UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Flexible Functional Split in the 5G Radio Access Networks

Doctoral Dissertation

Candidate Davit Harutyunyan, University of Trento and FBK CREATE-NET, Italy

Advisor Dr. Roberto Riggio, FBK CREATE-NET, Italy

Committee Prof. Abbas Bradai, University of Poitiers, France
Prof. Ramon Ferrús, Polytechnic University of Catalonia, Spain
Prof. Wolfgang Kellerer, Technical University of Munich, Germany

Trento, May 2019

A man who dares to waste one hour of time has not discovered the value of life.
Charles Darwin

Acknowledgements

First time I visited Trento as an exchange program student for my Master's study. To be quite honest, I just fell in love with the city and with the education system at the University of Trento. These factors motivated me to pursue a doctoral study at this university. I was lucky to meet **Leonardo Goratti**, who helped me to realize my goal. He introduced me to **Tinku Rasheed**, who that time was the head of the Future Networks area at CREATE–NET research center, with whom I had the shortest and by far the most memorable interview in my life. I would like to thank them both for trusting me a PhD position and welcoming me to their team.

During my entire PhD study at CREATE–NET, I consider myself as a fortunate guy for having **Roberto Riggio** as my advisor. I would like to express my sincere gratitude to him for his indispensable help to shape my research topic and for giving me freedom and autonomy in conducting research activities. If I were to name only one thing that I am thankful to him for, I would mention the fact that he was so kind and friendly that I never hesitated to enter his office to discuss some research matters or ask some questions (often stupid or minor) even without having an appointment. I highly appreciate the fact that even when he was busy with his tons of commitments, he always found time for me.

Not only did I enrich my vocational knowledge and experience during my PhD journey, but I also made many friends, whom I would like to thank for their support. **Vevake**, my very first and the best friend in Trento, I deeply value your assistance in every aspect of my life.

I would like to show my gratitude to my lovely global and local families. My dear mother **Silva**, sister **Anush** and brother **Roman**, I am extremely delighted that with my achievements during my study I was able to make you proud of me, and I am sure, so would have been my father **Gagik**, if he was alive. Even though we were far apart, I always felt your love and support.

Being a PhD student is not an easy job especially when you are married. My lovely wife **Ani**, you were the first one among my friends by that time who knew my great news, getting a PhD position at the University of Trento. Ever since I continuously felt your support to me. You know the best that I experienced many challenges throughout this journey. My cordial thanks to you for always inspiring me, for your love, your patience and for just being with me. And last but not least, I would like to acknowledge my son **Gagik**, who is the greatest achievement during my PhD adventure. I want to thank him for empowering me with his hugs and smiles to work hard to complete my PhD study and to be hopeful for a bright future.

Trento, 8 May 2019 D. H.

Abstract

Recent developments in mobile networks towards the fifth generation (5G) communication technology have been mainly driven by an explosive increase in mobile traffic demand and emerging vertical applications with their diverse Quality–of–Service (QoS) requirements, which current mobile networks are likely to fall short of satisfying. New technological cost–efficient solutions are, therefore, required to boost the network capacity and advance its capabilities in order to support the QoS requirements of, for example, enhanced mobile broadband services and the ones requiring ultra–reliable low latency communication. Network densification is known to be as one of the promising approaches aiming to increase the network capacity. This is achieved thanks to aggressive frequency reuse at small cells. Nonetheless, this entails performance degradation especially for cell–edge users due to a high Inter–cell Interference (ICI) level.

