UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Telecomunicació
i Aeroespacial de Castelldefels

MASTER THESIS

TITLE: Resource Management with adaptive capacity in C-RAN

MASTER DEGREE: Master's Degree in Applied Telecommunications and Engineering Management

AUTHOR: Rolando Guerra-Gómez

ADVISOR: Silvia Ruíz Boqué

DATE: February 12, 2020

Title : Resource Management with adaptive capacity in C-RAN
Author: Rolando Guerra-Gómez

Advisor: Silvia Ruíz Boqué

Date: February 12, 2020

ABSTRACT

Efficient computational resource management in 5G Cloud Radio Access Network (C-RAN) environments is a challenging problem because it has to account simultaneously for throughput, latency, power efficiency, and optimization tradeoffs. This work proposes the use of a modified and improved version of the realistic Vienna Scenario that was defined in COST action IC1004, to test two different scale C-RAN deployments. First, a large-scale analysis with 628 Macro-cells (Mcells) and 221 Small-cells (Scells) is used to test different algorithms oriented to optimize the network deployment by minimizing delays, balancing the load among the Base Band Unit (BBU) pools, or clustering the Remote Radio Heads (RRH) efficiently to maximize the multiplexing gain. After planning, real-time resource allocation strategies with Quality of Service (QoS) constraints should be optimized as well. To do so, a realistic small-scale scenario for the metropolitan area is defined by modeling the individual time-variant traffic patterns of 7000 users (UEs) connected to different services. The distribution of resources among UEs and BBUs is optimized by algorithms, based on a realistic calculation of the UEs Signal to Interference and Noise Ratios (SINRs), that account for the required computational capacity per cell, the QoS constraints and the service priorities.

However, the assumption of a fixed computational capacity at the BBU pools may result in underutilized or oversubscribed resources, thus affecting the overall QoS. As resources are virtualized at the BBU pools, they could be dynamically instantiated according to the required computational capacity (RCC). For this reason, a new strategy for Dynamic Resource Management with Adaptive Computational capacity (DRM-AC) using machine learning (ML) techniques is proposed. Three ML algorithms have been tested to select the best predicting approach: support vector machine (SVM), time-delay neural network (TDNN), and long short-term memory (LSTM). DRM-AC reduces the average of unused resources by 96 %, but there is still QoS degradation when RCC is higher than the predicted computational capacity (PCC). For this reason, two new strategies are proposed and tested: DRM-AC with pre-filtering (DRM-AC-PF) and DRM-AC with error shifting (DRM-AC-ES), reducing the average of unsatisfied resources by 99.9 % and 98 % compared to the DRM-AC, respectively.

This work was supported in part by the Spanish ministry of science through the project RTI2018-099880-B-C32, with ERFD funds, and the Grant FPI-UPC provided by the UPC. It has been done under COST CA15104 IRACON EU project.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1. INTRODUCTION

1.1. C-RAN Overview

The current paradigm on mobile communication uses base stations as responsible for carrying out the Radio-Frequency (RF) functionalities and the baseband processing required to organize and schedule the transmission between the User Equipments (UEs) and the Core Network. The fundamental strategy to increase the capacity under this paradigm is becoming the network denser by introducing Small Cells (SCells), but this approach increases the inter-cell interference and cost. On the other side, the massive growth of mobile data traffic and the creation of new technologies as Internet of Things (IoT), Augmented Reality (AR), and autonomous vehicles have purchased mobile network operator and the research community to design new Radio Access Network (RAN) architectures for fifth-Generation (5G) systems.

Cloud Radio Access Network (C-RAN) is seen as a key technology to enable 5G systems, defined by [1]. C-RAN has been an interesting research field for many authors in recent years. The RF and baseband functionalities are separated; depending on the type of functional split option, a portion or all the baseband functionalities are centralized and shared among sites in the virtualized Baseband Unit (BBU) pools. The RF and the portion of baseband functions that are not centralized remain in the Remote Radio Head (RRH) device. Fig. 1.1 shows a general C-RAN architecture. Due to the separation between BBUs and RRHs, a fronthaul link is needed to communicate those entities.

RRHs transmit the In-phase and Quadrature (IQ) signals from UEs through the fronthaul link using synchronous protocol Common Public Radio Interface (CPRI). However, a comparison among CPRI, Analogue Radio-over-Fiber (ARoF), and Physical Layer Split (PLS) optical fronthaul networks has been addressed in [2], concluding that cost-effective solutions for 5G scenarios could be achieved by means of PLS and ARoF architectures.

As BBU pools centralize and virtualize the resources to handle dynamically many RRHs, data traffic from different types of cells to the backhaul link is aggregated, favoring the apparition of multiplexing gain if the traffic peaks of the cells are not overlapped in time. Moreover, some of the most promising techniques of Long Term Evolution (LTE) Advanced as Coordinated Multipoint (CoMP), enhanced Inter-Cell Interference Coordination (eICIC), and beamforming could be easily implemented thanks to C-RAN structure, contributing considerably to improve 5G network performance.

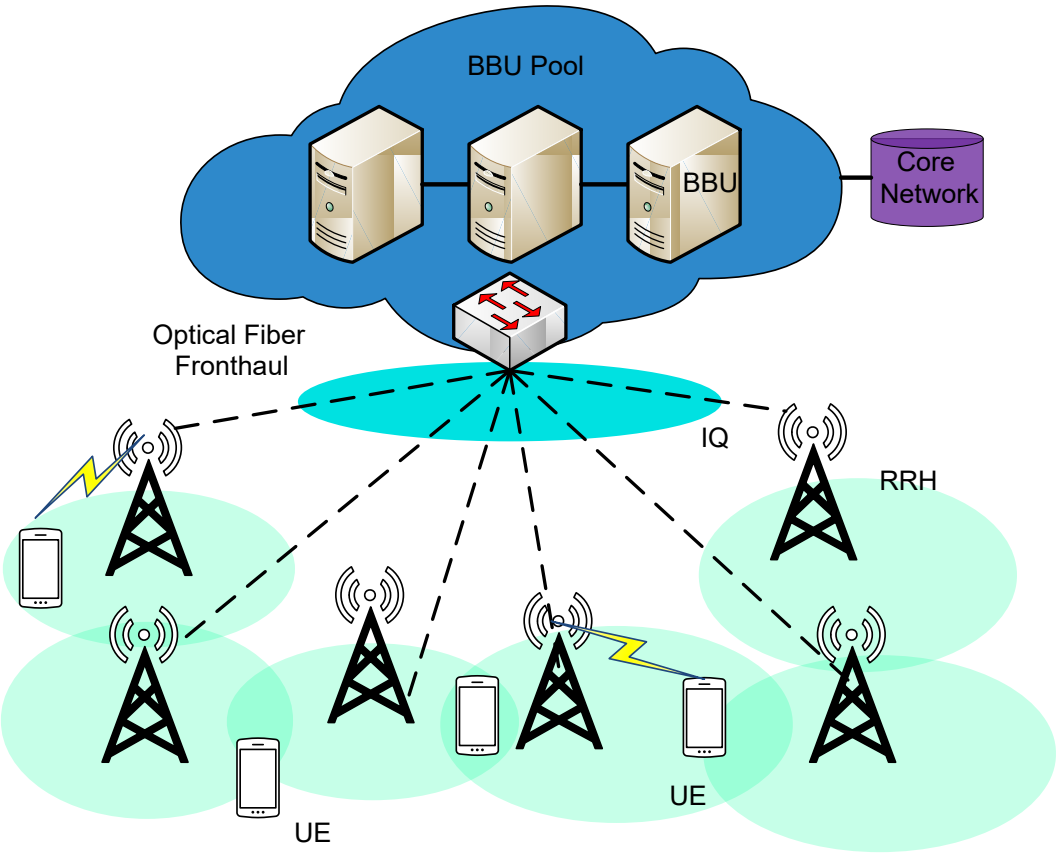


Figure 1.1 General C-RAN architecture.

1.1.1. Advantages of C-RAN

1.1.1.1. Adaptability to non-uniform traffic and efficient use of resources

Mobile networks face dynamic environments with high user mobility and fluctuating data traffic profiles. Fig. 1.2 shows an example of the traffic profile behavior of mobile networks along a day. The traffic profile presents high variations on a daily and weekly basis. The busy hours (peak of traffic) of each cell depends on the zone where the base station is allocated (office, residential, and mixed). Often, the demand in office cells begins to increase at 10:00 am, remaining at a high level until 6:00 pm when people move home, resulting in a drastic decrease in demand.

On the contrary, traffic becomes the highest in the evening at residential sites. This behavior is called: tidal effect. In the current paradigm of mobile communications (second-Generation (2G), third-Generation (3G), and fourth-Generation (4G)), each Base Station (BS) must have the processing capacity to satisfy the maximum demand of the cell, which results in inefficient resource utilization. However, as C-RAN centralizes the baseband processing capabilities in BBU pools, it adapts to non-uniform traffic profiles addressing the tidal effect with more efficient use of resources. This strategy produces a multiplexing gain because peaks of the traffic at different cells are not overlapped in time.

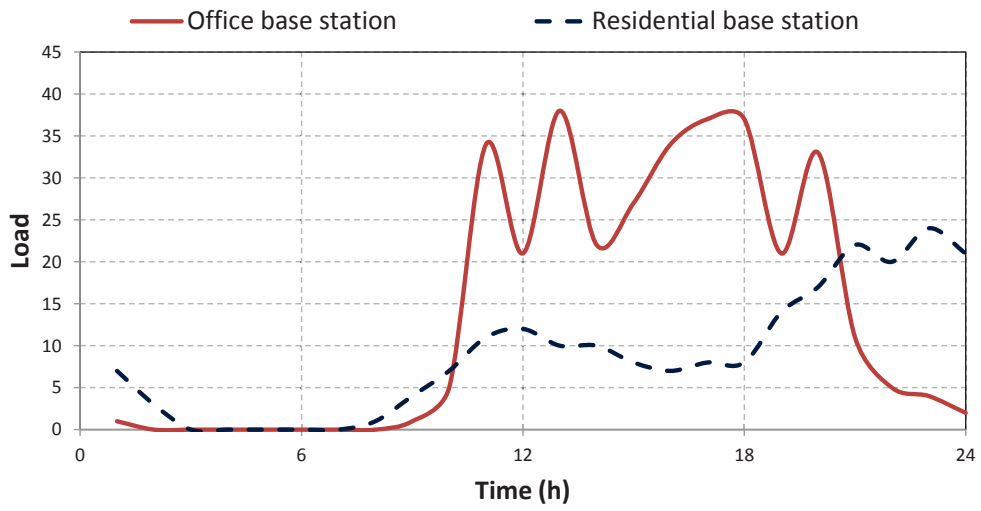


Figure 1.2 Daytime traffic profile depending on base station location [1, 3]

1.1.1.2. Energy and cost saving

Other improvements oriented to increase mobile network capacity have been introduced, such as Multiple-Input Multiple-Output (MIMO) and beamforming. The most successful strategy has been the deployment of SCells or Pico Cell (PCell). However, the enormous growth in mobile data traffic produced by the apparition of new technologies like IoT, autonomous vehicles, or augmented reality causes that the deployment of BSs with all the functionalities becomes no suitable due to the associated high cost and reduced energy efficiency. On the contrary, cost and power consumption experiment a significant reduction by centralizing the BSs in a C-RAN strategy.

1.1.1.3. Performance, scalability and network maintenance

The centralized architecture of C-RAN contributes to increasing the performance of the mobile networks, the scalability allowing cell densification using low cost RRHs, and the agility of network maintenance. Some of the most promising techniques of LTE networks as eICIC, Joint Transmission (JT), and beamforming can be easy implemented and upgraded. The network maintenance is also improved because most of the resources are centralized at the BBU pools, and the RRHs remain as simple as possible, depending on the function splitting option.

With Wireless Network Virtualization (WNV), multiple Mobile Network Operator (MNO) can efficiently share various network resources through different network slices; hence, the Capital Expenditures (CAPEX) and Operating Expenditures (OPEX) can be reduced significantly [4]. Enormous technical challenges have to be addressed, such as management of infrastructure resources for different MNO to optimize desired design objectives while guaranteeing a high level of isolation in C-RAN.

1.2. Research objective

As it has been mentioned above, the BBU pools concentrate and virtualize the resources to dynamically handle multiple RRHs, aggregating data traffic from different types of cells to the backhaul link and favoring the increase of the multiplexing gain.

As a consequence of centralization in C-RAN, the management of the computational resources at BBU pools to satisfy the traffic demand of the RRHs becomes a challenge. Previous works on BBU pool resource allocation has relied on the definition of optimization problems such as mixed-integer linear programming (MILP) or multi-objective optimization. These strategies allocate the resources assuming that the instantiated computational capacity at BBU pools is fixed and equal to the maximum BBU pool capacity. Under this assumption, the computational resources could be over-provisioned or under-provisioned, causing inefficient resource utilization or Quality of Service (QoS) degradation, respectively.

This issue could be addressed, combining the flexibility of virtualization and the availability of machine learning techniques to predict computational demands. As the resources are virtualized, they could be instantiated dynamically according to an anticipated computational capacity demand. For this reason, this work proposes the integration of Dynamic Resource Management (DRM) with a prediction of the required computational capacity based on machine learning (ML) techniques. It allows defining a DRM with adaptive capacity (DRM-AC), to avoid under-utilization of the computational resources, and to maintain QoS. Performance is evaluated on a realistic C-RAN platform over the Vienna city, which takes into account the non-uniformity of wireless network environments.

1.3. Publications

The evolution of this research work has been periodically published. This section summarizes the presented papers on conferences and journals.

1. Rolando Guerra-Gómez, Silvia Ruiz, M. García-Lozano, and Joan Olmos, “Using COST IC1004 Vienna scenario to test C-RAN optimization algorithms,” in COST IRACON, Dublin, Ireland, Jan. 2019. **Status: Presented** [5].

2. Rolando Guerra-Gómez, Silvia Ruiz, M. García-Lozano, and Joan Olmos, “A weighted-sum multi-objective optimization for dynamic resource allocation with QoS constraints in realistic C-RAN,” in COST IRACON, Oulu, Finland, May 2019. **Status: Presented** [6].

3. Rolando Guerra-Gómez, Silvia Ruiz, M. García-Lozano, and Joan Olmos, “Predicting Required Computational Capacity in C-RAN networks by the use of different Machine Learning strategies,” in COST IRACON, Gdańsk, Poland, September 2019. **Status: Presented** [7].

4. Rolando Guerra-Gómez, Silvia Ruiz, M. García-Lozano, and Joan Olmos, “Dynamic Resource Allocation in C-RAN with Real-Time Traffic and Realistic Scenarios,” in 2019 15th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). **Status: Presented and published in IEEE proceeding** [8].

5. Rolando Guerra-Gómez, Silvia Ruiz, M. García-Lozano, and Joan Olmos, “Machine Learning Adaptive Computational Capacity Prediction for Dynamic Resource Management in C-RAN,” IEEE Access. **Status: Under Review**.

6. Rolando Guerra-Gómez, Silvia Ruiz, M. García-Lozano, and Joan Olmos, “Machine-Learning based Traffic Forecasting for Resource Management in C-RAN,” in 2020 29th European Conference on Network and Communications (EuCNC). **Status: Under Review** [8].

The remainder of the document is organized as follows. Chapter 2 describes the state-of-the-art, it emphasizes on the evolution of the C-RAN architectures and the resource management strategies. Chapter 3 presents a theoretical background of the considered machine learning techniques. On the other side, Chapter 4 details the characteristics of the realistic scenarios of Vienna City. Chapter 5 contains the mathematical model of the proposed algorithms. Finally, chapter 6 discusses and compares the results.

CHAPTER 2. STATE OF THE ART

2.1. Evolution of RAN architectures

The C-RAN architecture evolution through the last years has been widely described in [9]; a brief graph representation of this evolution is shown in Fig. 2.1.

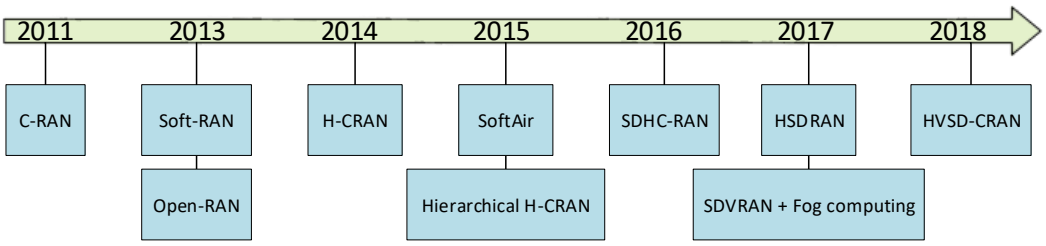


Figure 2.1 RAN architecture evolution

As stated above, the concept of C-RAN is first proposed in 2011 [1]. In 2013, Software-Defined for Radio Access Networks (Soft-RAN) and Open-RAN structures were proposed in [10] and [11], respectively. Soft-RAN is a flexible programmable architecture in which the control plane and data plane are decoupled. This structure enables a centralized management layer in a software-defined network controller entity to efficiently manage the resources. Open-RAN is an extension of the Soft-RAN, where Network Function Virtualization (NFV) is considered in the cloud infrastructure.

