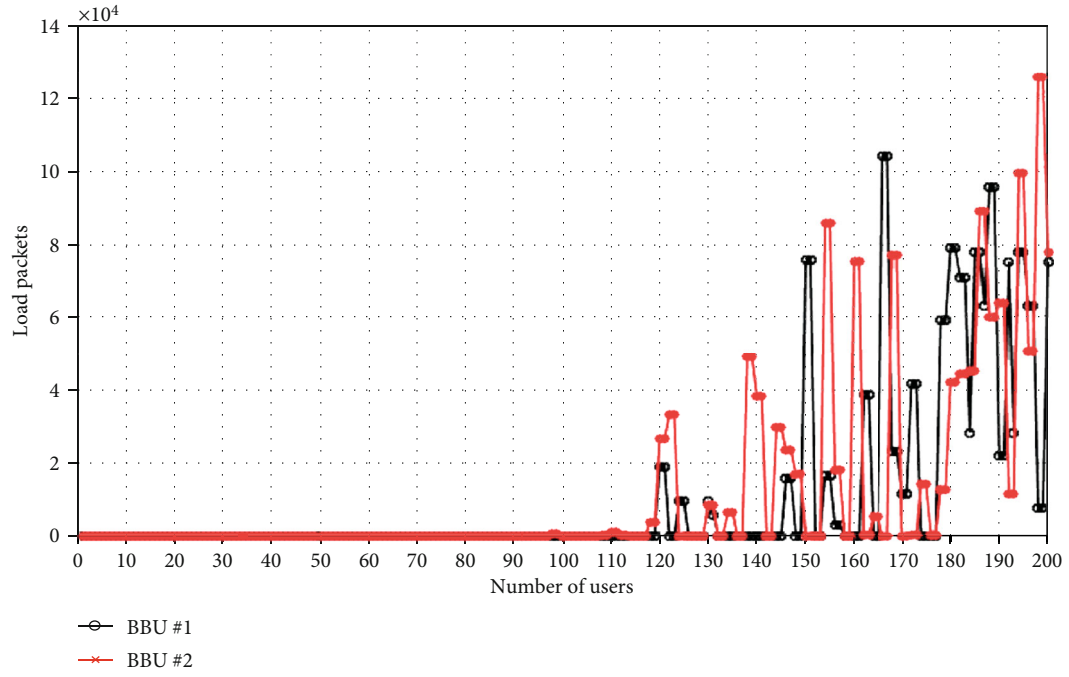


Figure 10: Total of lost packets in each BBU after load balancing (Case 1).

There was a sample division of 70%, 15%, and 15%, for the selection of groups for training, testing, and validation, respectively, which were used with epochs of cycles that obeyed the criteria of the premature stop to avoid overfitting. Besides, the number of input delays, number of feedback delays, number of neurons in the hidden layer, and the maximum number of epochs were 2, 2, 15, and 1000, respectively. This configuration represents the best possible configuration obtained by numerous trials and error methods.

To find the network simulation that best fits the data and produces the most accurate forecasts, two types of ANN prediction models were employed—closed-loop prediction and sep-ahead predictions. The closed-loop prediction model entails providing input data values obtained over time $(t-1)$,