Cloud Radio Access Network (C–RAN) architecture has been proposed as an efficient way to address the aforementioned challenges, tackle some of the problems persistent in the present–day mobile networks (e.g., inefficient use of frequency bands, high power consumption) and, by employing virtualization techniques, facilitate the network management while paving a way for new business opportunities for mobile virtual network operators. The main idea behind C–RAN is to decouple the radio unit of a base station, referred as a Decentralized Unit (DU) from the baseband processing unit, referred as a Centralized Unit (CU) and virtualize the latter in a centralized location, referred as a CU pool. Then, by executing so–called "functional split" in the RAN protocol stack between the CU and the DU, identify the RAN functionalities that are to be performed at the DU and the CU pool. Depending on the selected functional split (i.e., the resource centralization level), the bandwidth and latency requirements vary in the fronthaul network, which is the one interconnecting the DU with the CU pool. This results in a different level of resource centralization benefits. Thus, an inherent trade–off exists between resource centralization benefits and fronthaul requirements in the C–RAN architecture.

C–RAN, although provides numerous advantages, raises a series of challenges one of which, depending on the functional split option, is huge fronthaul bandwidth requirement. Optical fiber, thanks to its high bandwidth and low latency characteristics, is perceived to be the most capable fronthauling option; nevertheless, it requires a huge investment. Fortunately, recent advancement in the Millimeter Wave (mmWave) wireless technology allows for multi–Gbps transmission over the distance of one kilometer, therefore, making it a good candidate for the fronthaul network in an ultra–dense small cell deployment scenario.

In this doctoral dissertation, we first study the trade–offs between different functional splits, considering the mmWave technology in the fronthaul network. Specifically, we formulate and solve a Virtual Network Embedding (VNE) problem that aims at minimizing the fronthaul bandwidth utilization along with the number of active mmWave interfaces, and therefore, also minimizing the power consumption in the fronthaul network for different functional split scenarios. We then carry out a relative comparison between the mmWave and optical fiber fronthauling technologies in terms of their deployment cost in order to ascertain when it would be economically more efficient to employ mmWave fronthaul instead of optical fiber.

Different functional splits enable the Mobile Network Operators (MNOs) to harvest different level of resource centralization benefits and pose diverse fronthaul requirements. There is no one–fits–all functional split that can be adopted in C–RAN to cope with all of its challenges since each split is more appropriate to be employed in a specific scenario in comparison with the others. Thus, another problem is to select the optimal functional split for each small cell in the network. This is a non–trivial task since there are a number of parameters to be taken into account in order to make such a choice. To this end, we developed a set of algorithms that dynamically select an optimal split option for each small cell considering ICI level as the main criterion. The dynamic functional selection approach is motivated by the argument that a single static functional split is not a viable option especially in the long run. The proposed algorithms provide MNOs with various options to select between promptness, solution optimality, and scalability.

After having thoroughly analyzed the C–RAN architecture along with the pros and cons of different functional split options, the main objective for MNOs, who already own mobile network infrastructures and want to migrate to the C–RAN architecture, would be to accomplish such a migration with minimal investments. We developed an algorithm that aims at reducing the required investments by reusing the available infrastructure in the most efficient way. To quantify the economic benefit in terms of Total Cost of Ownership (TCO) savings, a case study is carried out considering a small cluster of an operational Long Term Evolution Advanced (LTE–A) network in the simulation, and the proposed infrastructure–aware C–RAN migration algorithm is compared with its infrastructure–unaware counterpart. We also evaluate the multiplexing gain provided by the C–RAN in a specific functional split case and draw a comparison with the one achievable in traditional LTE networks.

Key words: 5G, Cloud–RAN, Multiplexing Gain, Functional Split, Flexible Functional Split, DU–CU Mapping, CU Placement, Virtual Network Embedding, Inter–cell Interference, Wireless Fronthaul, mmWave.