In 2014, the Heterogeneous Cloud Radio Access Network (H-CRAN) was defined by [12] as an alternative to overcome the fronthaul capacity limitations of C-RAN. The proposed architecture consists of an Macro Base Station (MBS) and a set of RRHs inside its coverage area. MBS is connected to the BBU pool using a backhaul link while the RRHs use the fronthaul links as Fig. 2.2 shows. The functions of the control plane are only implemented in the MBS while RRHs manage the data traffic. Consequently, H-CRAN split the control plane from the data plane to reduce the overhead through the fronthaul link, enhancing the C-RAN capabilities. In 2015, another approach to overcome the fronthaul capacity limitation (function splitting) was defined, which is splitting the baseband processing tasks between RRHs and BBU pools. This method can overcome the additional transmission delay of the fronthaul link, especially where the distance between RRH and cloud center is significant. However, the disadvantage of this solution is financial cost increment since each RRH should have baseband processing capabilities, also called Remote Radio System (RRS).

Two different architectures were proposed in 2015. Firstly, a Hierarchical H-CRAN structure was proposed in [14]. This strategy combines both approaches to overcome the fronthaul capacity limitation: H-CRAN and function splitting. There is a control MBS, and the RRHs have function splitting capability, being able to process part of the required baseband functionalities. Although the fronthaul limitation is addressed, the over-

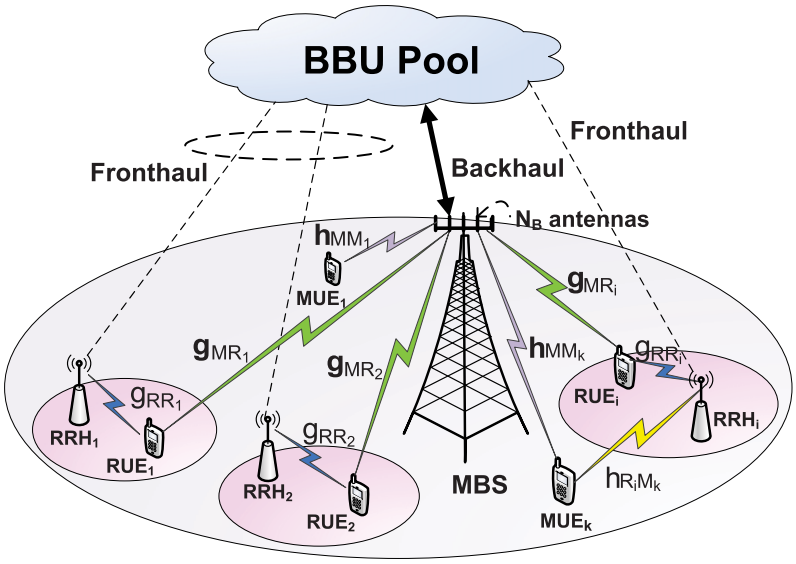


Figure 2.2 Heterogeneous C-RAN architecture [13]

all C-RAN advantages cannot be achieved, which is critical in high-density scenarios. Secondly, SoftAir is another RAN structure that was defined in 2015. The software approach is used in both RAN and the core network of communication systems. A fexible and programmable platform for the software-defined RAN (Flex-RAN) is proposed in [15].

Authors in [16] proposed an Software-defined Hyper-Cellular C-RAN (SDHC-CRAN) in 2016, which is a cloud-based software-defined RAN with physical decoupling and the ability to turn off a set of RRHs during low traffic hours. The concept of fog in C-RAN called Fog Radio Access Networks (F-RAN) is proposed in [17]. RRHs are equipped with caching capability to decrease the latency of popular services.

In 2017, another RAN structure was defined in [18]. Fig. 2.3 shows this Hierarchical Software-Defined RAN (HSD-RAN) architecture, instead of virtualizing all BSs in a single centralized controller as in software-defined RAN; multiple clusters are formed concerning the BS geographic locations, with each being assigned a virtual local controller. The connections between the groups and their associated local controllers are established via the capacity-limited fronthaul links. A virtual high-level controller is responsible for coordinating control plane decisions among the local controllers [18]. The management is split between the local controller and RRHs. This RAN is not suitable for dense regions due to the high financial cost of RRHs with the processing ability [9]. Moreover, authors in [19] proposed an integrated architecture for Software-Defined and Virtualized RAN (SDVRAN) with fog computing.

Recently, authors in [9] proposed a density-aware C-RAN design, called Heterogeneous Virtualized Software-Defined C-RAN (HVSD-CRAN) for 5G systems. The proposed architecture is able to manage two different scenarios in terms of network density: high-density and low-density modes. Fig. 2.4 shows the proposed architecture where the radio access layer is split into two parts, depending on the operation mode. The low-

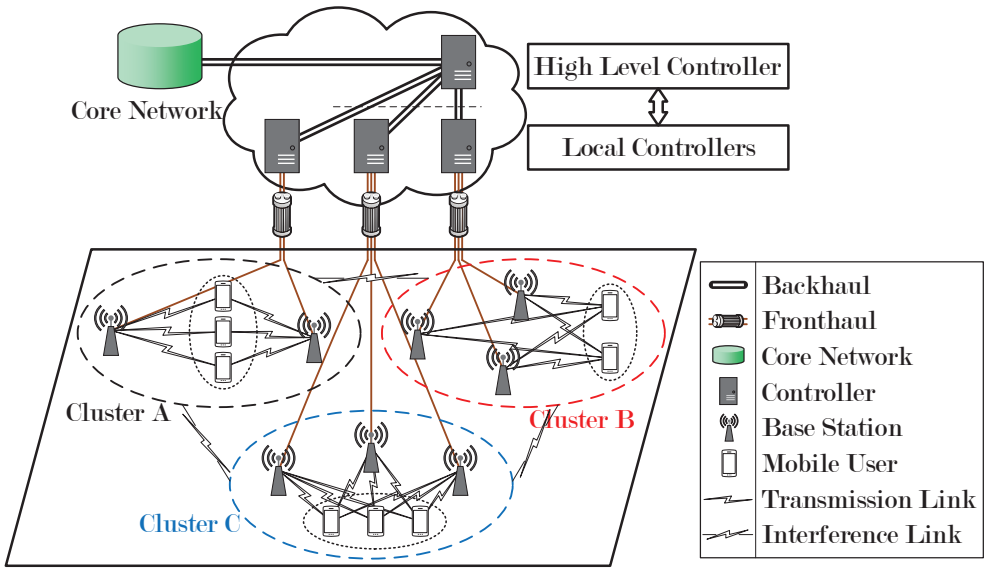


Figure 2.3 Hierarchical software-defined RAN architecture [18]

density mode consists of the deployment of RRSs that manage the UE data/control signals. It is a suitable strategy because the distance among RRSs is long, and the number of RRSs is low in these scenarios. The C-RAN advantages are not fully exploited; however, it is not a critical aspect due to low-density scenarios are not highly demanding.

On the contrary, high-density mode scenarios demand all the advantages of fully centralized architecture. For this reason, the radio access layer of this mode is implemented by an H-CRAN strategy where the control messages are sent through a coverage layer (MBS) and data message through a traffic layer with caching capability. The BBU cloud layer consists of a set of BBU processing servers and a virtualized layer where a slicing controller manages the slicing resource allocation. Core and application layers complete the structure.

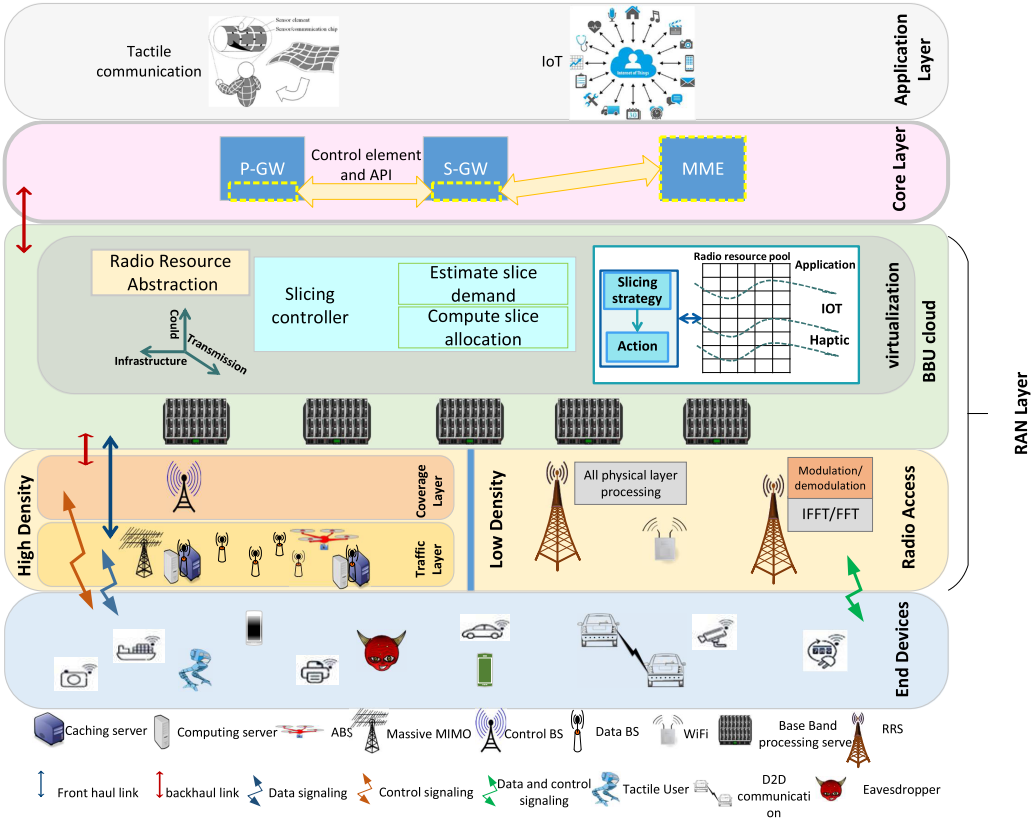


Figure 2.4 HVSD-CRAN architecture [9]

2.2. Resource management in C-RAN

To face the fluctuating traffic between day and night, weekdays and weekends, residential, commercial, and mixed areas, dynamic resource allocation algorithms have been proposed in many research works [9, 20–26]. However, resource allocation in C-RAN faces many challenges that need attention because resource management strategies for wireless communications are complex to design and implement. User mobility, radio channel variations, coverage, interference, frequency reuse, power control mechanism, and QoS requirements are some of the most critical factors that contribute to the complexity of wireless systems in terms of resource allocation. Furthermore, the unpredictable nature of wireless systems adds more challenges. For these reasons, optimized solutions for resource allocation are required to ensure adequate resource utilization and the performance of the 5G wireless networks.

Authors in [20] survey the literature on clustering algorithms applied to C-RAN architectures, evaluate the resulting configuration of BBU pools, and present different techniques for RRH clusterings such as multi-objective optimization clustering and bin-packing approach. Those clustering techniques have similar performances. The authors conclude that clustering can enhance the performance of the network. However, it is space for more analysis to select the best technique depending on the metric to optimize.

Before 5G, the resource allocation strategies of wireless systems often use one perfor-

mance metric as an objective function. However, the apparition services with different QoS requirements in 5G imposes the design of more flexible resource allocation strategies accounting for various features such as throughput, latency, power consumption, and load balancing. The resource allocation problem becomes multi-objective (Multi-Objective Resource Allocation (MORA)). In [9] have been recently proposed an adaptive architecture for C-RAN with two operation modes according to the average user density: High and Low-density modes that will coexist in real 5G networks. A MORA is presented, where data rate and power consumption are optimized in the high-density mode while total cost and delay become the objective functions in the Low-density mode. In high-density mode when there are many low-cost RRHs without baseband processing capability, resource allocation strategy is implemented in a centralized architecture. However, a small number of RRHs with baseband processing capability are deployed in the low-density mode, where a distributed resource allocation strategy is proposed to reduce latency and cost.

A multi-objective optimization problem for RRH clustering that minimizes the network transmission delay and power consumption is defined in [21]. RRHs are organized in disjoint clusters to reduce the number of active BBUs without reducing the QoS. Weighted-sum and e-constraint method are used to formulate the problem.

Letter [23] addresses the problem of maximizing the total throughput of the network via joint user association and power allocation in H-CRAN, accounting for QoS requirements. A generalized Stackelberg game approach was applied to this problem. A combination of centralized and distributed techniques was designed to achieve the solution. Notably, the user association problem is solved using a centralized strategy, and the power allocation scheme is implemented in a distributed manner.

A framework to optimize user association, radio resource allocation and power allocation in H-CRAN is also proposed by [27]. In this case, the optimization problem is formulated to maximize the overall rate while considering RRHs constraints, interference threshold for macro RRHs associated devices, and QoS constraints. A matching game approach is used to solve the formulated problem and a Lagrange dual-decomposition strategy optimizes the transmitter power. The authors in [28] have recently formulated a joint user association and resource allocation problem in the downlink of a fog network to provide a better QoS to IoT devices. They take into account the demand of QoS imposed by ultra-Reliable Low Latency Communication (uRLLC) and enhanced Mobile Broadband (eMBB) services. A matching game approach is also used to initiate a stable association between IoT users and Fog infrastructure.

A hybrid approach for RRH clustering based on game theory is presented in [24]. In this paper, the authors address the BBU-RRH association problem in a decentralized manner to reduce power consumption while the adequate number of active BBUs is calculated using a centralized strategy. Two different algorithms were implemented to solve the game among RRHs. The first relies on the best response algorithm, and the second is based on a reinforcement learning method. The results show close performance to the fully centralized approach. Using a similar hybrid strategy, authors in [25] proposed an RRH clustering scheme that jointly optimizes the power consumption and the re-association rate of the UEs that are performing handover. In this scheme, a non-

cooperated game is used to solve the problem.

Authors in [26] have proposed two different strategies for RRH clustering. Firstly, a centralized approach where a coalitional game is formulated. However, as this process is an exhaustive search that explores all possible solutions, it is intractable for high-density scenarios. For this reason, a distributed heuristic approach was proposed based on a merge and split algorithm adapted from image processing theory. The algorithm consists of two actions: coalitions are merged. Similarly, coalitions are divided if the sum of the utility of each resulting part is higher than the utility of the joint coalition.

2.3. Machine Learning in C-RAN

Machine and deep learning techniques have been widely used in many research fields. Primarily, they have been used in many tasks to the performance of mobile communications such as traffic classification, traffic load management, and cluster formation [29–38].

In [31], multitasks learning architecture using deep learning was presented. Authors use a big dataset of Telecom Italia to forecast minimum, average, and maximum traffic loads employing a practical multitask learning (MTL) approach. Different deep learning models were tested, such as Recurrent Neural Network (RNN), 3D-Convolutional Neural Network (CNN) and a combination of RNN and CNN. Results show that RNN-CNN can extract geographical and temporal traffic features.

Authors in [33] propose a centralized resource allocation scheme using online learning, which addresses interference mitigation, maximizing energy efficiency while maintaining QoS requirements challenge in H-CRAN for 5G networks. Resource block (RB) and transmission power are allocated subjected to inter-tier interference and capacity constraints. The resource allocation is performed at a dedicated controller integrated with the BBU pool and the MBS act as brokers between the controller and the RRHs for control exchange. The considered online learning model was a stochastic approximation method that solves the Bellman's optimality equation associated with the discrete-time markovian decision process.

In [36], authors use a Random Forests algorithm to design of a learning-based resource allocation scheme for 5G systems. The algorithm acts as a multi-class classifier to predict the modulation and coding scheme of a terminal at any given position served by the C-RAN. One of the aims is to reduce the signal overhead in the network. Results show that due to the reduction in signaling, the proposed algorithm has better performance in high user density scenarios than Channel State Information (CSI) schemes.

A reinforcement learning-based resource allocation strategy is proposed in [37]. The algorithm consists of two stages. First, to predict the user position, a neural network model called Long Short-Term Memory (LSTM), which is a kind of RNN, is proposed. Consequently, a reinforcement learning strategy based on the mobility pattern previously estimated is used to maximize the network throughput.

2.4. Challenges and open issues

As C-RAN has been identified as enabling technology for 5G systems, it must address the radical evolution in flexibility, security, and performance to support uRLLC, eMBB, and Massive Machine Type Communications (mMTC) services. Latency, throughput, resource allocation, handover, energy efficiency, power consumption, and cost-saving are parameters that must be enhanced. This improvement demands efforts from the research community and the combination of some of the most promising technologies as Software-Defined Network (SDN) and NFV. This complexity is a big challenge by itself. In this section, a description of the leading open issues, the research community is facing, are summarized.

2.4.1. Huge fronthaul capacities needed

Fronthaul links between BBUs and RRUs must have high bandwidth capability with low delay and cost requirements. The fully centralized architecture demands the highest fronthaul bandwidths due to the signal is completely processed at the BBU pool resulting in considerable overhead. Different functionality split options have been defined by [39] to reduce the fronthaul bandwidth requirements. However, the potential of C-RAN, depending on the number of centralized functionalities, is reduced; [40] carries out a detailed analysis of this situation. For this reason, this leaves space for improvement.

2.4.2. RRH clustering (BBU-RRH mapping)

Designing real-time RRH clustering, also called BBU-RRH mapping methods with efficient BBU coordination algorithms with minimal overhead that optimize parameters or strategies as load balancing, multiplexing gain, inter-cell interference, throughput using CoMP, handover frequency, energy efficiency or power consumption is a real challenge. Many authors are dedicating efforts to overcome this challenge [20, 21, 25, 41–43].

2.4.3. Security and management of network slicing

Another significant challenge in C-RAN is the security in terms of user privacy and isolation between slices. As resources are shared between BBUs, breaking user privacy, and accessing secured data is a possibility. Besides, as C-RAN has to support services that are provided by different Mobile Virtual Network Operator (MVNO) using network slicing over the same infrastructure, isolation among slices is a vital challenge. The definition of resource management strategies to overcome these issues has been the aim of many researchers [9, 44–46]. Hence, providing reliable, cost-effective, and quality of service guaranteed network slices under C-RAN architecture is one of the significant challenges in 5G.