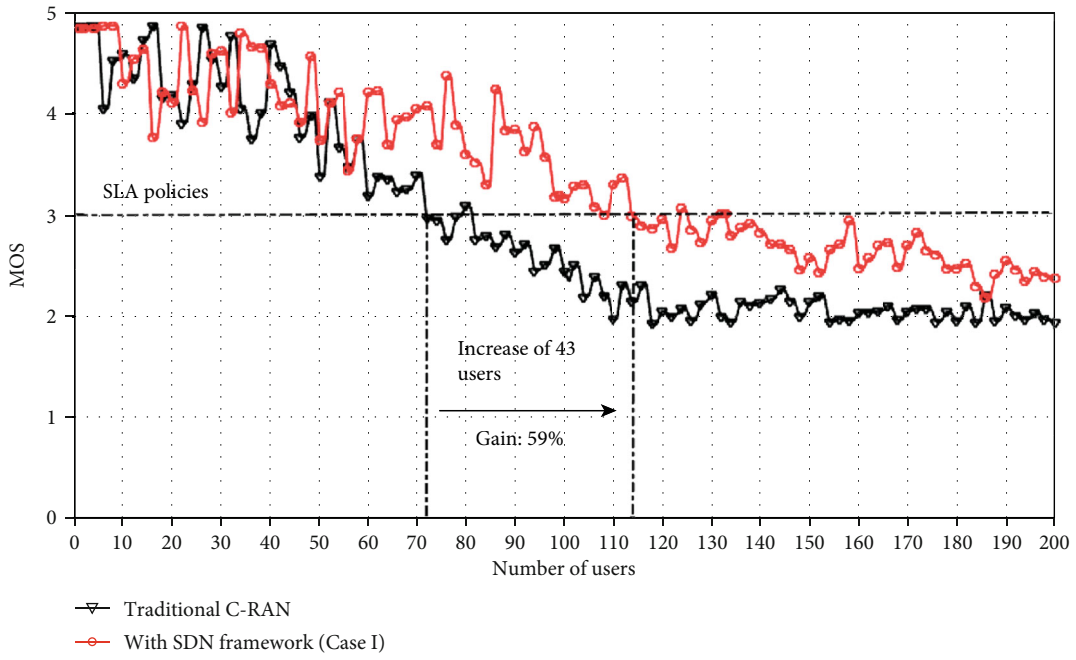


Figure 11: Comparison of MOS after load balancing (Case 1).

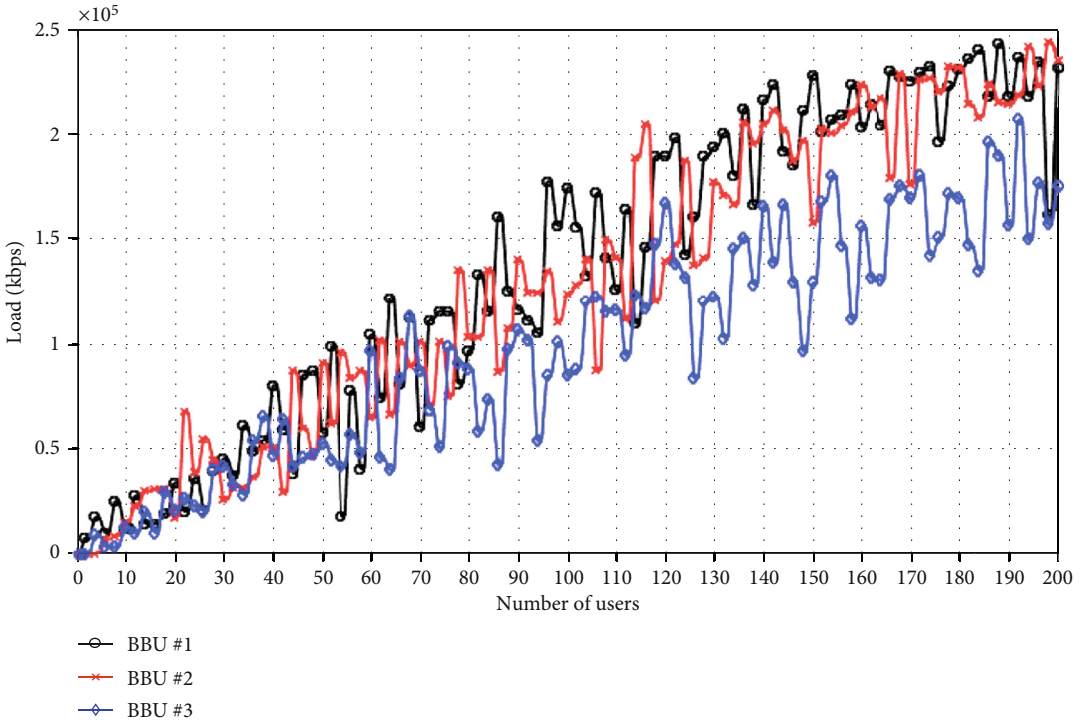


Figure 12: Load balancing (Case 2).

which also have output values in time (t). In contrast, the step-ahead prediction model involves providing input time (t) values and obtaining outputs at the time ($t + 1$). The performance of each model is obtained utilizing the mean square error (MSE) metric, a function that results from estimating the difference between the actual output and the output calculated by ANN, as observed in Equation (1) [28]. We decided to use this metric because it is a widely used function in the evaluation of ANNs.

$$\text{MSE} = \frac{1}{n}\sum_{j=0}^{n}(y_i - y\wedge_i)^2, \tag{1}$$

where n is the data number of the points, y_i represents the observed values, and $\hat{y}_i$ represents the predicted values.