Contents

List of Figures

List of Tables

Acronyms

3GPP Third Generation Partnership Project

ACK ACKnowledgment

API Application Programming Interface

ARPU Average Revenue Per User

AZ Availability Zone

BBU Baseband Unit

BS Base Station

BTS Base Transeiver Station

eNB evolved NodeB

CAPEX CAPital EXpenditures

CCI Co–channel Interference

CP Cyclic Prefix

CPU Central Processing Unit

CPRI Common Public Radio Interface

CSI Channel State Information

CU Centralized Unit

CWDM Course Wavelength Division Multiplexing

CoMP Coordinated Multi–Point

C–RAN Cloud Radio Access Network

DPS Dynamic Point Selection

DU Decentralized Unit

DWDM Dense Wavelength Division Multiplexing

D–RAN Distributed Radio Access Network

ETSI European Telecommunications Standards Institute

FDD Frequency Division Duplex

FEC Forward Error Correction

FSAN Full Service Access Network

FSO Free Space Optics

GPON Gigabit Passive Optical Network

GWCN GateWay Core Network

HARQ Hybrid Automatic Repeat reQuest

I/Q In–phase / Quadrature

ICI Inter–cell Interference

ILP Integer Linear Programming

IT Information Technology

IP Internet Protocol

ITU International Telecommunication Union

IaaS Infrastructure–as–a–Service

InP Infrastructure Provider

IoT Internet of Thing

JR Joint Reception

JT Joint Transmission

LTE Long Term Evolution

LTE–A Long Term Evolution Advanced

LoS Line–of–Sight

MAC Medium Access Control

MCS Modulation and Coding Scheme

MIMO Multiple–Input Multiple–Output

MIP Mixed Integer Programming

MKP Multiple Knapsack Problem

MEC Multi–access Edge Computing

MOCN Multi–Operator Core Network

MTC Machine Type Communication

MVNO Mobile Virtual Network Operator

mmWave Millimeter Wave

NFV Network Function Virtualization

NGFI Next–Generation Fronthaul Interface

NGMN Next Generation Mobile Networks

NSaaS Network–Slice–as–a–Service

OFDMA Orthogonal Frequency–Division Multiple Access

OLT Optical Line Terminator

OMT Optimization Modulo Theories

ONF Open Networking Foundation

ONU Optical Network Unit

OPEX OPErational EXpenditures

OTN Optical Transport Network

OTT Over–The–Top

PDCP Packed Data Convergence Protocol layer

PRB Physical Resource Block

PTP Precision Time Protocol

PBCH Physical Broadcast Channel

PDCCH Physical Downlink Control Channel

QoE Quality of Experience

QoS Quality of Service

RAN Radio Access Network

RANaaS RAN–as–a–Service

RCC Radio Cloud Center

RLC Radio Link Control

RRC Radio Resource Control

RRH Remote Radio Head

RRU Remote Radio Unit

RTT Round Trip Time

RoE Radio over Ethernet

SCF Small Cell Forum

SDN Software–Defined Networking

SFC Service Function Chain

SINR Signal to Interference plus Noise Ratio

SLA Service Level Agreement

SMT Satisfiability Modulo Theories

SON Self–Organizing Network

TCO Total Cost of Ownership

TDM Time Division Multiplexing

TTI Transmission Time Interval

UDP User Datagram Protocol

VNE Virtual Network Embedding

VNF Virtual Network Function

VNP Virtual Network Provider

VNR Virtual Network Request

VoIP Voice over IP

WDM Wavelength Division Multiplexing

WDM–PON WDM Passive Optical Network

1 Introduction

This chapter unveils the motivation and the objective of the work presented in this dissertation, states the problems and the methodology undertaken to solve the problems and outlines the main results. We first detail the requirements of the fifth generation mobile networks (5G), describe the approaches to meet these requirements and identify the challenges associated with the approach adopted in this work. We then summarize the optimization–based approach that we apply to tackle the optimal Cloud Radio Access Network (C–RAN) design and exploitation problems and highlight the main contributions of this work. Finally, we outline the structure of the dissertation and preview the contents of the subsequent chapters.