2.4.4. Energy efficiency, power consumption, and cost-saving

Increasing the energy efficiency of the mobile communication systems while the cost is reduced has been a relevant research field in recent years. The integration of different technologies (e.g., SDN, NFV, Mobile Edge Computing (MEC)) to build the 5G networks creates a new challenge: How to manage the high flexibility and capacity demanded by the system while energy efficiency, power consumption, and cost are enhanced. Many authors have proposed Green C-RAN deployments to address this challenge [3, 47–51]. For instance, MEC and caching, energy-efficient designs, multi-dimensional resource management, and physical layer security have been identified as an open issue.

For energy efficiency, power consumption, and cost-saving; authors in [47] have been identified and explained the following aspects as open issues:

- Energy-Efficient Joint H-CRAN and Edge Computing Deployment
- Energy-Aware Revenue Maximization
- Cost and Energy-Aware Cell Site Selection in Hybrid Power Supplied Deployment
- Energy-Efficient Data-Oriented Design

2.4.5. Resource management

The required density of RRHs to provide high data rates incurs high computational complexity due to the enormous amounts of data related to signal processing, resource allocation, and RRHs/BBUs coordination. This complexity is a big challenge facing the establishment of scalable networks. Resource allocation strategies, which determines distribution of the computation resources, fronthaul capacity, radio spectrum, and power allocation, is still a challenge. Some of the works that are related to this challenge have been presented in [23, 27, 47, 48, 52].

One of the fundamental challenges is how to assign the isolated resources efficiently to the different virtual operators. Resources allocation can be based on multiple criteria, e.g., bandwidth, data rate, power, interference, pre-defined contracts, channel conditions, traffic load, or a combination of these parameters. Coordination and communication protocols have to be well designed [53].

The introduction of adaptive machine learning techniques to achieve a proactive network capable of adapting to data demands (e.g., IoT demands) that fluctuates over time and places while optimizes the available resources is a significant challenge [31, 33, 38, 47, 54]. Due to that, Infrastructure Providers (InPs) rent the maximum peak of capacity demanded by each service provider or mobile network operator regardless of the required instantaneous capacity.

2.5. Partial conclusions

This chapter summarizes the leading papers in the context of resource allocation strategies in C-RAN. Most of the resource management techniques in mobile networks, such as game theory and machine learning combine with the design of a flexible architecture based on SDN and NFV approaches, enable dynamic resource allocation strategies for 5G systems. The presented study of the state-of-the-art shows that there is space for more research in the context of C-RAN for 5G. Notably, the definition of real-time resource management and the introduction of machine learning techniques to achieve proactive networks capable of adapting to high fluctuation on data traffic has been identified as an open issue.

CHAPTER 3. BACKGROUND ON ML TECHNIQUES

3.1. Support Vector Machine

Support Vector Machine (SVM) theory was first proposed in [55]; since then, it has been widely used in classification and regression tasks of different scientific and engineering fields. The original idea focuses on element classification. Let us assume a simple case to illustrate how it works. Fig. 3.1 shows a set of training samples that belong to two classes (circles and squares).

The aim is to find the best hyper-plane (dotted line in Fig. 3.1) that allows classifying the data. The algorithm uses optimization theory to maximize the width of the street. Let us assume that ω is a vector perpendicular to the hyper-plane, and u is a vector that points to an unknown observation. The decision rule used to decide if u belongs to the circle class is presented in (3.1)

$$u \cdot \omega + b \geq 0 \tag{3.1}$$

It means that if the projection of u onto the perpendicular line of the hyper-plane is greater than the distance from the origin to the hyper-plane then the sample is on the right side (circle), where $b \in \mathbb{R}$ and $\cdot$ is the scalar product. However, as the idea is to find the line that maximizes the width of the street, let take the square ($x^{\mathrm{s}} \in \mathcal{S}$) and circle examples ($x^{\mathrm{c}} \in O$) into (3.2) and (3.3) respectively, which guarantee that all the samples on the dataset are out of the street. $\mathcal{S}$ and O represent the set of squares and circles, respectively.

$$\omega \cdot x^{\mathrm{s}} + b \leq -1 \tag{3.2}$$

$$\omega \cdot x^{\mathrm{c}} + b \geq 1 \tag{3.3}$$

Equations (3.2) and (3.3) are joint into (3.4) introducing the variable y_i

$$y_i(\omega \cdot x_i + b) \geq 1 \tag{3.4a}$$

$$y_i = \begin{cases} +1, & x_i \in O \\ -1, & x_i \in \mathcal{S}, \end{cases} \tag{3.4b}$$

where x_i represents the vector of the i^{th} training sample. It is possible to compute the width of the street ($\mathbb{W}$) by taking one example per class over each boundary due to they hold the equality condition in (3.4a), the result is shown in (3.5)

$$\mathbb{W} = (x^{\mathrm{c}} - x^{\mathrm{s}}) \cdot \frac{\omega}{\|\omega\|} = \frac{2}{\|\omega\|} \tag{3.5}$$

where $\|\cdot\|$ denotes the Euclidean norm. SVM aims to maximize (3.5) subject to (3.4) to obtain the best hyper-plane for classifying data. After solving this optimization problem using a Lagrangian function, it is possible to realize that the solution depends only on the samples, and vector ω is a linear combination of those samples [55].

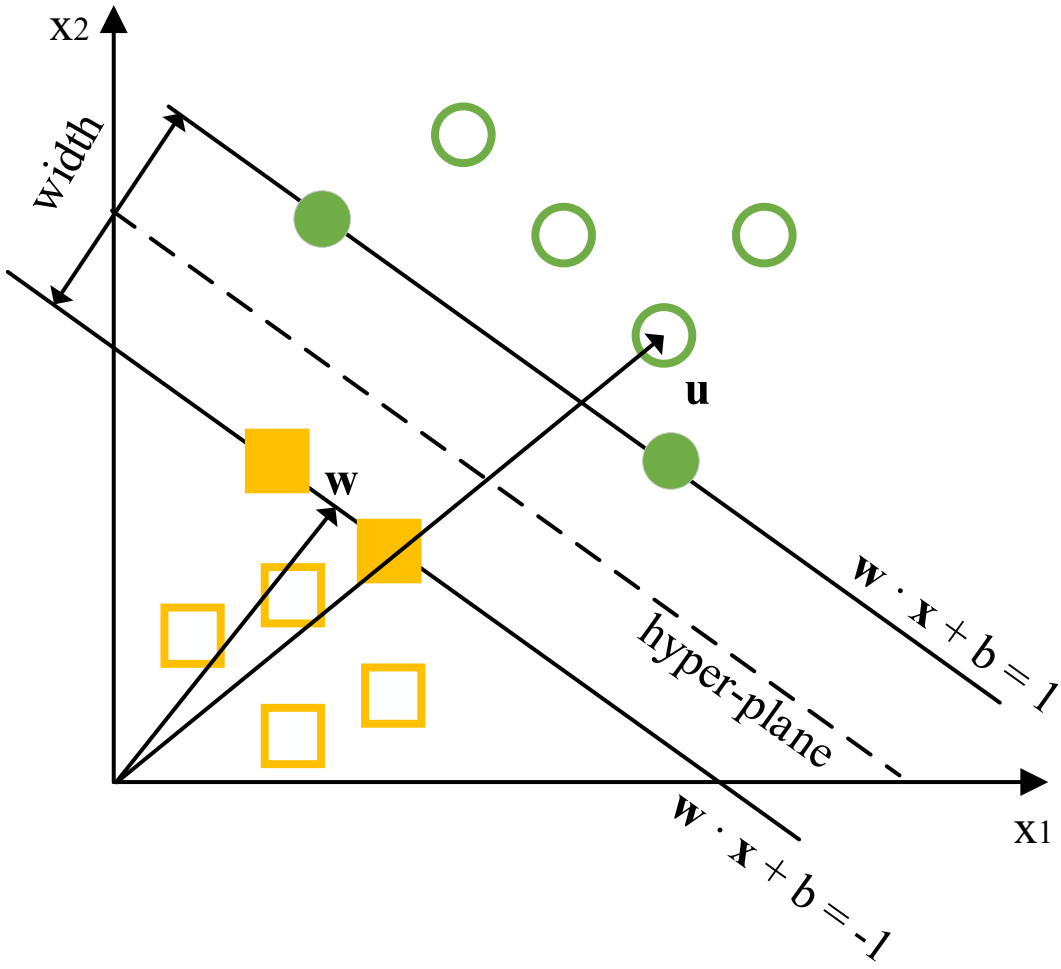


Figure 3.1 Basic classification example of SVM.

This strategy was also extended to address regression tasks in [56]. In this case, the idea is to find a linear function $f(x) = x \cdot \omega + b$ that fits the training data. The optimization problem is formulated to minimize the different (error) between the predicted value extracted from $f(x)$ and the real observation of the regression. The mathematical process is detailed in [56]. Equation (3.6) shows the fitting function

$$f(x) = \sum_{i=1}^{N_t} \alpha_i x_i \cdot x + b \tag{3.6}$$

Where a_i and b are real values obtained after the training process where the optimization problem is solved, N_t is the number of samples in the training dataset.

The previous analysis of SVM strategies assumes that it is possible to classify or predict data based on a linear hyper-plane or a linear fitting function. However, in many applications, linear approaches are not able to process the data. In those cases, it is not suitable to find a linear function that describes the data. A transformation (Φ) over the data plane to solve this problem is applied; this method is called kernel trick. After the transformation, it is possible to use a linear approach in a higher-order space to fit or

classify data. The fitting function after applying the kernel is shown in (3.7).

$$f(x) = \sum_{i=1}^{N_t} \alpha_i K(x_i, x) + b \tag{3.7}$$

where $K(x_i, x) = \Phi(x_i)\Phi(x)$ depicts the kernel function.

3.2. Time-Delay Neural Network

Artificial Neural Networks (NNs) have been widely used during the last years to solve different machine-learning problems, even regression and time series forecasting tasks. Time Delay Neural Network (TDNN) is a combination of typical NN architecture and an input layer that reshapes sequence time series data into parallel (shift register), employing a set of delays (N) to use the previous time steps as features of the NN. The learning process takes place in the hidden layers of the neural network. Fig. 3.2 shows the TDNNs structure, as well as the basic block diagram of a neural network entity (also called neuron). Equation (3.8) shows the behavior of a single neuron. The inputs are multiplied by the weights (W), and a bias (b) is added before applying the activation function (f_a) to compute the output of the neuron. The knowledge is in the weights and bias of each neuron in the hidden layers.

$$N_o = f_a(W \cdot X + b) \tag{3.8}$$

where X is the input vector, and N_o is the output of the neuron.

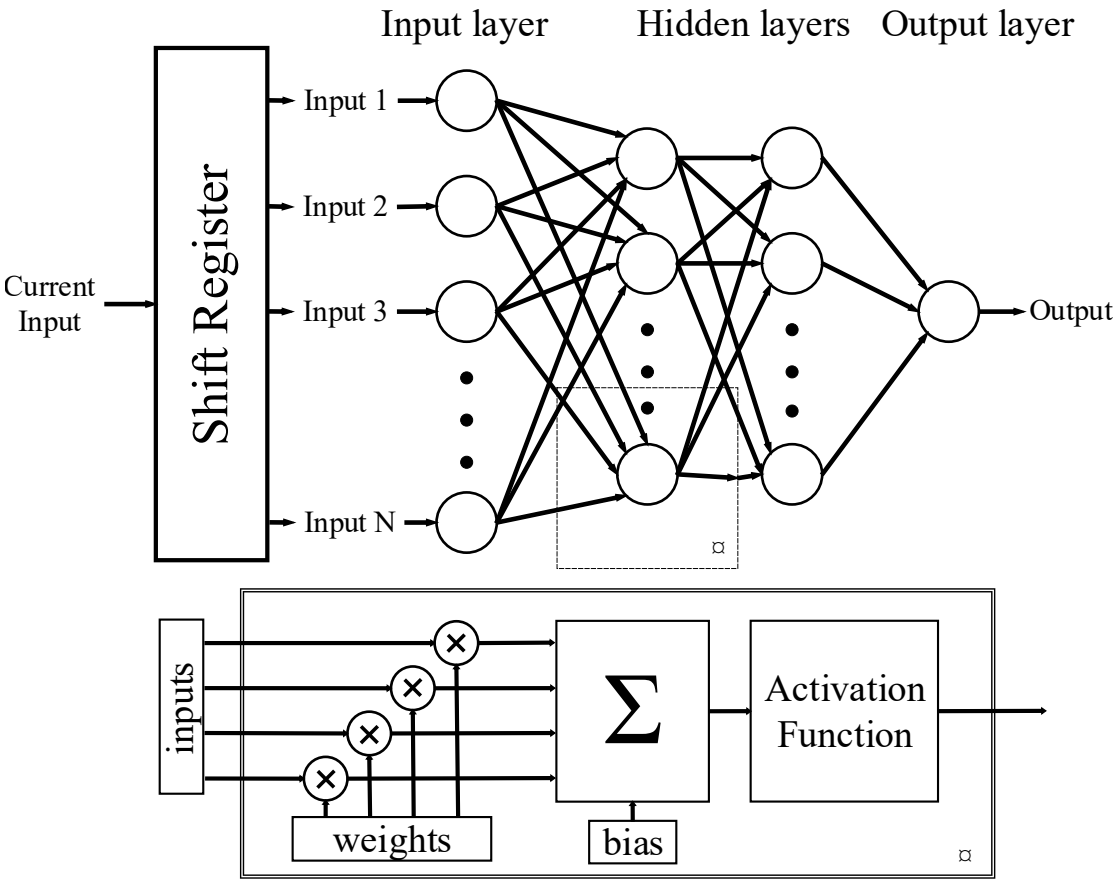


Figure 3.2 General scheme of a time-delay neural network for time series forecasting with N previous time instants.

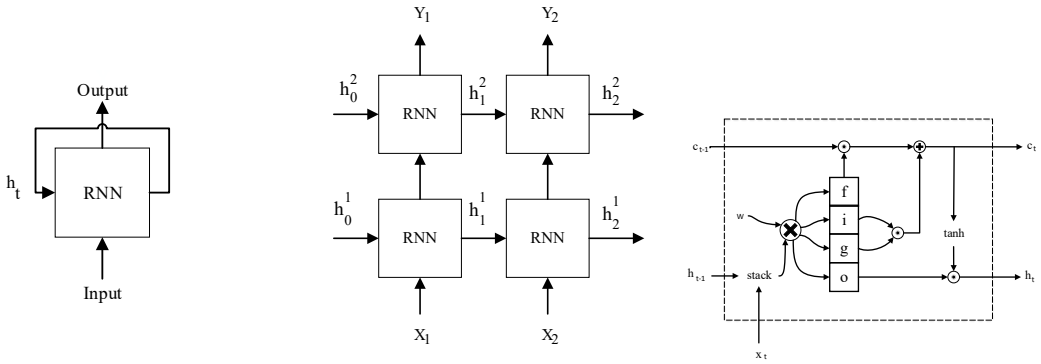
3.3. Long Short-Term Memory

Traditional NNs have outstanding prediction performance when based on the status of the input variables. However, they are not able to remember sequential data. RNNs try to address this issue using a feedback loop to create a hidden state where the information of previous time steps is stored. It means that RNNs predict the next output based on the current input and the hidden state. Fig 3.3(a) shows a basic structure of a recurrent neural network unit.

The hidden state of the RNN is upgraded recursively, using the same approach of a neural network (see (3.8)) but considering the previous hidden state (h_{t-1}) as another input. Equation (3.9) shows the process to upgrade the hidden state (h_t)

$$h_t = f_{\mathrm{a}}(W \cdot [h_{t-1}, X_t] + b) \tag{3.9}$$

where W is the weight vector, f_{a} the activation function and $[h_{t-1}, X_t]$ denotes the concatenation or stack operation between the previous hidden state and the current input, respectively. The scheme of an RNN shown in Fig 3.3(a) could be unrolled to create deeper designs as the multilayer RNN in Fig. 3.3(b), where the hidden states of the first layer are inputs of the second layer.



(a) basic scheme of a RNN unit. (b) multilayer architecture. (c) diagram of a LSTM entity.

Figure 3.3 General deep learning architecture with LSTM cells

Those architectures face the vanishing gradient problem that was solved by [57], defining a different kind of RNN called LSTM. Moreover, LSTM improves long-term predictions.

The structure of an LSTM entity is shown in Fig 3.3(c). The critical aspect of an LSTM unit is the cell state that has been denoted by c_t. LSTM units could remove and aggregate information to the cell state. Those processes are regulated by entities called gates that are a combination of a neural network and a pointwise multiplication; it controls the amounts of information at the output of the gate. The output of the neural network of each cell is often obtained using a sigmoid activation function, which allows quantifying the portion of the information that could pass through the gate with a coefficient from zero to one. As the output is a pointwise product, zero means no signal to the output, and one means the whole signal remains in the output.