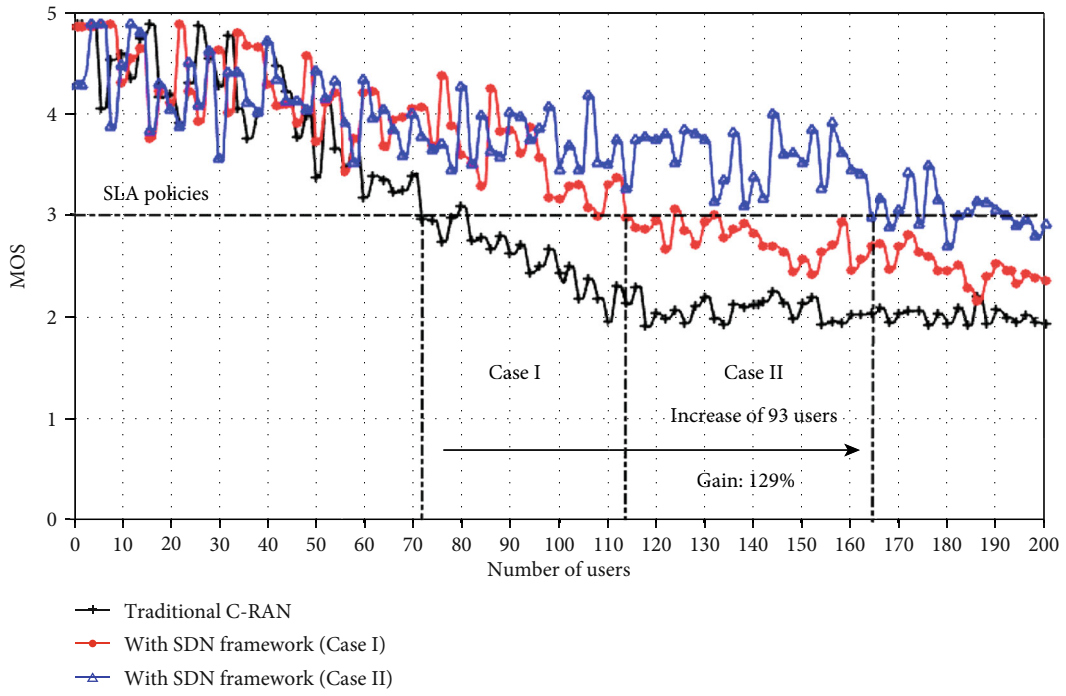


Figure 13: Comparison of MOS after load balancing (Case 2).

The results obtained in the evaluation are shown in Table 3, where the network performance can be observed at each stage in the execution of ANN and in the two models evaluated. Thus, the step-ahead prediction model achieved the best performance in predicting all data, and for this reason, it was used as a standard model for QoE prediction within the established framework.

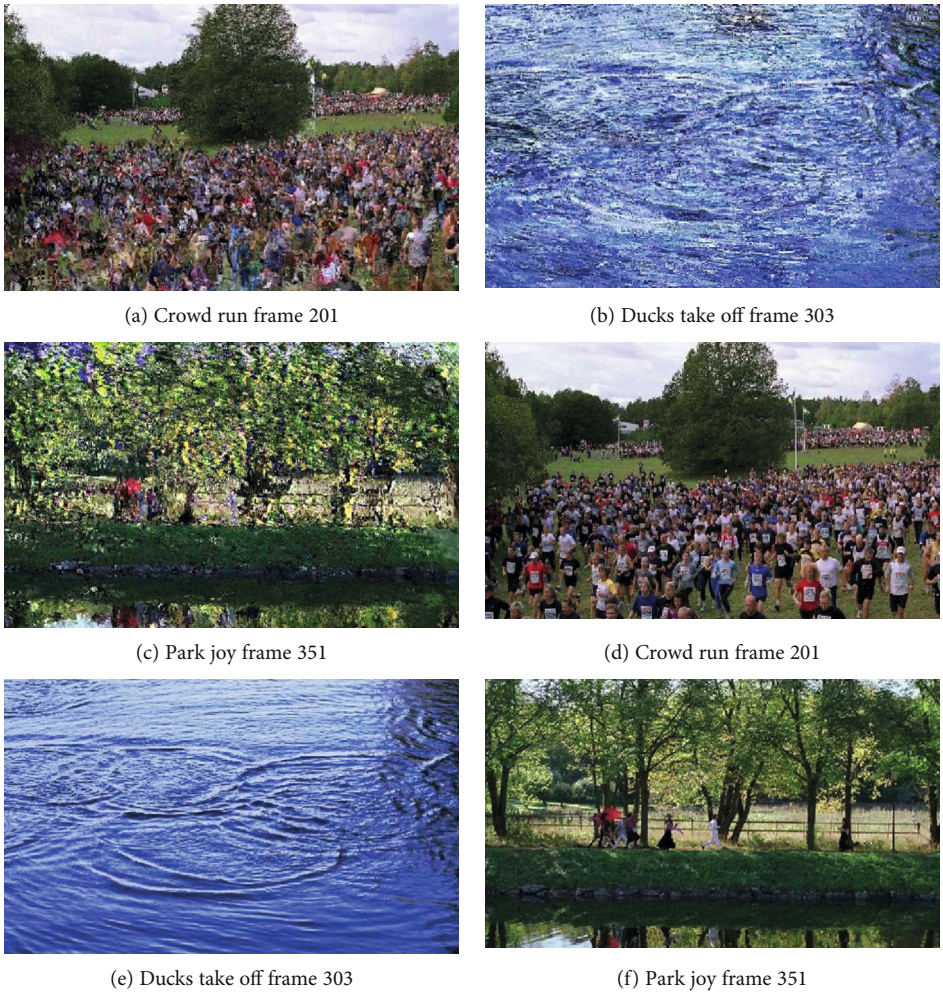
4.2. Numerical Results. First, users are randomly allocated into a traditional C-RAN scenario with two BBUs (#1 and #2), where each BBU is connected to a maximum of 9 RRHs. The purpose of this is to assess the impact of data traffic generated in the BBUs in relation to the number of active users and the satisfaction rating of these. All results obtained in the simulations had a confidence interval of 95% and represent the average of several runs.