1.1 Motivation and Objectives

Human–technology interaction has been rapidly increasing, making profound improvements in many lives. Indeed, due to recent technological developments, many more smart devices have become an integral part of humans' lives. As a consequence, mushrooming number of internet–capable devices and services in various spheres of life have been continuously driving daily mobile data traffic to an unprecedented level [1, 2]. According to Cisco, the global mobile data traffic is expected to increase from 7 exabytes in 2017 up to 49 exabytes in 2021, while Ericsson's forecast that by 2022 the global mobile data traffic will reach up to 71 exabytes out of which around 75% accounting for video traffic. Thus, on the one hand, mobile data traffic demand has been abruptly growing, forcing Mobile Network Operators (MNOs) to increase the system capacity in order to accommodate the traffic demand. On the other hand, the baseband processing resources and the frequency resources of the present–day mobile networks are not used efficiently [3]. This is because MNOs allocate resources to their Base Station (BS) (Base Transeiver Station (BTS) in 2G, NodeB (NB) in 3G and evolved NodeB (eNB) in 4G) in such a way as to be able to meet the peak hour traffic demand, thereby entailing frequent underutilization of these resources due to spatio–temporarily fluctuating traffic.

Recent advancements in mobile networks have also paved a way for new emerging applications such as e–health, massive Internet of Thing (IoT), Machine Type Communication (MTC), self–driving cars, augmented reality, etc. These applications have diverse Quality of Service (QoS) requirements which pose several challenges to the mobile networks [4].

Compared to Long Term Evolution (LTE – 4G) mobile networks, in the fifth generation (5G) mobile networks, whose standalone specifications (Release 15) have been approved in the mid of 2018 by the Third Generation Partnership Project (3GPP) [5], it is expected to deliver a thousand times increase in the system capacity, a ten times increase in the transmission data rate, sub–millisecond latency, enhanced cell–edge users' performance, etc [6]. Several approaches, such as traffic offloading, Multi–access Edge Computing (MEC), network densification, centralization and virtualization of Radio Access Network (RAN) resources (C–RAN), additional spectrum usage, exist in order to meet the expectations of 5G networks. Among these approaches, network densification along with C–RAN deployment is gaining ground as a promising way to tackle some of the challenges of 5G networks, while using the network resources efficiently. Indeed, small cells are widely deployed by MNOs in order to boost the system capacity, while alleviating the load of macro BSs. Its vaunted benefits include the high frequency reuse factor, reduced transmit power and path loss, and decreased distance between BSs and users, which enables users to employ higher order Modulation and Coding Scheme (MCS). Its main drawback, however, lies in the fact that it dramatically increases the level of Inter–cell Interference (ICI), which may significantly degrade users' Quality of Experience (QoE) unless interference mitigation techniques are used.

C–RAN concept has been proposed in order to address the aforementioned drawback of small cells while using scarce and precious frequency bands in an efficient manner [7]. In the traditional RAN, which is the part between users and BSs, a BS is considered as a single entity, and the entire baseband signal processing is taking place at BSs. Conversely in the C–RAN architecture, a BS is decomposed into a Decentralized Unit (DU) and a Centralized Unit (CU), which are interconnected via fronthaul network. CUs of different BSs can be virtualized and centralized in a CU pool. Which part of baseband signal processing where is performed (in the DU or in the CU pool) is completely dependent on the chosen functional split in the RAN protocol stack functionalities between the DU and the CU. The C–RAN[1] architecture, although yields numerous benefits, as will be detailed in Section 2.2.3.1, raises several challenges such as optimal functional split selection between the DU and CU, stringent fronthaul bandwidth and latency requirements, and the fronthaul network cost.

Given the aforementioned challenges, the goal of this dissertation is to study the trade–offs between different functional splits in C–RAN and develop optimization–based algorithms for efficiently exploiting C–RAN resources while optimizing different aspects of mobile networks (e.g., energy consumption, ICI) in different functional split scenarios. Moreover, we aim to coin a realistic mobile network deployment scenario and devise an algorithm that can dynamically select the optimal functional split option based on some criteria (e.g., interference level). Lastly, we intend to conduct a case study considering an operational mobile network and develop an algorithm that can enable MNOs to transit from their legacy networks to C–RAN with minimal investment.