First, the forget (f) gate decides what information to remove from the cell state. Consequently, the input (i) and gate (g) gates decide what information aggregate to the cell state. Finally, the output gate (o) decides what information go to the output. The whole process of the LSTM is summarized in (3.10)

$$f_t = \sigma(W_{\mathrm{f}} \cdot [h_{t-1}, X_t] + b_{\mathrm{f}}) \tag{3.10a}$$
$$i_t = \sigma(W_{\mathrm{i}} \cdot [h_{t-1}, X_t] + b_{\mathrm{i}}) \tag{3.10b}$$
$$o_t = \sigma(W_{\mathrm{o}} \cdot [h_{t-1}, X_t] + b_{\mathrm{o}}) \tag{3.10c}$$
$$g_t = \tanh(W_{\mathrm{g}} \cdot [h_{t-1}, X_t] + b_{\mathrm{g}}) \tag{3.10d}$$
$$c_t = f_t * c_{t-1} + i_t * g_t \tag{3.10e}$$
$$h_t = o_t * \tanh c_t, \tag{3.10f}$$

where W_{k} and b_{k} are the weights and the bias of the neural network in gate k, respectively. The activation functions of the gates are σ or $\tanh$ that represent the sigmoid and hyperbolic tangent functions, respectively; $*$ operation denotes the pointwise product.

CHAPTER 4. SCENARIO DESCRIPTION

To analyze the performance of the proposals, two realistic scenarios have been defined: a large scale scenario with a per hour daily traffic (see Fig.4.2), and a small case dense scenario where individual data traffic on a per-service basis is considered. The whole research of this thesis is based on these scenarios. Their details are summarized in this section and they were also presented in [5, 8], respectively.

4.1. Scenario 1: Large-scale C-RAN

COST IC1004 agreed on the definition of the realistic Vienna scenario that has been widely used by researchers as a common platform to compare the performance of radio resource optimization algorithms. The scenario has been modified to include C-RAN deployment as well as dense small cell deployment in the city center. The scenario covers an area of 455 km^2 with a perimeter of 86 km. The blue and green points over the map in Fig. 4.1, represent the MBSs and possible BBU pools coordinates, while the red points are the Small Base Stations (SBSs) [58, 59].

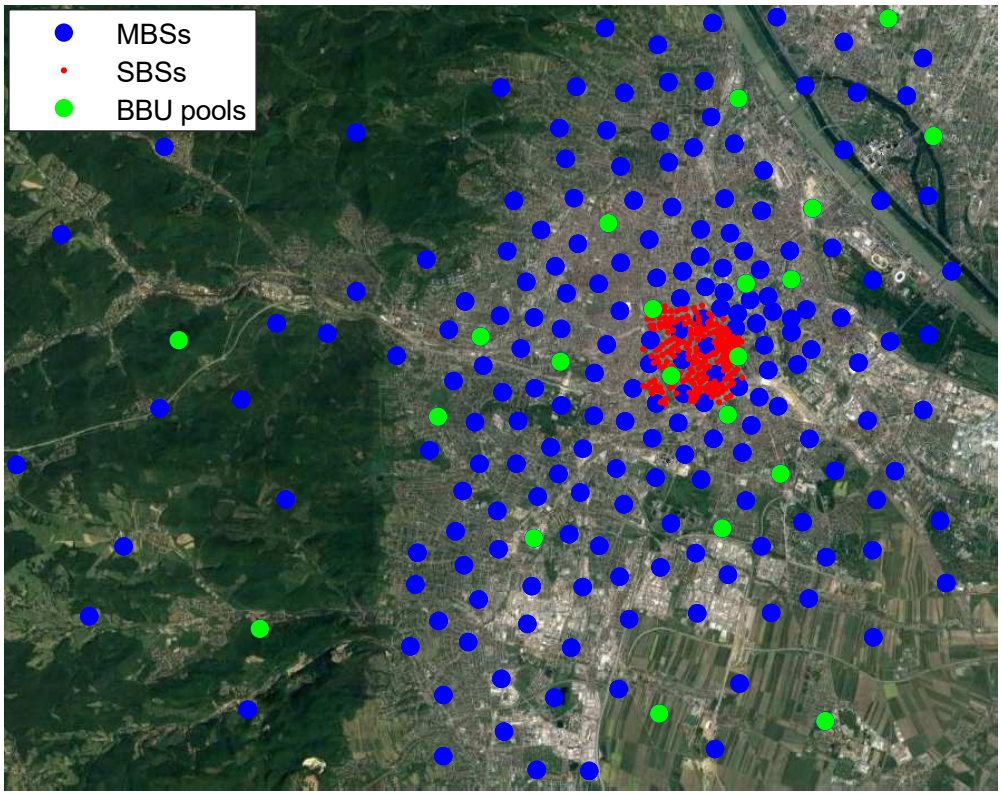


Figure 4.1 Scenario 1: Vienna city map.

MBSs are sectorized in 3 or 2 cells. As a result, the scenario includes 628 MBSs allocated in 233 sites and 221 SBSs, which represent a total of 849 RRHs and 21 BBU pools (see Table 4.1).

Mobile networks face dynamic environments with high mobility and load fluctuations.

Table 4.1 Main parameters of the scenario 1

Parameters	Scenario 1
Dimension	455 km^2
Sites	444
MBSs (sites)	628(233)
SBSs (sites)	221(221)
BBU pools	21
RRHs	849

Nowadays, operators allocate to each cell the resources needed to manage the peak traffic per day, causing inefficient use of the allocated resources. On the contrary, C-RAN architecture centralizes resources in a BBU pool gaining in flexibility to address the tidal effect; aggregating traffic from different types of RRHs, a multiplexing gain is obtained.

To test this scenario three different types of cells have been considered in terms of traffic profiles (office, residential and mixed). The traffic profiles are modeled by multiple Gaussian functions. By controlling the mean and deviation of the Gaussian functions we can adapt to realistic traffic profiles. The data traffic profiles used are shown in Fig. 4.2.

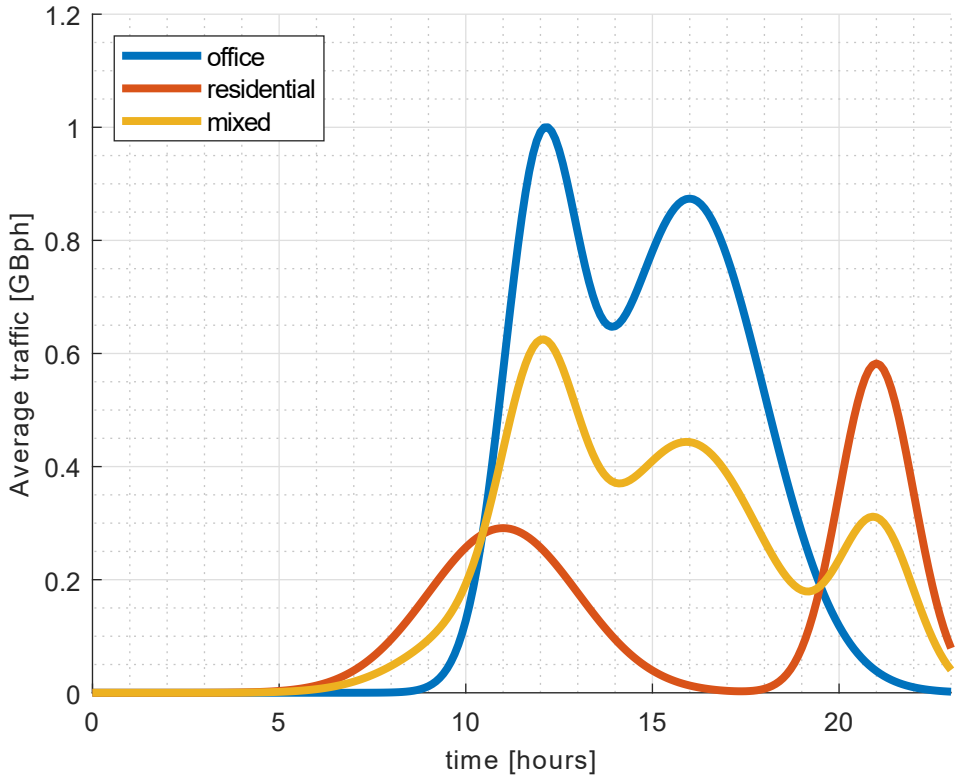


Figure 4.2 Realistic traffic profile for office, residential and mixed cells.

Figure 4.3 C-RAN deployment over Vienna city downtown. Green points represent BBU pools location while blue and red marks are MBSs and SBSs (RRHs), respectively.

4.2. Scenario 2: Small-scale C-RAN

Fig. 4.3 shows the location of the cells over the small-scale scenario. Blue and green points depict MBSs, sectorized in three or two cells, and green points also represent BBU pools coordinate. The location of BBU pools matches with macro site coordinates where there are more infrastructure and resources. Red points are SBSs that are installed in street corners to boost line-of-sight connections.

This scenario was first defined in [60], and it has been widely used as a common platform to test the performance of different radio resource optimization algorithms [59, 61]. The protocol stack and the processing capacity of each base station on the scenario are split using option 8, defined by [39], so RRHs are only responsible for transmitting/receiving the in-phase and quadrature components of the signal to/from the BBU pool, being the remaining functionalities centralized at the BBU pools.

Table 4.2 summarizes the features of the scenario 2, where path-loss is computed by a 3D ray-tracing tool as in [60]. RRHs to BBU pool association (fronthaul links) is done by minimizing the delay, being the advantages and inconveniences of this assumption discussed in [8].

After defining the scenario, realistic UEs, services, and consequently, traffic have been modeled, accounting for QoS constraints and service priorities. Each UE is connected to

Table 4.2 Main parameters of the scenario 2.

Parameters	Value
Area (km^2)	25
Sites	228
MBS (sites)	51(17)
SBS (sites)	221(211)
BBU pools	3
RRHs	272
Power (dBm)	(43,24)*
Quantization resolution (bit)	(24,16)*
RRH antenna gain (dB)	(18,10)*
Bandwidth (MHz)	20
Number of RBs	100
Total UEs	7000
UEs antenna gain (dB)	0
Noise + Interference (dBm)	-97

* The format of the data is (MBSs,SBSs)

the RRH that maximizes the Signal-to-Noise-plus-Interference-Ratio (SINR), estimated through (4.1):

$$\text{SINR} = P_{\text{RRH}} + G_{\text{RRH}} + G_{\text{UE}} - L - 10\log(N+I), \tag{4.1}$$

where P_{RRH} is the power transmitted by the RRH, G_{RRH}, and G_{UE} are the RRH and UE antenna gains respectively, L is the path-loss from the RRH to the UE, and N and I are the UE noise and interference power respectively.

Conversational, streaming, and interactive services have been generated based on a packet level model used in [3, 49, 62] and summarized in Table 4.3.

Table 4.3 Service parameters.

Services	w_{s}	Size	Time interval	Duration (s)	Traffic mix (%)
VoIP	83	packet: 40 B	20 ms	Exp(120)	25
video	59	packet: [20-250] B	100 ms	Exp(300)	25
Web	36	mean page: 315 kB	Exp(30)	Exp(400)	30
FTP	36	mean file: 2 MB	Exp(180)	–	20

The traffic mix parameter describes the percentage of active sessions per service and RRH. The w_{s} weight column defines the service priority that is used by the scheduler and the DRM to allocate the computational resources to each RRH, trying to guarantee

the QoS. Higher priority has been assigned to Voice over IP (VoIP) and video flows due to their delay restrictions. Session duration follows an exponential distribution, except for File Transfer Protocol (FTP) services, where total duration depends on the size of the packet to be transmitted and the UE throughput [62]. The time interval between consecutive packets is fixed on 20 ms and 100 ms for VoIP and video streaming services, respectively, and follows an exponential distribution for non-real time services.

To calculate the required computational resources, the Modulation and Coding Scheme (MCS), as well as the number of RBs needed to transmit a packet should be known. The mapping between MCS and SINR is summarized in Table 4.4 and has been obtained using [63], which presents a link-abstraction model based on mutual information at the modulation symbol level. The number of RBs required to transmit a packet is extracted from [64].

Table 4.4 Mapping between SINR and MCS.

SINR [dB]	Modulation order (M)	code rate (ρ)
< -5	QPSK (2)	0.076
$[-5, 1]$	QPSK (2)	0.3
$[1, 3.1]$	QPSK (2)	0.44
$[3.1, 6.1]$	QPSK (2)	0.59
$[6.1, 9]$	16QAM (4)	0.48
$[9, 13]$	16QAM (4)	0.6
$[13, 16]$	64QAM (6)	0.65
> 16	64QAM (6)	0.85

4.2.1. Resource Demand Estimation

The Required Computational Capacity (RCC) is defined as the minimum amount of computational operations necessary to implement physical layer functions at the BBU pool, such as channel coding, modulation, MIMO precoding, and Orthogonal Frequency-Division Multiplexing (OFDM) symbol mapping. The RCC is calculated based on the strategy proposed by [65] and modified by [62] to introduce parallel processing. The strategy uses a LTE reference scenario, where the RCC and a set of scaling factors that describe how the RCC evolves to other scenarios are tabulated. Those scaling factors depend on the network parameters and the physical function to be implemented. Equation (4.2) describes this method.

$$\begin{aligned} C &= \sum_{i \varepsilon I} C_i^{\mathrm{ref}} \prod_{x \varepsilon \mathcal{X}} \left(\frac{x_{\mathrm{act}}}{x_{\mathrm{ref}}} \right)^{s_{i,x}} \\ \mathcal{X} &= \{B_{\mathrm{w}}, N_{\mathrm{a}}, Q, M, \rho, N_{\mathrm{s}}\}, \end{aligned} \tag{4.2}$$

where C represents the RCC of the desired scenario, C_i^{ref} is the processing capacity needed to address the function i in the reference scenario in Giga operations per second (GOPS). Subscripts act and ref depict actual scenario and reference scenario

Table 4.5 Scaling factors ($s_{i,x}$) for function i and RCC of the reference scenario (C_i^{ref}) (based on [62, 65]).

Function index i	$C_{i,ref}$	B_{w}	N_{a}	Q	M	ρ	N_{s}
OFDM modulation (CF)	1.3	1	1	1.2	-	-	-
OFDM demodulation (CF)	2.7	1	1	1.2	-	-	-
MIMO precoding (UF)	1.3	1	1	1.2	0	0	1
MIMO decoding (UF)	5.3	1	2	1.2	0	0	0
Modulation (UF)	1.3	1	0	1.2	1.5	1.5	1
Demodulation (UF)	2.7	1	0	1.2	1.5	1.5	1
Channel coding (UF)	1.3	1	0	1.2	1	1	1
Channel decoding (UF)	8	1	0	1.2	1	1	1

respectively, $s_{i,x}$ is the scaling factor of the function i and parameter $x \in \mathcal{X}$. The set $\mathcal{X}$ contains the operating bandwidth (B_{w}), the number of antennas (N_{a}), the quantization resolution (Q), the modulation order (M), the code rate (ρ) and the number of streams ($N_{\text{s}} \leq N_{\text{a}}$). Finally, set I contains the PHY functionalities that has shown in Table 4.5.

As the resources are centralized at BBU pool entities and the functionalities are virtualized, it is possible to split those functions into two groups: The functions that may be implemented by user sessions, processed independently and in parallel are called user-processing functions (UFs), such as channel coding and modulation. The functions that are common to all users in the same carrier component/cell and could not be split by user sessions, such as OFDM modulation, are denoted as common-processing functions (CFs). Table 4.5 summarizes the reference computational-capacity, as well as the scaling factors of the considered PHY functions. Function indexes are the identifiers of the PHY functionalities. The total RCC of a BBU is calculated by (4.3):

$$C_{r,t} = \sum_{i=1}^{N_{\text{CF}}} C_{r,i,t}^{\text{CF}} + \sum_{u=1}^{N_{r,t}} \sum_{j=1}^{N_{\text{UF}}} C_{r,u,j,t}^{\text{UF}}, \tag{4.3}$$

where $C_{r,t}$ is the RCC to handle the RRH r at time t, $C_{r,i,t}^{\text{CF}}$ is the capacity associated with the common functions i needed to handle the RRH r at time t, and $C_{r,u,j,t}^{\text{UF}}$ is the capacity to run the UF j of the active UE u through the RRH r. N_{CF} and N_{UF} are the amount of CFs and UFs respectively, while $N_{r,t}$ is the number of active UEs in RRH r at time instant t.

CHAPTER 5. MATHEMATICAL MODEL

5.1. RRH-BBU pools association strategies

To test C-RAN performance four different RRH-BBU pools association algorithms have been considered: minimum delay, load balancing based on traffic or number of RRHs and multiplexing gain optimization. This section presents a brief mathematical description of this strategies.

5.1.1. Minimum delay (MD)

The minimum delay algorithm only takes into account the distance to establish the connections between RRHs and BBU pools. To minimize the delay, the algorithm selects for each RRH the nearest BBU pool following (5.1).

$$s_i = \{j \mid d_{ij} \leq d_{max} \cap d_{ij} = min(d_i)\} \tag{5.1}$$

where d_i is a vector that contains the distance from the RRH i to each BBU pool, d_{max} is the maximum allowed fronthaul distance, $min(\cdot)$ operator returns the minimum value and s_i is the BBU pool selected to connect to RRH i.

5.1.2. Load balancing (LB) algorithm

The load balancing algorithms can use two different metrics: the number of RRHs already assigned and the capacity handled by BBU pools. The i^{th} RRH is connected to BBU pool c following (5.2).

$$c = \{j \mid d_{ij} \leq d_{max} \cap C_j = min(C)\} \tag{5.2}$$

where C is a vector that depending on the version used contains the number of RRH connected to each BBU pool or the capacity handled per BBU pool. C_j is the capacity of the less loaded BBU pool (j) that is selected for the algorithm to establish the connections.