In light of this, the results shown in Figure 7 suggest that as the number of associated UEs to the BBU pool increases, the load curves generated by BBU #1 and the BBU #2 tend to diverge, that is, it is observed that the traffic load generated by the RRHs assigned to BBU #1 is higher than that from the RRHs assigned to BBU #2. This can probably be attributed to the initial random allocation of the UEs within the coverage area of the RRHs every round and/or inter-BBU-pool handover processes [29], as there are no restrictions in this area. This characteristic can be observed in the load oscillations (peaks) presented by each BBU. In Figure 8, for the same experiment, we observed a significant growth in the number of packages not processed by BBU #1 (around 70%) when compared to BBU #2. This can probably be attributed to the overload on BBU caused by capacity limitations in hardware and/or software.

The same experiments were carried out in an identical scenario, although in this case, we regarded the QoE prediction mechanism as a key factor in the delivery of network services. The aim was to observe the behavior of the network in response to the predictive guidelines and load balancing (Cases 1 and 2) requirements imposed by the framework. Hence, for better understanding, we will discuss the results of each case separately.

For Case 1, we evaluated the performance of the framework in relation to the number of users served in accordance with the preestablished SLA policies (MOS equal to or greater than 3 points) and the capacity limitations in hardware and software of the BBUs active in the BBU pool. The results, shown in Figure 9, revealed that the framework was able to balance the loads on BBUs #1 and #2 as it acted proactively, updating flows and redefining the BBU-RRH mapping. It achieved this through the traffic load balancing scheme proposed in the algorithm depicted in Figure 4, that is, a balance in the packet flow was found between the BBUs, which resulted in the mapping of 7 and 11 RRHs assigned BBUs #1 and #2, respectively. As shown in Figure 10, this led to the reduction in the number of lost packets that had previously been generated by BBU #1.

In Figure 11, we compared the results related to the MOS observed in the previous experiments. Thus, it was possible to conclude that in a traditional C-RAN architecture the results of MOS, in compliance with SLA policies, served approximately 36% of all users observed. However, with the inclusion of the framework, we observed that 55% of users rated streaming video as (fair, good, or excellent), demonstrating a gain of approximately 59% compared to the traditional architecture model.



(a) Crowd run frame 201

(b) Ducks take off frame 303

(c) Park joy frame 351

(d) Crowd run frame 201

(e) Ducks take off frame 303

(f) Park joy frame 351

Figure 14: Frame comparison in video sequences.

For Case 2, we evaluated the performance of the framework in situations where the limiting factor in meeting SLA policies is justified exclusively by hardware or software limitations presented by BBU. This can be seen in Figure 10, where the number of lost packets increases from the 120th UEs association in the BBU pool. In this case, the framework triggers new BBU whenever the capacity limit of hardware or software is not sufficient to serve new UEs. In Figure 12, the results reveal that when triggering BBU #3, the total traffic load of the RRHs in the BBU pool is reallocated among the BBUs (respecting the capacity limit of each BBU). In the end, a new BBU-RRH mapping is obtained, with 7, 7, and 4 RRHs assigned BBUs #1, #2, and #3, respectively. This is possible because, to redefine a new BBU-RRH mapping, the framework analyzes separately the traffic load of each RRH, which is also highlighted in the algorithm seen in Figure 4.