[1]The term "C–RAN" is used throughout this dissertation to refer to the RAN architecture, unless otherwise specified.

1.2 Problem Statement

In the C–RAN architecture, the lower is the functional split execution point in the RAN protocol stack, the more are the benefits of resource centralization and virtualization at the CU pool; nonetheless, the more stringent are the fronthaul bandwidth and latency requirements in the fronthaul network. These requirements proclaim the expensive optical fiber as the most suitable fronthaul medium thanks to its high bandwidth and low latency characteristics. However, since in the course of time mobile networks are becoming denser due to the proliferation of small cells, providing fiber–based fronthaul links to all small cells in an ultra–dense network cannot simply be a viable option for MNOs due to its huge cost. Therefore, the fronthaul network of C–RAN is likely to leverage on optical fiber as well as other fronthaul mediums and technologies such as wireless links, ethernet links, coaxial cables, etc. These fronthaul mediums, however, require meticulous analysis on their bandwidth and latency in order to ascertain their applicability to different functional split scenarios. **Thus, one of the problems is to find cost–efficient fronthauling mediums and technologies that can be alternatively used in the fronthaul network of C–RAN in different functional split scenarios.**

As it will be detailed in Section 3.5, a number of functional splits exist within the RAN protocol stack between the DU and the CU. Each of these splits has its pros and cons and is characterized by its fronthaul bandwidth and latency requirements [8]. For example, the split in which the entire RAN protocol stack is centralized at the CU pool (i.e., PHY–RF split) enables MNOs to employ advanced inter–cell coordination techniques such as Joint Transmission (JT)/Joint Reception (JR) [9] and has the maximum allowed fronthaul latency of 250 μs. Whereas, the split in which the Packed Data Convergence Protocol layer (PDCP) and the layers above are centralized at the CU pool, although may accept 30 ms of fronthual delay [8], can no longer support the aforementioned techniques. Thus, from the aforementioned arguments, it can be deduced that depending on the specific scenario, a particular split may be more suitable/beneficial to be employed compared to the other splits. Therefore, the flexibility in selecting a functional split option for a small cell, in our opinion, is of paramount importance in order to reap the benefits of all functional splits while ensuring that all the requirement of the selected functional split are satisfied and the network resources are used in an efficient manner. However, a functional split selection is a non–trivial problem, since a vast number of parameters (e.g., mobile data traffic demand, ICI scenario, fronthaul capacity and latency) have to be considered in order to select the optimal split point. **Thus, another problem is to develop an algorithm that enables dynamic functional split selection at small cells based on a certain criterion such as ICI.**

There are many organizations and standardization bodies, such as 3GPP [10], Small Cell Forum (SCF) [8], Next Generation Mobile Networks (NGMN) [11], Next–Generation Fronthaul Interface (NGFI) [12], working on various functional split options and fronthaul technologies. Only after thoroughly investigating different functional split options, their deployment scenarios and the applicability of different fronthaul mediums, MNOs would be willing to adopt the C–RAN architecture. Ubiquitous deployment of the C–RAN architecture with different

functional splits is a long–lasting process that requires huge investments. Therefore, the main concern for the MNOs that currently own mobile network infrastructure would be to adopt the C–RAN architecture with minimal investments. **Thus, the last problem that we tackle in this work is to develop an algorithm that enables MNOs to realize the migration from the legacy RAN architecture to the C–RAN architecture in a cost–efficient manner.**

All the aforementioned problems are modeled as Virtual Network Embedding (VNE) problems, which are formulated and solved using Integer Linear Programming (ILP) techniques. Heuristic algorithms are also proposed to tackle the scalability issue of the ILP–based algorithms. The following section details the methodology adopted to solve the described problems.

1.3 Methodology

Recent advances in Network Function Virtualization (NFV), a technology that is expected to play a key role in 5G networks, enables MNOs, who own mobile network infrastructures, also called Infrastructure Provider (InP), to share their resources (e.g., RF bands, RF front–ends) among, for example, multiple Mobile Virtual Network Operator (MVNO)[2]s, such as Mint Mobile [13], Affinity Cellular [14], who can make Virtual Network Request (VNR)[3]s, specifying their desired characteristics for the nodes (e.g., a base station) and the links in the request.