5.1.3. Multiplexing gain algorithm

This algorithm balances different type of traffic profiles to improve the multiplexing gain. The connections are established following two steps, described by (5.3) and (5.4). First, the algorithm connects RRH i^{th} to BBU m using (5.3) where $max(\cdot)$ denotes maximum operator.

$$\begin{aligned} m = \{j \mid d_{ij} \leq d_{max} \\ \cap MG_j < max(MG_{jn}) \\ \cap MG_j = min(MG)\} \end{aligned} \tag{5.3}$$

If $m = \emptyset$ the algorithm uses the second condition to establish the connection (5.4), where the RRH is connected to the BBU pool with the highest multiplexing gain. The algorithm repeats this process until each RRH is connected to the network.

$$m = \{ j \mid d_{ij} \leq d_{max} \cap MG_j = min(MG) \} \tag{5.4}$$

In (5.3) and (5.4) MG is a vector that contains the multiplexing gain of each BBU pool, MG_j is the multiplexing gain of the BBU pool j and MG_{jn} is a vector that stores the achievable multiplexing gain after connecting each possible RRH, computed as:

$$MG_j = \frac{\sum_{k=1}^{N_{RRH,j}} C_{RRH,k}[GBph]}{C_j[GBph]} \tag{5.5}$$

where $N_{RRH,j}$ is the number of RRHs connected to the j^{th} BBU pool, $C_{RRH,k}$ is the peak traffic through the k^{th} RRH and C_j is the traffic handled by the j^{th} BBU pool.

5.2. Dynamic resource management design

In this section, the dynamic resource allocation problem with QoS constraint is addressed. The aim is to optimize the allocated capacity at each BBU pool considering the required computational capacity, the priority of running services and the maximum capacity available at the BBU pool [8].

Let's assume that the coverage area of a specific region is served by a set of $R = \{1, \ldots, N\}$ RRHs, managed by a BBU pool. The required computational capacities to handle each RRH are $C_t = \{C_{1,t}, \ldots C_{N,t}\}$, which are computed using (4.3). The objective is to maximize the allocated computational capacity, that is described by the set $ACC_t = \{ACC_{1,t}, \ldots, ACC_{N,t}\}$

The problem could be modeled using a game-theoretical approach where RRHs are connected to BBUs that are competing for computational resources at each transmission time interval (TTI). The allocated resources must not surpass the total capacity of the BBU pool (M), as expressed in (5.6). We call this condition C_1.

$$C_1 : \sum_{i \in R} ACC_{i,t} \leq M \quad \forall t \tag{5.6}$$

The weight of each service allows to establish priorities by aggregating QoS constraints. We denote the average service weights at each RRH as $\overline{w}_t = \{\overline{w}_{1,t}, \ldots, \overline{w}_{N,t}\}$. A bargaining power is defined in (5.7), where the average service weights at each RRH act as a fitness parameter.

$$C_2 : B_{i,t} = \frac{\overline{w}_{i,t} C_{i,t}}{\sum_{j \in R} \overline{w}_{j,t} C_{j,t}} \tag{5.7}$$

BBU allocated resources must not be greater than the required computational capacity (5.8).

$$C_3 : ACC_{i,t} \leq C_{i,t} \quad \forall i \in R, \forall t \tag{5.8}$$

Then the underlying optimization problem to perform the proposed strategy is formulated as:

$$\begin{aligned} \underset{ACC_t}{\text{maximize}} \quad & \sum_{i \in R} B_{i,t} ACC_{i,t} \\ \text{subject to :} \quad & \mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3 \end{aligned} \tag{5.9}$$

The problem becomes a weighted-sum multi-objective optimization problem, where BBUs-RRHs running higher priority services are prioritized because the allocated resources are weighted by the bargaining power factors.

Problem (5.9) is solved by CVX tool [66] iteratively during the simulation period.

5.3. DRM with adaptive capacity (DRM-AC)

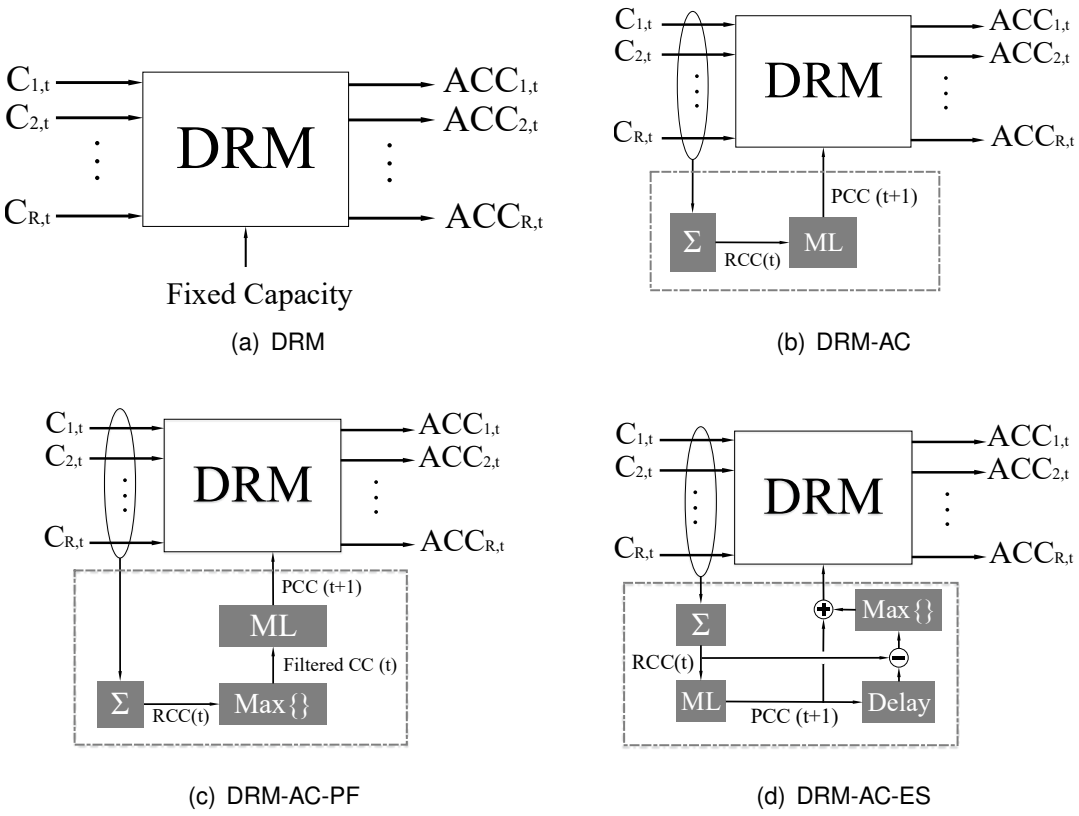
Fig. 5.1(a) shows the general scheme of the DRM, where $C_{r,t}$ depicts the required computational capacity to handle the RRH r at time t (computed using (4.3)). Moreover, ACC$_{r,t}$ represents the allocated computational capacity, where $r \in [1, \mathrm{R}]$, being R the amount of RRHs connected to the BBU pool under analysis. The DRM allocates the resources available at the BBU pool to manage each RRH with service priority as well as QoS constraints. This strategy was presented in [8]. However, the instantiated computational capacity at BBU pool is fixed (see 5.1(a)). It causes QoS degradation (under-provisioned) or inefficient resource usage (over-provisioned). To tackle this issue, we propose to dynamically instantiate resources using the schemes shown in 5.1(b), 5.1(c) and 5.1(d).

Fig. 5.1(b) shows the block diagram of the DRM-AC. A machine learning entity is introduced. Its mission is to predict the required computational resources at the BBU pools, based on the current network load, is introduced. An aggregation block computes the RCC(t) based on the current demand of each RRH ($C_{r,t}$), which depicts the database of the ML block to predict the computational capacity at the next time step PCC$(t+1)$.

However, negative errors in the prediction produce QoS degradation. Two approaches to address this issue are proposed as it will be detailed afterwards.

- Filtering the data before the training process using a sliding window method and applying the maximum operation. Fig 5.1(c) shows the general diagram of this approach that has been called DRM-AC with prefiltering (DRM-AC-PF).
- Establishing a margin amount of computational resources equal to the maximum error in a previous time window. Fig 5.1(d) depicts the block diagram to implement this strategy, which is a DRM with error shifting (DRM-AC-ES).

ML block contains the machine-learning algorithm to predict the computational capacity. On the other hand, the delay block is a memory that stores the input value for the next iteration. The $\mathrm{Max}\{\}$ block depicts a non-linear filter, it computes the maximum, sliding a window through the input data. The output of the $\mathrm{Max}\{\}$ block is equal to the maximum of the θ previous time steps.



(a) DRM (b) DRM-AC

(c) DRM-AC-PF (d) DRM-AC-ES

Figure 5.1 Block diagrams of the dynamic resource management strategies

DRM-AC-PF employs the $\mathrm{Max}\{\}$ block to filter the RCC and the ML block to predict the computational capacity in terms of the envelope of the RCC. On the other hand, the DRM-AC-ES predicts the computational resources based on the RCC; it makes use of a delay block to save the previous Predicted Computational Capacity (PCC) for calculating the error. Finally, it applies a $\mathrm{Max}\{\}$ filter to the error, which is aggregated to the PCC as a marginal amount of computational operations to the predicted computational capacity.

CHAPTER 6. PERFORMANCE EVALUATION

This chapter describes the main results, the former section discusses the performance of the RRH-BBU pool association strategies on the large-scale scenario (scenario 1). It emphasizes on the fronthaul distribution and the network balancing. Moreover, the performance of the proposals to manage the computational resources at BBU pools are discussed in details.

6.1. RRH-BBU pool association analysis

The C-RAN architecture proposed in scenario 1 has been analyzed based on four different planning strategies: minimum delay, load balancing based on traffic or number of RRHs and multiplexing gain algorithm. Those strategies have been adapted from [67]. The maximum fronthaul distance was fixed at 15 km in order to satisfy the delay requirement when optical fiber links are considered.

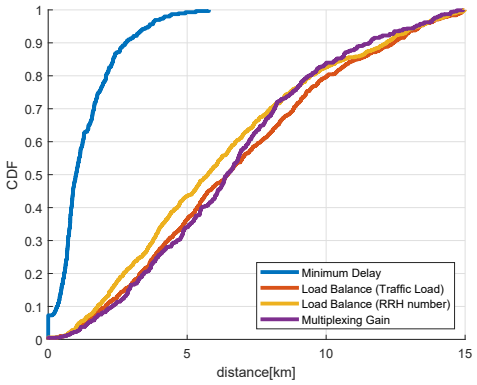
The minimum delay algorithm minimizes the fronthaul distance connecting each RRH to the nearest BBU pool in order to reduce the round trip time. Load balancing algorithms establish the connections balancing the capacity or the number of connected RRHs per BBU pool in the network. Finally, the multiplexing gain algorithm mixes different types of traffic in each BBU pool to achieve a good performance of the overall network.

Fig. 6.1(a) shows the cumulative distribution function (CDF) of the fronthaul distance for each strategy. As expected, with minimum delay strategy most of the RRHs are connected close to the BBU pool, while for the other strategies some RRHs are connected with the maximum fronthaul distance. Minimum delay design not only minimizes the latency, but also reduces the CAPEX of the fronthaul because all the RRHs are connected with fronthaul distances below 6 km. The rest of the strategies exhibit similar performance in terms of delay and fronthaul cost.

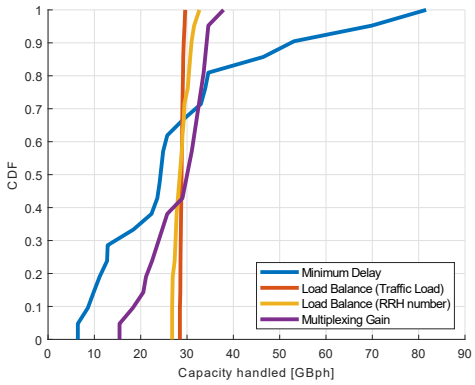
To increase flexibility, an important metric is network balancing. Fig. 6.1(b) shows the distribution of the capacity handled by the BBU pools. For load balancing planning strategies the capacity handled and the number of RRHs per BBU pool are almost constant around 40 RRHs and 28 GBph, which is more robust to face dynamic network variations. On the opposite, minimum delay and multiplexing gain strategies exhibit wider CDFs, hence the worst performance, because there are overloaded BBU pools while others are underutilized.

Fig. 6.1(c) shows that multiplexing gain planning strategy achieves almost constant values of multiplexing gain per BBU pool while the rest of the methods experience lower values for some BBU pools. The performance of this strategy is strongly connected to the traffic profiles handled by the network. Nevertheless, C-RAN architecture improves the current paradigm in mobile communications due to the multiplexing gain, regardless of the planning strategy used.

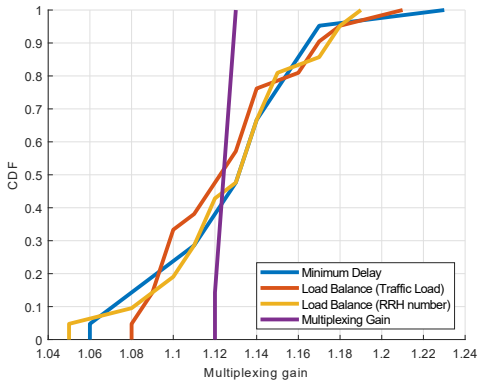
One of the most important requirements of 5G systems is the delay, which must take values up to 1 ms in applications such as virtual reality. The maximum fronthaul distance to satisfy the delay constraint could be estimated: some authors have estimated distances



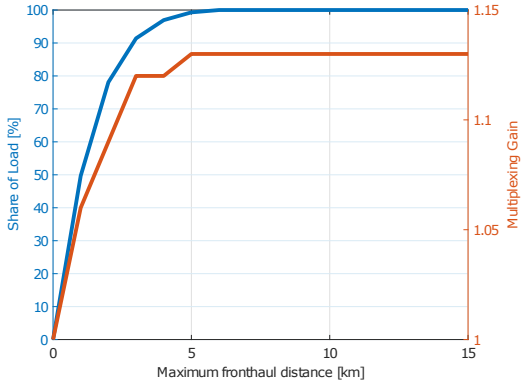
(a) CDF of the fronthaul distance.



(b) CDF of the capacity handled by BBU pool in GBph.



(c) CDF of the multiplexing gain per BBU pool.



(d) Performance of CRAN deployment in terms of the fronthaul distance.

Figure 6.1 Evaluation of the fronthaul connections according to each strategy.

between 20-40 km using optical fiber [68]. Furthermore, the cost to deploy a C-RAN is strongly related to the cost of the optical fiber [3], for this reason in large scenarios mobile operators will centralize the resources of only a certain percentage of the total number of base stations to reduce the CAPEX.

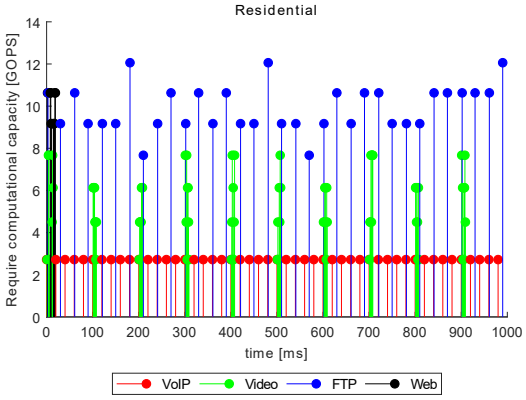
Fig. 6.1(d) shows the performance of the proposed C-RAN deployment in terms of the allowed maximum fronthaul distance. When the maximum fronthaul distance is decreased the percentage of RRHs that are sharing resources in BBU pools also decreases, which results in a degradation of the multiplexing gain because operators have to allocate additional resources to these RRHs. However, the cost to deploy the C-RAN is also reduced, becoming more attractive for small size networks in dense environments as scenario 2. Mobile operators or Infrastructure providers have to take into account this tradeoff in order to reduce the investment. Table 6.1 summarizes the results obtained by each planning strategy for scenario 1.

Table 6.1 Resume table for Scenario 1

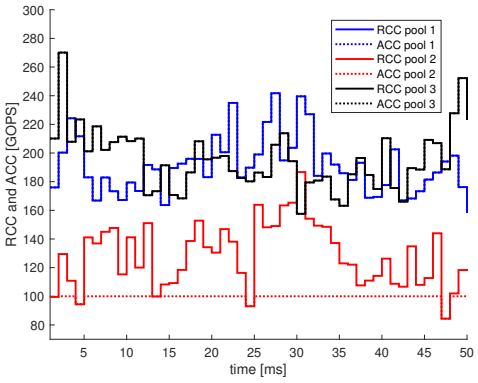
Algorithms	d_{max} [km]	Δ RRHs	Δ C [GBph]	Δ MG
MD	***6***	104	75.18	0.17
LB traffic	15	7	***1.14***	0.13
LB RRHs	15	***1***	5.89	0.14
MG	15	32	22.45	***0.01***

6.2. DRM performance discussion

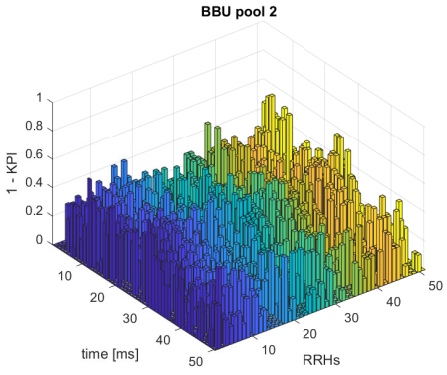
As we have discussed, the advantages of the C-RAN are highlighted in small and dense environments. For this reason, the analysis of the proposed DRM is evaluated in the densest urban zone of the Vienna city (Scenario 2). Fig. 6.2(a) shows the data traffic per service for a residential RRH, during a second at 12:00 pm. It can be appreciated that VoIP and video packets are prioritized by the scheduler. VoIP packets are transmitted in intervals of 40 ms while video service generates 1 frame per 100 ms where each frame contains 8 packets.