In Figure 13, we compared the MOS results observed in the three experiments presented (traditional C-RAN and C-RAN with the SDN framework in Cases 1 and 2). The results revealed that, with the SDN framework (Case 2), approximately 82.5% of users rated video streaming as excellent, good or fair, which means a gain of around 129% compared to traditional C-RAN architecture and 43% in relation the load balancing proposed in Case 1.

Figure 14 shows the visual gains achieved with the application of the framework. The effects of the load balancing scheme and BBU-RRH mapping reset can be observed by analyzing the comparative sequence of frames (selected at random) of the three videos used in the experiments. Figures 14(a)–14(c) represent the frames transmitted in a traditional C-RAN architecture and Figures 14(d)–14(f) represent the frames obtained from applying the framework (Case 2).

5. Conclusions

In this paper, we investigated the opportunities that software-defined networks can provide to C-RAN architectures. By leveraging the advantages of SDN-based logical centralization and ANN predictability, we were able to establish a high-level framework for the user that can ensure better QoE for the service agreements (SLA) of video streaming than that offered by traditional C-RAN architectures. A new load balancing algorithm of three stages based in QoE

predictions was proposed to solve the BBU-RRH mapping problem. The analytical results revealed that the framework is able to guarantee gains in QoE between 59% and 129% compared to the traditional C-RAN architecture model. It is recommended that future projects include investigations of on/off BBU methods to optimize grid energy consumption while taking into account the guarantees in the SLA agreements.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the PROPESP/UFPA, CAPES under grant (20130056) and CNPq PQ under grant (31079920180).

References

[1] Index, Cisco Visual Networking, *Global mobile data traffic forecast update, 2017-2022*, Cisco white paper, 2019.

[2] M. Khan, R. S. Alhumaima, and H. S. Al-Raweshidy, "Quality of service aware dynamic BBU-RRH mapping in cloud radio access network," in *2015 International Conference on Emerging Technologies (ICET)*, pp. 1–5, Peshawar, 2015.

[3] R. Serral-Gracià, E. Cerqueira, M. Curado, M. Yannuzzi, E. Monteiro, and X. Masip-Bruin, "An overview of quality of experience measurement challenges for video applications in IP networks," in *Wired/Wireless Internet Communications*, E. Osipov, A. Kassler, T. M. Bohnert, and X. Masip-Bruin, Eds., pp. 252–263, Springer, 2010.

[4] M. Varela, P. Zwickl, P. Reichl, M. Xie, and H. Schulzrinne, "From service level agreements (SLA) to experience level agreements (ELA): the challenges of selling QoE to the user," in *2015 IEEE International Conference on Communication Workshop (ICCW)*, pp. 1741–1746, London, 2015.

[5] E. J. Kitindi, S. Fu, Y. Jia, A. Kabir, and Y. Wang, "Wireless network virtualization with SDN and C-RAN for 5G networks: requirements, opportunities, and challenges," *IEEE Access*, vol. 5, pp. 19099–19115, 2017.

[6] O. Awobuluyi, J. Nightingale, Q. Wang, and J. M. Alcaraz-Calero, "Video quality in 5G networks: context-aware QoE management in the SDN control plane," in *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, pp. 1657–1662, Liverpool, 2015.

[7] D. Mishra, P. C. Amogh, A. Ramamurthy, A. A. Franklin, and B. R. Tamma, "Load-aware dynamic RRH assignment in cloud radio access networks," in *2016 IEEE Wireless Communications and Networking Conference*, pp. 1–6, Doha, 2016.

[8] D. Pompili, A. Hajisami, and H. Viswanathan, "Dynamic provisioning and allocation in cloud radio access networks (C-RANs)," *Ad Hoc Networks*, vol. 30, pp. 128–143, 2015.

[9] S. Namba, T. Warabino, and S. Kaneko, "BBU-RRH switching schemes for centralized RAN," in *7th International Conference on Communications and Networking in China*, pp. 762–766, Kun Ming, China, 2012.

[10] J. Wu, Z. Zhang, Y. Hong, and Y. Wen, "Cloud radio access network (C- RAN): a primer," *IEEE Network*, vol. 29, no. 1, pp. 35–41, 2015.