Efficient mapping of a VNR onto a substrate network is known as a VNE problem [15]. Upon receiving a VNR, the InP has to decide whether to accept and map the request or to reject it. The VNE problem is *NP*–hard (refer to [16] for the proof) and has been extensively studied in the literature [17, 18, 19]. The embedding process is composed of two steps: the node embedding and the link embedding. In the node embedding step, each node in the VNR is mapped to a substrate node, while in the link embedding step, each virtual link is mapped to a single substrate path. In both steps, nodes and links constraints must be satisfied.

ILP techniques are used to formulate the VNE problems, which are then solved using Matlab (intlinprog [20]) and CPLEX [21] solvers. In an ILP problem, there is a linear cost function, and one or more constraints. The cost function can model different aspects of the network, such as energy consumption, link usage or interference level, which are to be optimized; whereas, the constraints can model the given network characteristics, such as the maximum capacity of the links or the maximum number of antennas at BSs, and are used to set the conditions that have to be respected in order to find a valid solution. An algorithm such as Branch & Bound (B&B) or Branch & Cut (B&C) can be used by the solvers to minimize or maximize the cost function. B&B algorithm, for example, uses linear relaxation of a solution to find the best achievable cost. If the mapping variable contains a fractional value, the algorithm divides the problem into subproblems (branches in the search space tree) and checks the lower and upper integer values of that fractional solution. As soon as an integral solution is found on a branch, no

[2]MVNOs are wireless communication service providers that own no mobile network infrastructures and, therefore, exploit other MNOs' infrastructure to provide their services.

[3]VNRs are virtual networks, also called slices in mobile networks, with certain characteristics (e.g., number and location of base station, RF bandwidth per base station) that can be requested by, for example, MVNOs.

further branching is done on that branch and its cost is compared with the cost of the other branches. If the cost estimate on the other branches is lower than the best solution found so far then those branches are pruned; otherwise, if a better solution (with higher cost) is found, the best feasible solution is updated. This process is continued until all the branches are checked. As opposed to B&B algorithm, B&C used in most of existing ILP solvers continuously adds valid inequality constraints (cuts) to the ILP problem seeking to find an integer solution. While the cuts added by B&C usually reduce the time required to solve most of the problems, on occasion, they may have the reverse effect.

The issue with the aforementioned algorithms that are used to solve the ILP–based problems is that they become computationally intractable when the problem size increases (e.g., the substrate and/or the virtual network size increase) in the VNE problem. In order to address this issue, we resort to heuristic solutions, employing Dijkstra's shortest path algorithm. We then assess the efficiency of the suboptimal embedding solutions found by the heuristics by comparing them with the optimal solutions found by the ILP–based algorithms.

It is worthwhile to mention that apart from the ILP–based and heuristic algorithms, the optimization problems described in Section 1.2 can be formulated and solved by using, for example, Optimization Modulo Theories (OMT) [22] techniques, in which the main principle is to solve Satisfiability Modulo Theories (SMT) problems and update the cost–related terms of the SMT formula until no better solution is found.

1.4 Main Contribution

The contributions provided by this dissertation can be broadly divided into three parts, one contribution per problem highlighted in Section 1.2.

- Recent advances in the Millimeter Wave (mmWave) wireless communication in the E–Band (70–80 GHz) enables a few Gbps of bandwidth to be carried up to one Km of distance. This makes mmWave wireless technology a promising alternative to the optical fiber to be used in the fronthaul network of C–RAN in a dense mobile network deployment scenario.