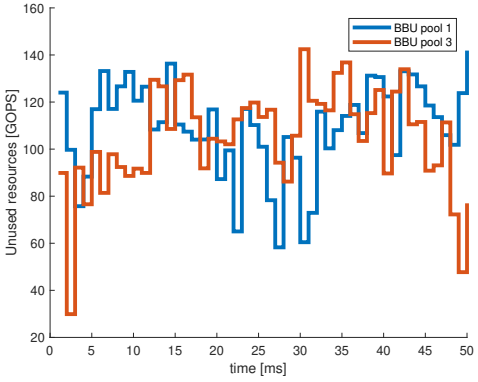
(a) Traffic by services through a residential RRH in scenario 2.



(b) Aggregated capacity at each BBU pool: Required and Allocated.



(c) Temporal QoS per RRH in BBU pool 2.



(d) Unused resources of the hired.

Figure 6.2 Performance evaluation of the DRM

The maximum capacity of each BBU pool has been fixed at 300, 100 and 300 GOPS respectively. Although it is not an optimum selection, it is useful to analyze in detail the behavior of the proposed algorithm. Fig. 6.2(b) shows the performance of the resource management algorithm for each BBU pool, showing clearly the disadvantage of deploying a fixed amount of capacity. The required capacity at each BBU pool is described in a solid line while the allocated capacity is in the dotted line. Assuming that the traffic profile satisfies a fractal property of complex systems, the analysis has been done in an interval of 50 ms.

While for BBU pools 1 and 3 allocated capacity equals the required capacity (there is no difference between solid and discontinuous lines), it is clearly shown that the capacity of BBU pool 2 is not enough to handle the traffic demand. The key performance indicator (KPI) to quantify the QoS is defined as the ratio between the allocated capacity and the required capacity ($KPI \in [0, 1]$). At BBU pool 2 for most of the simulation time, the required capacity is higher than the maximum capacity, which results in a degradation of the QoS. Fig. 6.2(c) represents the percentage of unsatisfied resources per RRH, which is calculated as $1 - KPI$, for each Transmission Time Interval (TTI) considered in the simulation. Details are given only for BBU pool 2, which is in degradation because less than the needed total capacity has been assigned. It can be appreciated that many RRHs experience a high dissatisfaction level which corresponds to a low QoS. Also, each RRH shows a high fluctuation in the QoS parameter along the simulation time. The computational capacity of BBU pool 2 has been intentionally selected low with the aim of highlighting how the proposed algorithm is powerful enough to reveal clearly those cases that have not been properly designed. The influence of bargaining power is observed in Fig. 6.2(c). Notice that at the same time instant there are BBUs with different QoS because the optimization algorithm is allocating more resources to the cells with high priority services.

BBU pools 1 and 2 have enough capacity to handle the demand, but as this capacity is fixed there are intervals where it is underutilized as Fig. 6.2(d) remarks. A proactive network capable of forecasting the required capacity while optimizing QoS with efficient use of the contracted capacity is necessary. The increasing complexity of 5G networks makes planning based on mathematical models not suitable anymore. So, intelligent resource management tools based on Machine Learning (ML) approaches, where the system is able to learn from past situations to proactively predict the traffic demand, are required to optimize future dynamic infrastructure networks.

6.3. DRM-AC performance discussion

6.3.1. Configuration of ML models and data analysis

As it has been above-mentioned, ML based resource management tools are required to optimize the use of the resources at BBU pools. In this case, the system would be able to learn from past situations to proactively predict the traffic demand. This section describes the database, and it establishes the simulation conditions of the supervised

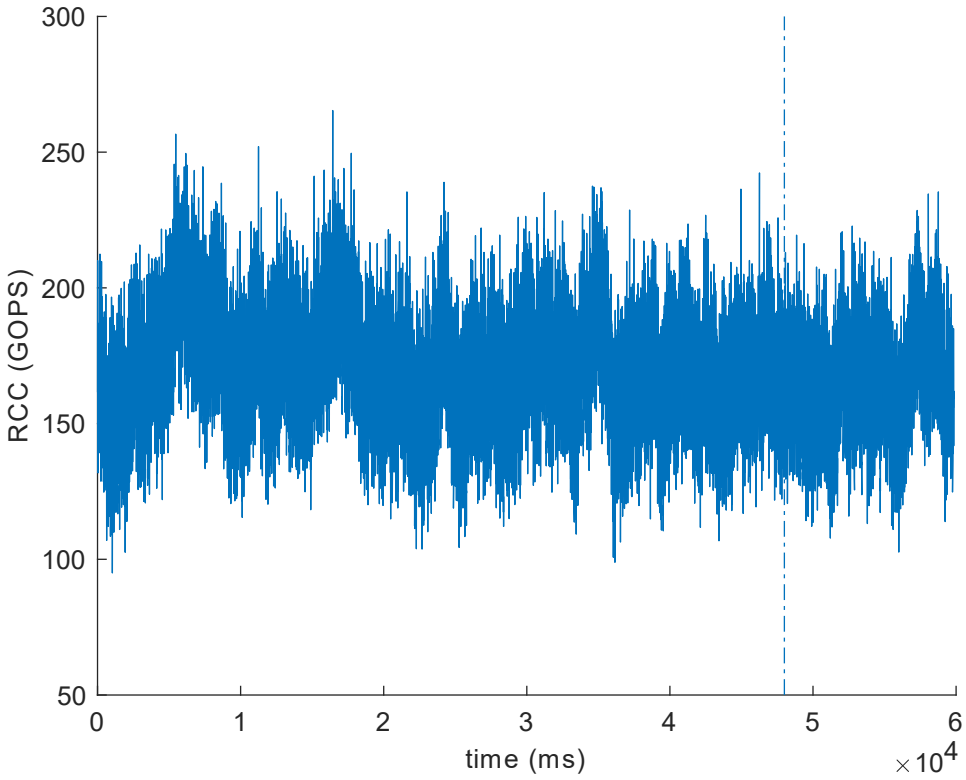


Figure 6.3 Instantaneous evolution of the RCC at BBU pool 1. Database of 60000 samples. First 80 % of the data is used as a training set and the remaining 20 % as a testing set.

learning techniques (SVM, TDNN, and LSTM) in the DRMAC.

6.3.1.1. Data Configuration

For simplicity, the analysis of the forecasting models has been limited only to BBU pool 1, and one minute of traffic database is generated. Fig. 6.3 shows the database, which is split in a training set (first 80 %) and a testing set (the remaining 20 %); the dotted line indicates the boundary between those sets.

6.3.1.2. Models Configuration

SVM and TDNN models predict the RCC based on a set of previous time steps. Hence, an analysis of how many previous time-steps are required to predict the RCC is necessary. First approximation is carried out by the calculation of the sample partial autocorrelation function (PACF), represented in Fig. 6.4. PACF values are split according to their amplitudes in high and low contribution with a threshold of 10 % of the maximum value. The PACF decreases with the number of previous time steps, with the exception of some isolated values (four samples after 250 ms). The cumulative distribution function (CDF)

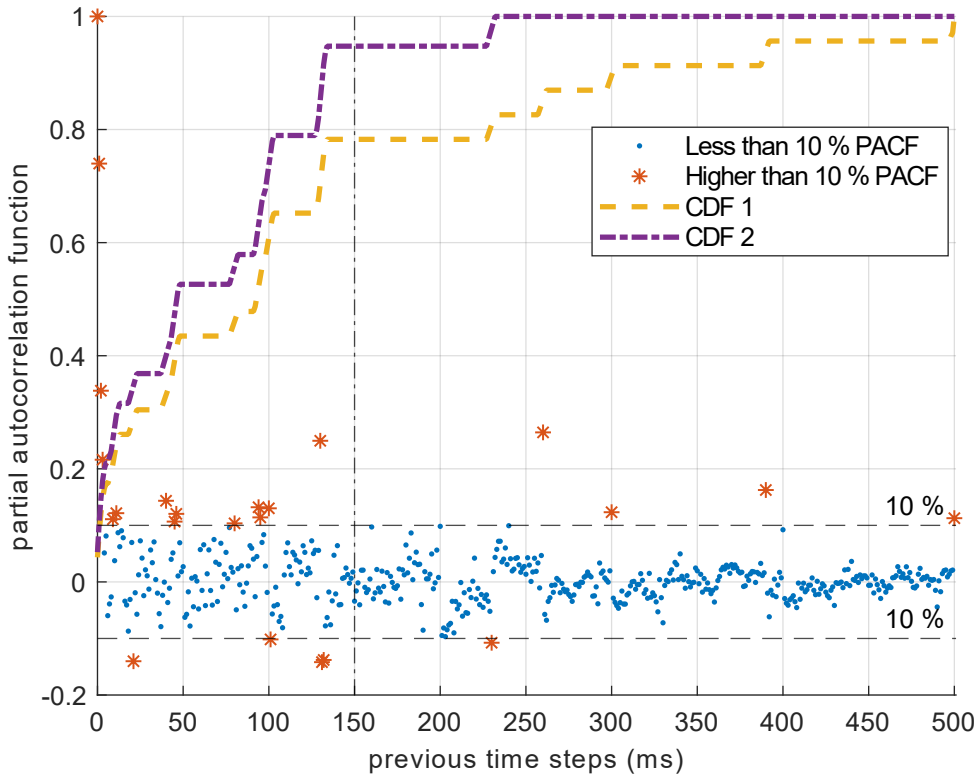


Figure 6.4 Partial autocorrelation function of the database concerning 500 previous time-steps.

of the high contribution values (CDF 1) is shown on Fig. 6.4, the 78 % of the values are located before 150 ms. Furthermore, the CDF of the high contribution values without concerning the isolated samples after 250 ms is also shown (CDF 2), where the 97 % of the samples are before 150 ms. Based on this fact, previous 150 ms are considered as a significant time window to adjust this parameter in SVM and TDNN.

After testing multiple configurations of SVM and TDNN, the best results were obtained using SVM with a Gaussian kernel and TDNN with two hidden layers of 10 neurons and sigmoid as the activation function. Fig. 6.5 shows the root-mean-square error (RMSE) of SVM and TDNN using different amounts of previous time-steps until 150 ms. RMSE decreases when the number of previous time-steps increases; however, after 100 ms and 130 ms in SVM and TDNN respectively, RMSE remains almost constant. For this reason, only $\theta = 100$ ms and $N = 130$ ms previous time-steps are considered in the subsequent analysis. Nevertheless, the method based on PACF is shown to be a perfectly valid rule-of-thumb and there would be no need to test the different cases each time.

Regarding the LSTM approach, its performance does not depend on the number of previous time steps because their contribution is saved in the internal gates of the LSTM cell. However, different network architectures were tested and compared to find a suitable deep learning scheme. Table 6.2 summarizes those architectures. Two hidden

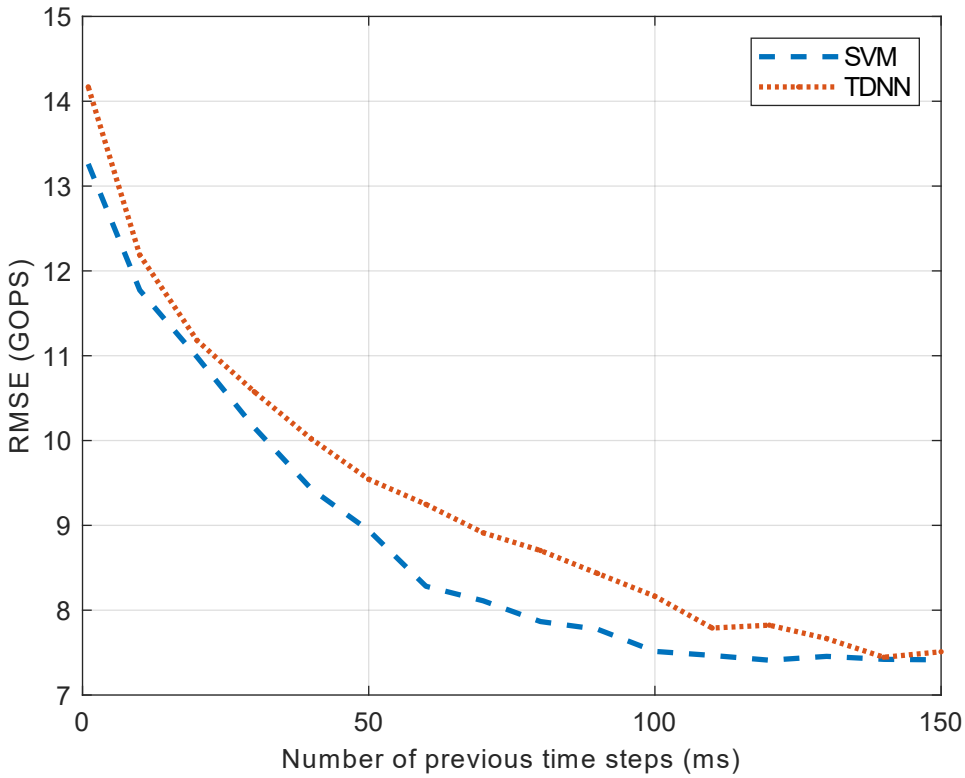


Figure 6.5 Gaussian SVM and TDNN performance in terms of the number of previous steps.

layers with different numbers of LSTM cells, where the learning process takes place, are used. Following [69] recommendation, dropout layers (with a dropping probability of 0.2) are used after each hidden layer to prevent overfitting. Finally, a regression output layer is aggregated to map the output of the last hidden layer to a predicted value.

Fig. 6.6 shows the performance of the network structures in Table 6.2, based on the RMSE achieved in the testing dataset (last 20 % of the data). The RMSE decreases when the number of LSTM cells increases, reaching the its minimum value for network structure number four. For this reason, this structure is selected for the comparison with SVM and TDNN strategies. It has less computational cost than higher network structure labels. The RMSE under this architecture is 12.6 GOPS, which represents 7.6 % of the mean value.

6.3.2. Evaluation and results

In this section, the performance of a DRM with fixed capacity is presented as a benchmark, as well as the results of the proposed strategies: DRM-AC, DRM-AC-PF, and DRM-AC-PF.

However, although BBU pools 1 and 2 had enough capacity to handle the demand Fig.

Table 6.2 Tested deep learning LSTM architectures

	Network structure : index									
	1	2	3	4	5	6	7	8	9	10
L1	Sequential input layer									
L2	Hidden layer: number of LSTM cells									
	20	40	60	80	100	120	140	160	180	200
L3	Dropout: probability of dropping out 0.2									
L4	Hidden layer: number of LSTM cells									
	10	20	30	40	50	60	70	80	90	100
L5	Dropout: probability of dropping out 0.2									
L6	Regression output layer									

6.2(d) shows that resources were underutilized. Consequently, there is a trade-off between the QoS degradation when the computational capacity is under-provisioned and the inefficient use of the resources when the network is over-provisioned. Hence, the next subsections present how the proposed DRM-AC, DRM-AC-PF, and DRM-AC-ES address this trade-off in the BBU pool 1.

6.3.2.1. DRM-AC results evaluation

Fig. 6.7 summarizes the performance of the DRM-AC using each ML approach. Fig. 6.7(a), 6.7(b) and 6.7(c) show the predicted computational capacity in terms of the real computational demand of each strategy. Most of the predicted values are close to the perfect prediction line, being the degree of dispersion an indicator of the quality of the prediction strategy. The maximum error of SVM and TDNN approaches are around 35 GOPS, and the RMSE is close to 7.5 GOPS, which represents a deviation of 4.5 % of the mean value of the overall dataset. The Pearson correlation coefficients (slope of the regression line) are 0.92 and 0.89 for SVM and TDNN, respectively. On the other hand, the LSTM strategy presents a RMSE $=$ 12.6 GOPS that depicts the 7.6 % of the mean value and the Pearson coefficient is r $= 0.7$, which is more deviated from the perfect prediction line.

Fig. 6.7(d) shows the error distribution of each approach. Regardless of the used strategy, the error distribution is almost a Gaussian curve with zero mean. As the ML algorithms predict the required computational capacity at the BBU pool, it is important to analyze the effect of these errors. Positive errors (right side of perfect prediction line on Fig. 6.7(d)) represent the amount of underutilized resources, while negative errors are the amount of unsatisfied resources. The main objective is to minimize the underutilized resources while maintaining the QoS. Improving the prediction capacity of the machine learning strategies is not enough to address this challenge because the negative errors always reduce the QoS. Table 6.3 summarizes the behaviour of the three proposed strategies.