[11] W.-C. Chien, C.-F. Lai, and H.-C. Chao, "Dynamic resource prediction and allocation in C-RAN with edge artificial intelligence," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4306–4314, 2019.

[12] Y. Zhang, F. Barusso, D. Collins, M. Ruffini, and L. A. DaSilva, "Dynamic allocation of processing resources in cloud-RAN for a virtualised 5G mobile network," in *2018 26th European Signal Processing Conference (EUSIPCO)*, pp. 782–786, Rome, 2018.

[13] M. Khan, R. S. Alhumaima, and H. S. al-Raweshidy, "QoS-aware dynamic RRH allocation in a self-optimized cloud radio access network with RRH proximity constraint," *IEEE Transactions on Network and Service Management*, vol. 14, no. 3, pp. 730–744, 2017.

[14] M. Y. Lyazidi, N. Aitsaadi, and R. Langar, "Dynamic resource allocation for Cloud-RAN in LTE with real-time BBU/RRH assignment," in *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6, Kuala Lumpur, Malaysia, 2016.

[15] J. Yao and N. Ansari, "QoS-aware joint BBU-RRH mapping and user association in cloud-RANs," *IEEE Transactions on Green Communications and Networking*, vol. 2, no. 4, pp. 881–889, 2018.

[16] E. Aqeeli, A. Moubayed, and A. Shami, "Power-aware optimized RRH to BBU allocation in C-RAN," *IEEE Transactions on Wireless Communications*, vol. 17, no. 2, pp. 1311–1322, 2018.

[17] Y.-S. Chen, W.-L. Chiang, and M.-C. Shih, "A dynamic BBU–RRH mapping scheme using borrow-and-lend approach in cloud radio access Networks," *IEEE Systems Journal*, vol. 12, no. 2, pp. 1632–1643, 2018.

[18] E. A. Ramos da Paixao, R. F. Vieira, W. V. Araujo, and D. L. Cardoso, "Optimized load balancing by dynamic BBU-RRH mapping in C-RAN architecture," in *2018 Third International Conference on Fog and Mobile Edge Computing (FMEC)*, pp. 100–104, Barcelona, 2018.

[19] Y. S. Chen, F. Y. Liao, and Y. K. Kan, "A bandwidth adaptation scheme for cloud radio access networks," in *Ad Hoc Networks*, Y. Zhou and T. Kunz, Eds., pp. 234–245, Springer, 2017.

[20] Y.-S. Chen, C.-S. Hsu, and F.-Y. Liao, "A bandwidth adaptation mechanism for cloud radio access networks," *Pervasive and Mobile Computing*, vol. 40, pp. 639–659, 2017.

[21] Q. Zhang and L. Ljung, "Multiple steps prediction with nonlinear ARX models," *IFAC Proceedings Volumes*, vol. 37, no. 13, pp. 309–314, 2004.

[22] X. Chen, J. Racine, and N. R. Swanson, "Semiparametric ARX neural-network models with an application to forecasting inflation," *IEEE Transactions on Neural Networks*, vol. 12, no. 4, pp. 674–683, 2001.

[23] L. J. Chaves, I. C. Garcia, and E. R. M. Madeira, "OpenFlow-based mechanisms for QoS in LTE backhaul networks," in *2016 IEEE Symposium on Computers and Communication (ISCC)*, pp. 1233–1238, Messina, 2016.

[24] N. Baldo, M. Miozzo, M. Requena-Esteso, and J. Nin-Guerrero, "An open source product-oriented LTE network simulator

based on ns-3," in *In Proceedings of the 14th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems (MSWiM '11). ACM*, pp. 293–298, New York, NY, USA, 2011.

[25] 3GPP Technical Report 38.913, *Study on Scenarios and Requirements for Next Generation Access Technologies (Release 14)*, 2017, v14.2.0.

[26] M. Mezzavilla, M. Zhang, M. Polese et al., "End-to-end simulation of 5G mmwave networks," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2237–2263, 2018.

[27] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[28] C. Chen, J. Twycross, and J. M. Garibaldi, "A new accuracy measure based on bounded relative error for time series forecasting," *PLoS One*, vol. 12, no. 3, article e0174202, 2017.

[29] H. Zhang, C. Jiang, J. Cheng, and V. C. M. Leung, "Cooperative interference mitigation and handover management for heterogeneous cloud small cell networks," *IEEE Wireless Communications*, vol. 22, no. 3, pp. 92–99, 2015.