  Our first contribution, which yielded the following publications [23, 24, 25], is that we study the applicability of mmWave links in the C–RAN architecture with different functional splits, and its economic efficiency compared to optical fibers to be used in the fronthaul network. In particular, we tackle the problem of efficient utilization of mmWave fronthaul links having an overarching goal of reducing the network–wide energy consumption. This problem is formulated and solved considering different mobile network topologies, having different functional split options. Additionally, we examine the trade–offs between the considered functional splits, and carry out a comparative analysis on the fronthaul network deployment cost between optical fiber and mmWave wireless fronthaul links.

- Mobile data traffic keeps snowballing with new emerging applications and massive device connectivity that require an internet connection to interconnect with each other. Given the formidable traffic increase forecast [1], in our opinion, implementing a single functional split option in the C–RAN architecture is not a viable solution in the long run. We, therefore, believe that the possibility of dynamically selecting the optimal functional split is of utmost importance in order to garner the benefits of all of the selected functional splits while efficiently employing the network resources.

  Our second contribution, which yielded the following publications [26, 27], is that we develop a set of algorithms that enable flexible functional split selection at the small cells. Specifically, the most optimal functional split option at the considered moment is selected based on the estimated ICI level. The rationale behind opting ICI as the main criterion for functional split selection is its expected growth with the proliferation of small cells, while the appropriate split option selection enables the use of advances techniques/algorithms seeking to suppress the ICI. Additionally, we demonstrate the benefits of the proposed flexible functional split approach by drawing a comparison with the classical PHY–RF static split, considering different fronthaul network topologies.

- There are a plethora of studies conducted on the C–RAN architecture. Nevertheless, to date, the vast majority of these studies has tended to focus on building future C–RANs from scratch based on the traffic demand and some other parameters. A common characteristic of the state–of–the–art studies presented in Section 3.3 is that none of them has studied how MNOs, owning legacy LTE networks, can upgrade their RAN adopting the C–RAN architecture with the minimal cost by employing the available site locations, transmission links, the knowledge of the CU pool candidate locations, and hourly traffic demand statistics per cell. This is a critical problem since MNOs main concern pertaining to the adoption of C–RAN is its huge deployment cost.

  In the light of the aforementioned argument, as our last contribution [28], we conduct a case study, considering an operational LTE–A network, and proposing an approach that enables MNOs to employ the legacy mobile network architecture while transiting to C–RAN, thereby curtailing the required investments. More specifically, we show the CAPital EXpenditures (CAPEX) and OPErational EXpenditures (OPEX) savings that can be obtained in the infrastructure–aware C–RAN migration scenario compared to the infrastructure–unaware one in which the C–RAN has to be designed from scratch without the use of the legacy network infrastructure.

1.5 Outline of the Dissertation

This dissertation consists of seven chapters, which are briefly summarized as follows. In the current chapter, we first motivate our work and present its objectives. We then state the problems and discuss the methodology undertaken to solve the problems. Lastly, we highlight the contributions of our work.

Chapter 2 provides background on the virtualization techniques expected to be vastly applied in 5G networks. It then details the RAN architecture evolution starting from the traditional RAN reaching up to C–RAN. Since the C–RAN architecture is in the focus of our work, this chapter also discusses the advantages and more importantly the challenges of C–RAN deployment some of which are targeted in our work.

Chapter 3 walks through the state–of–the–art studies on Virtual Network Function placement, Virtual Network Embedding, CU placement and on mobile network sharing problems. The functional split concept in C–RAN is then introduced presenting the key findings of the most promising split options that are in the focal point of research by the industry and the academia. Additionally, several fronthaul technologies/protocols for different splits are discussed.

In Chapter 4, we investigate the use of mmWave technology in the fronthaul network in different functional split scenarios. Specifically, we first motivate the need for other fronthauling medium/technologies, apart from optical links, which are then analyzed in terms of their deployment cost. We then present the ILP–based formulation and the solution to a CU placement problem whose purpose is to minimize the fronthaul bandwidth utilization, and therefore, also minimize the network–wide power consumption at the C–RAN employing various functional splits. We also develop a scalable dynamic placement heuristic to tackle the scalability issue of the ILP formulation, solving the placement problem for large–scale networks. After conducting an extensive simulation, we summarize the results and draw conclusions on the mmWave technology as a candidate to be used in the fronthaul network for different functional split scenarios.