SVM and TDNN improve the performance of LSTM in 3 %. However, as it is possible

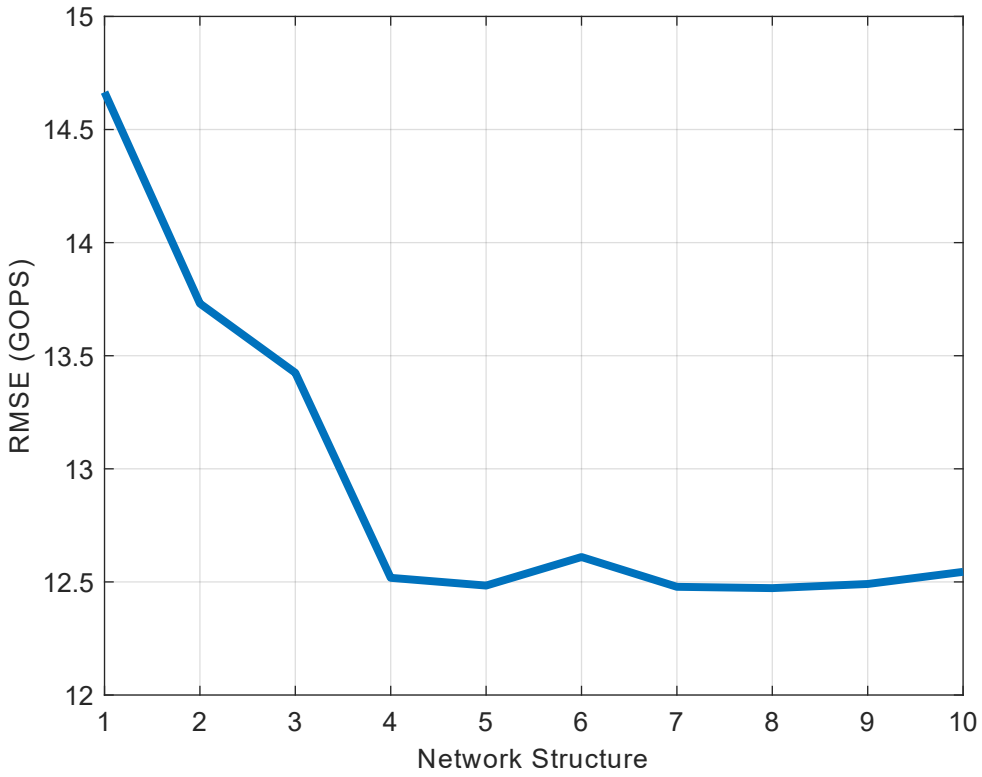


Figure 6.6 Performance (RMSE) on the testing data of the deep learning LSTM architectures in Table 6.2.

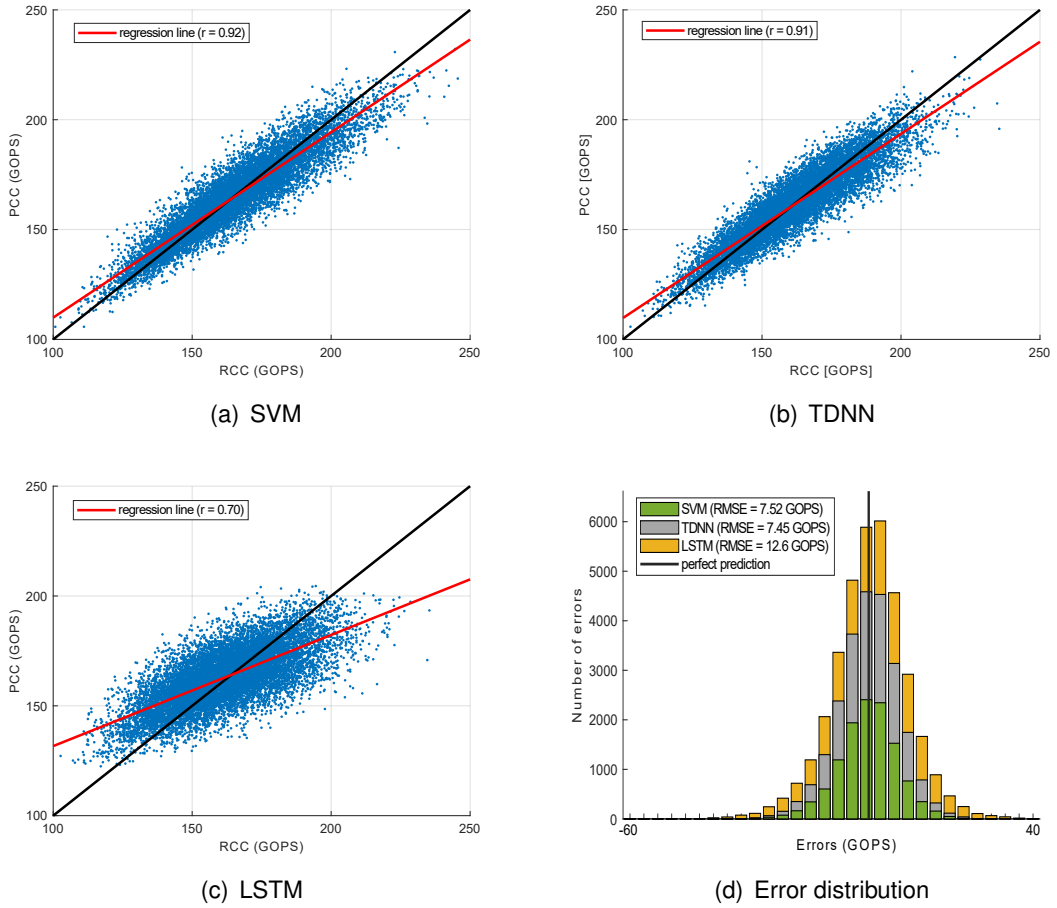
Table 6.3 Summary of the proposed ML techniques.

ML technique	RMSE (GOPS)	RMSE (%)	Pearson coefficient
SVM	7.52	4.5	0.92
TDNN	7.45	4.47	0.91
LSTM	12.6	7.6	0.7

to see in Fig. 6.5, the behavior of SVM and TDNN strongly depend on the number of previous time steps used in the prediction. As mobile networks experience large fluctuations and they are not stationary processes, results obtained under the assumption of variable parameters as the number of previous time-steps might be more robust. The design based on LSTM cells is an example; it obtains similar performance to Gaussian SVM and TDNN without requiring a fixed number of previous time steps. The useful information of the previous time-steps is stored in the forget gates of the LSTM entities in the hidden layers.

6.3.2.2. DRM-AC-PF and DRM-AC-ES performance evaluation

As it was afore-mentioned, the LSTM approach could be more robust to face high fluctuation environments. For this reason and without losing generality, performances of



(a) SVM (b) TDNN

(c) LSTM (d) Error distribution

Figure 6.7 Performance of the DRM-AC for each ML technique. (a), (b), and (c) show the predicted computational capacity in terms of the real computational demand of SVM, TDNN and LSTM respectively. Black lines denote perfect prediction lines, red line depicts the regression line and r is the Pearson correlation coefficient. (d) represents the histogram of the error distribution.

DRM-AC-PF and DRM-AC-ES, reducing negative errors, are evaluated based on the LSTM approach.

Fig. 6.8 shows the performance of the solution applying DRM-AC-PF. The $\mathrm{Max}\{\}$ block extracts the envelope of the RCC acting as a low pass filter eliminating the fastest variations; the solid blue line represents the filtered computational capacity. The fixed capacity (300 GOPS) is also represented to remark the advantage of predicting the required computational capacity.

As it has been mentioned above, negative errors cause QoS degradation. Fig. 6.9 exhibits the distribution of the errors of the proposed schemes using the same LSTM architecture. Although positive errors in DRM-AC-ES have increased, negative errors are almost eliminated. In the case of the DRM-AC-PF, the results are similar; negative errors appear only in isolated cases at the cost of increasing the positive error with respect to the original LSTM approach (LSTM DRM-AC on Fig. 6.9).

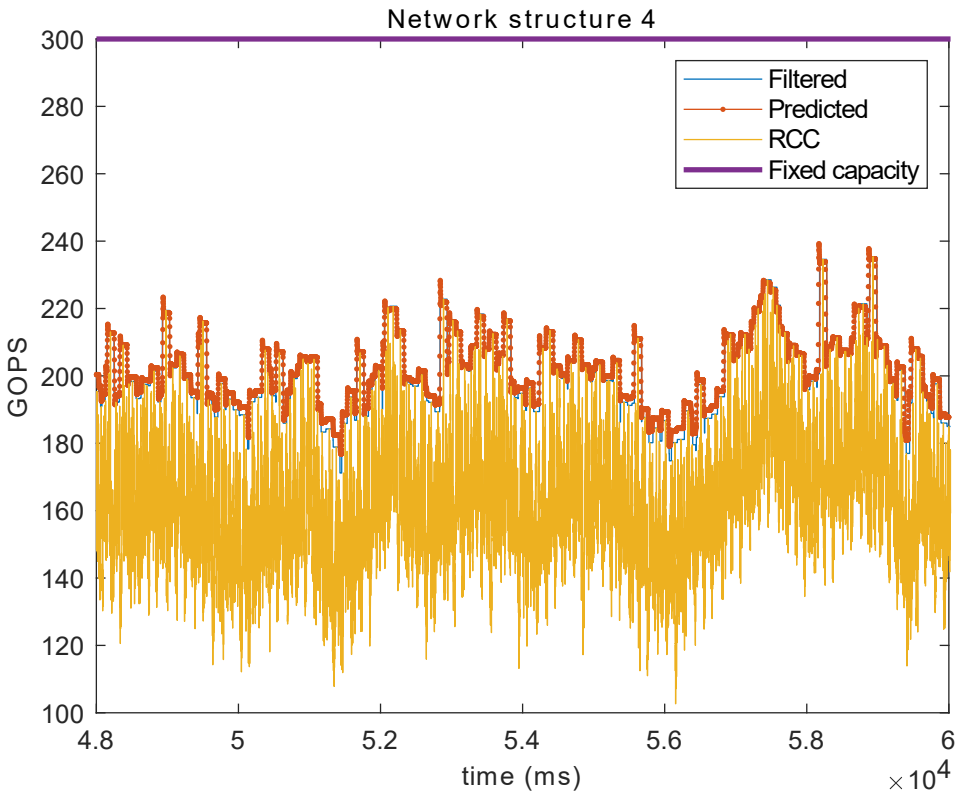


Figure 6.8 Evolution of the computational capacity at BBU pool 1 showing the fixed maximum computational capacity, the RCC during the testing dataset, the filtered RCC and the predicted computational capacity after applying DRM-AC-PF strategy.

Two key performance indicators have been defined to facilitate a numerical comparison of the strategies: the mean of unused resources (MUR_+) and the mean of unsatisfied resources (MUR_-), calculated by (6.1) and (6.2), respectively.

$$\text{MUR}_+ = \frac{1}{K}\sum_{j=1}^{K} e_j^+ \tag{6.1}$$

$$\text{MUR}_- = \frac{1}{K}\sum_{j=1}^{K} e_j^-, \tag{6.2}$$

being K the number of time-steps in the whole database ($K = 60000$ ms), e_j^+ and e_j^- depict the absolute values of each kind of error at instant j in GOPS. Those errors are complementary because only one of them could be different from zero.

Table 6.4 shows the advantage of using each strategy in terms of MUR_+ and MUR_- key performance indicators. The DRM without adaptive capacity has an average of 138.56 GOPS/ms of unused resources. Under the considered traffic conditions and with a fixed capacity (300 GOPS) in BBU pool 1, the resources are enough to handle the instantaneous RCC ($\text{MUR}_- = 0$). However, as the maximum capacity is fixed, if the RCC surpasses the maximum capacity at BBU pool 1, UEs would be in degradation;

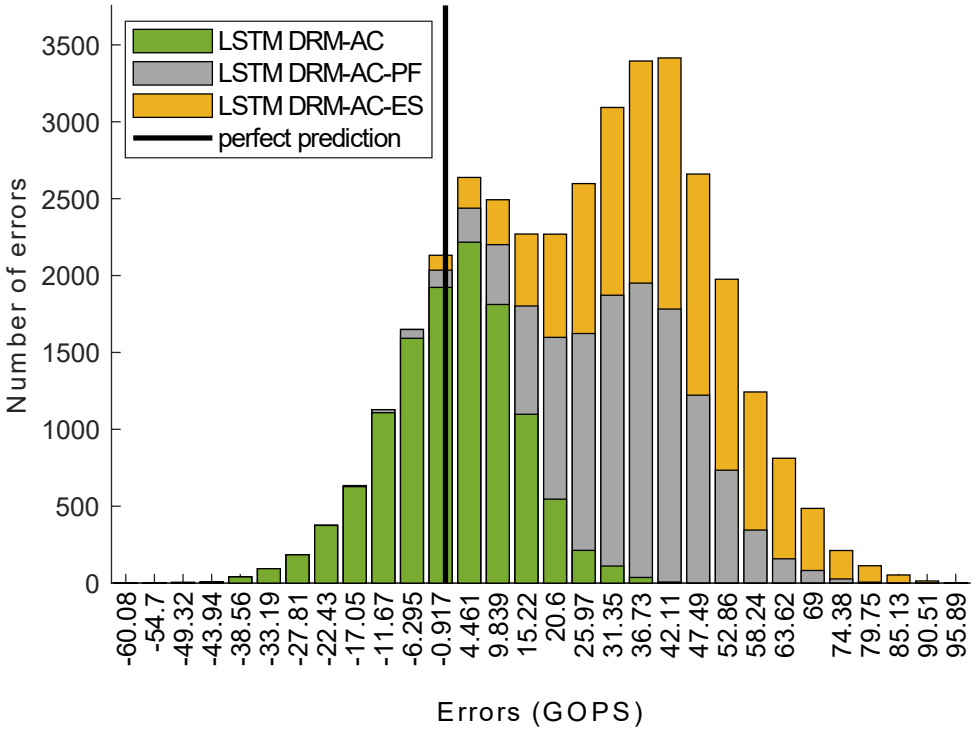


Figure 6.9 Error distribution for DRM-AC, DRM-AC-PF and DRM-AC-ES.

Table 6.4 Performance summary in terms of the MUR_+ and MUR_-.

Proposals	MUR_+ (GOPS/ms)	MUR_- (GOPS/ms)
DRM	138.56	0
DRM-AC	5.5	4.49
DRM-AC-PF	41.15	0.0016
DRM-AC-ES	34.08	0.072

consequently, the MUR_- would increase, and the QoS would be degraded. DRM-AC reduces the MUR_+ considerably (5.5 GOPS/ms), but the error in the prediction causes that approximately 50 % of the time, the instantiated resources are not large enough to satisfy the demand. DRM-AC-PF and DRM-AC-ES strategies reduce considerably the MUR_- at the cost to increase the average of unused resources but maintain the UEs QoS.

CONCLUSIONS

Two different scale and realistic deployments of C-RAN in COST 1004 Vienna city scenario have been proposed. Firstly, four RRH-BBU pool association strategies were compared in the preliminary planning stage using a simple per hour traffic profile in a large-scale scenario. A large-scale C-RAN design, in which a percentage of cells are centralized to reduce the investment, is also presented. This analysis may support network operators to implement an optimal design accounting for the cost of the optical fiber, the area to be covered, and the users' density.

The design of a C-RAN deployment for the metropolitan area has been analyzed jointly with a resource management strategy to allocate resources at the BBU pools. The study includes realistic traffic profiles where UEs generate traffic of different services at the packet level. Results show that the proposed optimization algorithm is capable of allocating resources introducing QoS constraints with service priority. However, due to the classical assumption of fixed capacity at the BBU pools, there are intervals where the resources are underutilized.

For this reason, this research work integrates ML techniques to dynamic resource management. Three ML strategies have been implemented and exhaustively compared: SVM, TDNN, and LSTM in terms of their ability to predict the instantaneous computational capacity at the BBU pools. DRM-AC reduces the underutilized resources by 96 % when compared with the DRM with fixed computational resources. However, it degrades the QoS when the predicted computational resources are not enough to satisfy the demand. This situation appears approximately 50 % of the time due to the error follows a gaussian distribution with zero mean. This issue is solved by proposing two novel strategies. Firstly, a DRM-AC with prefiltering, where high-frequency variations in input data are removed. DRM-AC-PF extracts the envelope of the RCC, it improves the learning process, and it almost eliminates the QoS degradation. Secondly, DRM-AC-ES monitors the maximum error computed in past observation times. It allows estimating a marginal amount of resources to be added to the predicted computational capacity.

As a consequence, DRM and DRM-AC are outperformed. DRM-AC-PF and DRM-AC-ES reduce the unsatisfied resources by 99.9 % and 98 % compared to the DRM-AC, respectively. Moreover, they reduce the underutilized resources by 70 % and 75 % compared to the DRM, respectively.

Future Works

Adaptive capabilities have already been aggregated to the network to avoid under and over-provision of the resources. However, there is room for improvement using the natural big-data of the network to upgrade the performance. This work addresses the issue of optimizing the resources at BBU pools, but there is a physical capacity limit at BBU pools. The required computational capacity could surpass this physical limit. Consequently, the prediction of high peaks of traffics to reallocate the traffic demands to an available BBU pool proactively is a new challenge that needs attention. The introduction of network slicing to the current C-RAN model is a critical improvement to solve this issue. Because the network should differentiate among services to reallocate the traffic of not delay sensible services (e.g., uRLLC). It should be combined with a testbed implementation using Universal Software Radio Peripherals (USRPs).

Besides, it is necessary to create an autonomous cognitive network to jointly account for the previous concepts and the introduction of the proactive skill of reacting to failures. To do so, the definition of a procedure for training the network is a key aspect. On the other hand, a set of challenges and open issues that could be tackled in future works have been identified in section 2.4..

ACRONYMS

BTS Base Transceiver Station

RF Radio-Frequency

UE User Equipment

SCell Small Cell

SBS Small Base Station

PCell Pico Cell

MCell Macro Cell

3GPP 3rd Generation Partnership Project

DRM Dynamic Resource Management

DRM-AC DRM with adaptive capacity

DRM-AC-PF DRM-AC with prefiltering

DRM-AC-ES DRM with error shifting

IoT Internet of Things

AR Augmented Reality

BBU Baseband Unit

RRH Remote Radio Head

RRS Remote Radio System

IQ In-phase and Quadrature

CPRI Common Public Radio Interface

ARoF Analogue Radio-over-Fiber

PLS Physical Layer Split

LTE Long Term Evolution

CoMP Coordinated Multipoint

JT Joint Transmission

eICIC enhanced Inter-Cell Interference Coordination

BS Base Station

MBS Macro Base Station

2G second-Generation

3G third-Generation

4G fourth-Generation

5G fifth-Generation

MIMO Multiple-Input Multiple-Output

WNV Wireless Network Virtualization

OPEX Operating Expenditures

CAPEX Capital Expenditures

TCO Total Cost of Ownership

k-NN k-Nearest Neighbor

DT Decision Tree

NN Neural Network

TDNN Time Delay Neural Network

SVM Support Vector Machine

RL Reinforcement Learning

DRL Deep Reinforcement Learning

RAN Radio Access Network

C-RAN Cloud Radio Access Network

Soft-RAN Software-Defined for Radio Access Networks

NFV Network Function Virtualization

SDN Software-Defined Network

H-CRAN Heterogeneous Cloud Radio Access Network

Flex-RAN fexible and programmable platform for the software-defined RAN

SDHC-CRAN Software-defined Hyper-Cellular C-RAN

F-RAN Fog Radio Access Networks

HSD-RAN Hierarchical Software-Defined RAN

SDVRAN Software-Defined and Virtualized RAN

HVSD-CRAN Heterogeneous Virtualized Software-Defined C-RAN

QoS Quality of Service

MORA Multi-Objective Resource Allocation

EH Energy Harvesting

uRLLC ultra-Reliable Low Latency Communication

eMBB enhanced Mobile Broadband

mMTC Massive Machine Type Communications

MTL multitask learning

ML machine learning

RNN Recurrent Neural Network

3D-CNN 3D convolutional-NN

CNN Convolutional Neural Network

RB Resource block

CSI Channel State Information

LSTM Long Short-Term Memory

OAI OpenAirInterface

IP Internet Protocol

UAV Unmanned Aerial Vehicle

SDR Software-Defined Radio

SBI Southbound Interface

MILP mixed-integer linear programming

RCC Required Computational Capacity

PCC Predicted Computational Capacity

MNO Mobile Network Operator

MVNO Mobile Virtual Network Operator

InP Infrastructure Provider

MEC Mobile Edge Computing

LOS Line of Sight

MCS modulation and coding scheme

VoIP Voice over IP

FTP File Transfer Protocol

PRB Physical Resource Block

UF User Processing Function

CF Common Processing Function

CDF Cumulative Distribution Function

TTI Transmission Time Interval

KPI Key Performance Indicator

CAN Cognitive Autonomous Network

USRP Universal Software Radio Peripheral

MCS Modulation and Coding Scheme

UF user-processing function

CF common-processing function

SINR Signal-to-Noise-plus-Interference-Ratio

BIBLIOGRAPHY

[1] China Mobile. C-RAN The Road Towards Green RAN. Technical Report Version 2.5, China Mobile, October 2011.

[2] C. Ranaweera, E. Wong, A. Nirmalathas, C. Jayasundara, and C. Lim. 5G C-RAN architecture: A comparison of multiple optical fronthaul networks. In *2017 International Conference on Optical Network Design and Modeling (ONDM)*, pages 1–6, May 2017.

[3] A. Checko, H. Holm, and H. Christiansen. Optimizing small cell deployment by the use of C-RANs. In *European Wireless 2014; 20th European Wireless Conference*, pages 1–6, May 2014.

[4] V. N. Ha and L. B. Le. End-to-End Network Slicing in Virtualized OFDMA-Based Cloud Radio Access Networks. *IEEE Access*, 5:18675–18691, 2017.

[5] Rolando Guerra-Gómez, Silvia Ruiz, M. Garcia-Lozano, and Joan Olmos. Using COST IC1004 Vienna scenario to test C-RAN optimisation algorithms. In *COST IRACON*, Dublin, Ireland, January 2019.

[6] R. Guerra-Gómez, Silvia Ruiz, M. Garcia-Lozano, and Joan Olmos. A weighted-sum multi-objective optimization for dynamic resource allocation with QoS constraints in realistic C-RAN. In *COST IRACON*, oulu, Finland, May 2019.

[7] R. Guerra-Gómez, Silvia Ruiz, M. Garcia-Lozano, and Joan Olmos. Predicting required computational capacity in C-RAN networks by the use of different machine learning strategies. In *COST IRACON*, Gdańsk, Poland, September 2019.

[8] R. Guerra-Gómez, S. R. Boqué, M. García-Lozano, and J. O. Bonafé. Dynamic resource allocation in C-RAN with real-time traffic and realistic scenarios. In *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 1–6, Oct 2019.

[9] M. Baghani, S. Parsaeefard, and T. Le-Ngoc. Multi-Objective Resource Allocation in Density-Aware Design of C-RAN in 5G. *IEEE Access*, 6:45177–45190, 2018.

[10] Aditya Gudipati, Daniel Perry, Li Erran Li, and Sachin Katti. SoftRAN: software defined radio access network. In *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking - HotSDN '13*, page 25, Hong Kong, China, 2013. ACM Press.

[11] Mao Yang, Yong Li, Depeng Jin, Li Su, Shaowu Ma, and Lieguang Zeng. OpenRAN: a software-defined ran architecture via virtualization. In *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM - SIGCOMM '13*, page 549, Hong Kong, China, 2013. ACM Press.

[12] M. Peng, Y. Li, J. Jiang, J. Li, and C. Wang. Heterogeneous cloud radio access networks: a new perspective for enhancing spectral and energy efficiencies. *IEEE Wireless Communications*, 21(6):126–135, December 2014.

[13] M. Peng, H. Xiang, Y. Cheng, S. Yan, and H. V. Poor. Inter-Tier Interference Suppression in Heterogeneous Cloud Radio Access Networks. *IEEE Access*, 3:2441–2455, 2015.

[14] Jingchu Liu, Shugong Xu, Sheng Zhou, and Zhisheng Niu. Redesigning fronthaul for next-generation networks: beyond baseband samples and point-to-point links. *IEEE Wireless Communications*, 22(5):90–97, October 2015.

[15] Xenofon Foukas, Navid Nikaein, Mohamed M Kassem, and Kimon Kontovasilis. FlexRAN: A flexible and programmable platform for software-defined radio access networks. In *CONEXT 2016, 12th International on Conference on emerging Networking EXperiments and Technologies, Irvine, California, USA*, Irvine, UNITED STATES, 12 2016.

[16] S. Zhou, T. Zhao, Z. Niu, and S. Zhou. Software-defined hyper-cellular architecture for green and elastic wireless access. *IEEE Communications Magazine*, 54(1):12–19, January 2016.

[17] M. Peng, S. Yan, K. Zhang, and C. Wang. Fog-computing-based radio access networks: issues and challenges. *IEEE Network*, 30(4):46–53, July 2016.

[18] X. Chen, Z. Han, Z. Chang, G. Xue, H. Zhang, and M. Bennis. Adapting Downlink Power in Fronthaul-Constrained Hierarchical Software-Defined RANs. In *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–6, March 2017.

[19] K. Liang, L. Zhao, X. Chu, and H. Chen. An Integrated Architecture for Software Defined and Virtualized Radio Access Networks with Fog Computing. *IEEE Network*, 31(1):80–87, January 2017.

[20] K. Thaalbi, M. T. Missaoui, and N. Tabbane. Short Survey on Clustering Techniques for RRH in 5G networks. In *2018 Seventh International Conference on Communications and Networking (ComNet)*, pages 1–5, November 2018.

[21] H. Taleb, M. E. Helou, S. Lahoud, K. Khawam, and S. Martin. Multi-Objective Optimization for RRH Clustering in Cloud Radio Access Networks. In *2018 International Conference on Computer and Applications (ICCA)*, pages 85–89, August 2018.

[22] D. Zhang, Z. Chen, L. X. Cai, H. Zhou, S. Duan, J. Ren, X. Shen, and Y. Zhang. Resource allocation for green cloud radio access networks with hybrid energy supplies. *IEEE Transactions on Vehicular Technology*, 67(2):1684–1697, February 2018.

[23] H. Dai, Y. Huang, J. Wang, and L. Yang. Resource Optimization in Heterogeneous Cloud Radio Access Networks. *IEEE Communications Letters*, 22(3):494–497, March 2018.

[24] Karen Boulos, Kinda Khawam, Melhem El Helou, Marc Ibrahim, Steven Martin, and Hadi Sawaya. A hybrid approach for RRH clustering in Cloud Radio Access Networks Based on Game Theory. In *Proceedings of the 16th ACM International Symposium on Mobility Management and Wireless Access*, MobiWac'18, pages 128–132, New York, NY, USA, 2018. ACM.

[25] K. Boulos, K. Khawam, M. El Helou, M. Ibrahim, H. Sawaya, and S. Martin. An Efficient Scheme for BBU-RRH Association in C-RAN Architecture for Joint Power Saving and Re-Association Optimization. In *2018 IEEE 7th International Conference on Cloud Networking (CloudNet)*, pages 1–6, October 2018.

[26] H. Taleb, M. El Helou, K. Khawam, S. Lahoud, and S. Martin. Centralized and distributed RRH clustering in cloud radio access networks. In *2017 IEEE Symposium on Computers and Communications (ISCC)*, pages 1091–1097, July 2017.

[27] M. K. Elhattab, M. M. Elmesalawy, T. Ismail, H. H. Esmat, M. M. Abdelhakam, and H. Selmy. A Matching Game for Device Association and Resource Allocation in Heterogeneous Cloud Radio Access Networks. *IEEE Communications Letters*, 22(8):1664–1667, August 2018.

[28] S. F. Abedin, M. G. R. Alam, S. M. A. Kazmi, N. H. Tran, D. Niyato, and C. S. Hong. Resource allocation for ultra-reliable and enhanced mobile broadband IoT applications in fog network. *IEEE Transactions on Communications*, 67(1):489–502, Jan 2019.

[29] J. Riihijarvi and P. Mahonen. Machine Learning for Performance Prediction in Mobile Cellular Networks. *IEEE Computational Intelligence Magazine*, 13(1):51–60, February 2018.

[30] E. Balevi and R. D. Gitlin. Unsupervised machine learning in 5G networks for low latency communications. In *2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC)*, pages 1–2, December 2017.

[31] C. Huang, C. Chiang, and Q. Li. A study of deep learning networks on mobile traffic forecasting. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 1–6, October 2017.

[32] M. Chen, W. Saad, C. Yin, and M. Debbah. Echo State Networks for Proactive Caching in Cloud-Based Radio Access Networks With Mobile Users. *IEEE Transactions on Wireless Communications*, 16(6):3520–3535, June 2017.

[33] I. AlQerm and B. Shihada. Enhanced machine learning scheme for energy efficient resource allocation in 5G heterogeneous cloud radio access networks. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 1–7, October 2017.

[34] G. Shen, L. Pei, P. Zhiwen, L. Nan, and Y. Xiaohu. Machine learning based small cell cache strategy for ultra dense networks. In *2017 9th International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 1–6, October 2017.

[35] A. Y. Nikravesh, S. A. Ajila, C. Lung, and W. Ding. Mobile Network Traffic Prediction Using MLP, MLPWD, and SVM. In *2016 IEEE International Congress on Big Data (BigData Congress)*, pages 402–409, June 2016.

[36] S. Imtiaz, H. Ghauch, G. P. Koudouridis, and J. Gross. Random forests resource allocation for 5G systems: Performance and robustness study. In *2018 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pages 326–331, April 2018.

[37] T. Gao, M. Chen, H. Gu, and C. Yin. Reinforcement learning based resource allocation in cache-enabled small cell networks with mobile users. In *2017 IEEE/CIC International Conference on Communications in China (ICCC)*, pages 1–6, October 2017.

[38] L. Le, B. P. Lin, L. Tung, and D. Sinh. SDN/NFV, Machine Learning, and Big Data Driven Network Slicing for 5G. In *2018 IEEE 5G World Forum (5GWF)*, pages 20–25, July 2018.

[39] 3GPP. 3GPP TR 38.801 V14.0.0 (2017-03): Study on new radio access technology: Radio access architecture and interfaces. Technical report, 3GPP, 2017.

[40] L. M. P. Larsen, A. Checko, and H. L. Christiansen. A Survey of the Functional Splits Proposed for 5G Mobile Crosshaul Networks. *IEEE Communications Surveys Tutorials*, pages 1–1, 2018.

[41] K. Boulos, M. E. Helou, K. Khawam, M. Ibrahim, S. Martin, and H. Sawaya. RRH clustering in cloud radio access networks with re-association consideration. In *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–6, April 2018.

[42] 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)*, pages 1–5, December 2015.

[43] M. Khan, Z. H. Fakhri, and H. S. Al-Raweshidy. Semistatic Cell Differentiation and Integration With Dynamic BBU-RRH Mapping in Cloud Radio Access Network. *IEEE Transactions on Network and Service Management*, 15(1):289–303, March 2018.

[44] Behnam Rouzbehani, Luis M. Correia, and Luísa Caeiro. An SLA-Based Method for Radio Resource Slicing and Allocation in Virtual RANs. In *COST IRACON*, Cartagena, Spain, June 2018.

[45] C. Lee, M. Lee, J. Wu, and W. Chang. A Feasible 5G Cloud-RAN Architecture with Network Slicing Functionality. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 442–449, November 2018.

[46] S. Costanzo, I. Fajjari, N. Aitsaadi, and R. Langar. Dynamic Network Slicing for 5G IoT and eMBB services: A New Design with Prototype and Implementation Results. In *2018 3rd Cloudification of the Internet of Things (CIoT)*, pages 1–7, July 2018.

[47] A. Alnoman, G. H. S. Carvalho, A. Anpalagan, and I. Woungang. Energy Efficiency on Fully Cloudified Mobile Networks: Survey, Challenges, and Open Issues. *IEEE Communications Surveys Tutorials*, 20(2):1271–1291, 2018.

[48] A. Younis, T. X. Tran, and D. Pompili. Bandwidth and Energy-Aware Resource Allocation for Cloud Radio Access Networks. *IEEE Transactions on Wireless Communications*, 17(10):6487–6500, October 2018.

[49] Aleksandra Checko, Henrik Christiansen, and Michael S Berger. Evaluation of energy and cost savings in mobile Cloud RAN. In *Proceedings of OPNETWORK Conference*, page 8, 2013.

[50] K. Lin, W. Wang, Y. Zhang, and L. Peng. Green Spectrum Assignment in Secure Cloud Radio Network with Cluster Formation. *IEEE Transactions on Sustainable Computing*, pages 1–1, 2018.

[51] R. F. Ahmed, T. Ismail, L. F. Abdelal, and N. Hassan Sweilam. Optimization of Power Consumption and Handover Margin of Sleep/Active Cells in Dynamic H-CRAN. In *2018 11th International Symposium on Communication Systems, Networks Digital Signal Processing (CSNDSP)*, pages 1–6, July 2018.

[52] X. Wang, K. Wang, S. Wu, S. Di, H. Jin, K. Yang, and S. Ou. Dynamic Resource Scheduling in Mobile Edge Cloud with Cloud Radio Access Network. *IEEE Transactions on Parallel and Distributed Systems*, 29(11):2429–2445, November 2018.

[53] 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*, 5:19099–19115, 2017.

[54] O. Narmanlioglu and E. Zeydan. Learning in SDN-based multi-tenant cellular networks: A game-theoretic perspective. In *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pages 929–934, May 2017.

[55] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995.

[56] Harris Drucker, Christopher J. C. Burges, Linda Kaufman, Alex J. Smola, and Vladimir Vapnik. Support vector regression machines. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems 9*, pages 155–161. MIT Press, 1997.

[57] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, November 1997.

[58] Hui Wang, Mario Garcia Lozano, Edward Mutafungwa, Xuefeng Yin, and Silvia Ruiz. Performance comparison of uplink and downlink techniques under DUDe strategy for heterogeneous networks. In *EURO-COST*, Portugal, February 2017.

[59] S.Ruiz, H. Ahmadi, L.M.Caeiro, N.Cardona, L.M.Correia, M.Garcia-Lozano, T.Javornik, and V.Petrini. IRANCON reference scenarios. In *EURO-COST*, Nicosia, January 2018.

[60] H. Wang, M. Garcia-Lozano, E. Mutafungwa, X. Yin, and S. Ruiz. Performance study of uplink and downlink splitting in ultradense highly loaded networks. *Wireless Communications and Mobile Computing*, 2018:12, July 2018.

[61] U. Saeed, J. Hämäläinen, M. Garcia-Lozano, and G. David González. On the feasibility of remote driving application over dense 5G roadside networks. In *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*, pages 271–276, Aug 2019.

[62] Mojgan Barahman, Luis M. Correia, and Lucio S. Ferreira. A real-time computational resource management in C-RAN. In *EURO-COST*, Cartagena, Spain, May 2018.

[63] Joan Olmos, Silvia Ruiz, Mario Garcia Lozano, and David Martin Sacristan. Link abstraction models based on mutual information for LTE downlink. In *EURO-COST*, Aalborg, Denmark, June 2010.

[64] 3GPP. 3GPP TR 36.213 v14.3.0 (2017-05): Technical specification group radio access network; evolved universal terrestrial radio access (e-utra); physical layer procedures. Technical report, 3GPP, 2017.

[65] B. Debaillie, C. Desset, and F. Louagie. A flexible and future-proof power model for cellular base stations. In *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, pages 1–7, May 2015.

[66] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, March 2014.

[67] Tiago Monteiro, Luis M. Correia, and Ricardo Dinis. Implementation analysis of cloud radio access networks architectures in small cells. In *EURO-COST*, Portugal, February 2017.

[68] A. Checko, H. L. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. S. Berger, and L. Dittmann. Cloud RAN for mobile networks: A technology overview. *IEEE Communications Surveys Tutorials*, 17(1):405–426, 2015.

[69] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, January 2014.