In Chapter 5, we further elaborate on the functional split selection problem and justify the need for flexible functional split, which is then showcased in a mobile network deployment scenario. We present dynamic and static CU placement problems and propose ILP–based and heuristic CU placement algorithms in order to solve the problems having an ultimate goal of minimizing the network–wide ICI and the fronthaul bandwidth utilization by dynamically selecting the optimal functional split option for each small cell. We then present the results of the numerical evaluation highlighting the effect of flexible functional split. Chapter 5 is concluded by discussing the benefits and challenges posed by the flexible functional split selection in comparison with the static PHY–RF split.

In Chapter 6, we conduct a case study considering a commercial LTE–A mobile network and studying the problem of transiting from the legacy RAN to C–RAN with minimal investment. This goal is achieved by formalizing and solving an ILP–based DU–CU mapping problem that computes the optimal number of CU pools, determines their location and re–uses the available backhaul transport network infrastructure in the fronthaul network of C–RAN. By considering the classical PHY–RF split in the C–RAN, we quantify its multiplexing gain obtained by employing an inter–sector intra–carrier traffic aggregation technique and compare it with the one achievable in the traditional mobile network by employing an intra–sector inter–carrier traffic aggregation technique. Additionally, we compare the proposed infrastructure–aware C–RAN migration approach with the infrastructure–unaware one, showing the economic

advantage of the former approach in terms of CAPEX and OPEX savings. Lastly, we further discuss both traffic aggregation techniques with a particular focus on their pros and cons.

Finally, in Chapter 7, we recap the contributions of this dissertation and summarize the obtained results. After the conclusions, we also highlight several promising research directions for future work.

2 Background

RAN part of mobile networks has evolved tremendously over the last few years with the recent proposals suggesting virtualization of its functionalities. This chapter introduces the role of virtualization in RAN and presents the evolution of RAN architectures towards C–RAN. Specifically, we present the concepts of Software–Defined Networking (SDN) and NFV and discuss how they can be applied to the RAN of 5G mobile networks. We then detail the traditional RAN architecture and motivate the need for Distributed Radio Access Network (D–RAN). Lastly, having discussed the limitations of the traditional RAN and the D–RAN, we introduce the C–RAN architecture, examining in detail its benefits and challenges some of which are tackled in our work.

2.1 Virtualization (SDN & NFV) in 5G Networks

While virtualization has been extensively applied in the Information Technology (IT) domain, it is relatively new to mobile networks. Nowadays, however, it is considered to be one of the main enablers of 5G networks [29]. Specifically, SDN [30] and NFV [31] technologies, which implement virtualization techniques, are expected to play a key role in 5G networks by supporting different novel scenarios such as RAN slicing and flexible spectrum management. Although these techniques can be used separately, greater benefits can be accrued by combining the use of both techniques [32]. Nguyen et al. provides a comprehensive survey on where SDN and NFV can be applied in the mobile network architecture [33].

SDN facilitates network design, configuration, management and upgrade by separating forwarding and control planes of network devices, which traditionally reside together. While the former remains in the hardware, the latter is shifted to a centralized SDN controller. Centralization of control planes of several network devices in the same controller provides a global network view to the controller and endows it with the ability to centrally program and manage the network services on the adjustable underlying hardware. The interface between the controller and network devices is called the southbound interface, while the northbound interface refers to the communication between the controller and the network applications. OpenFlow [34], which is maintained by the Open Networking Foundation (ONF) [35], is a protocol commonly used in the southbound interface between network devices (e.g., switches) and the

controller, while, for example, REST API can be used to deliver northbound communication between network applications and the controller.

Decoupling control and forwarding planes also spurs realization of multi–vendor RAN deployment by allowing both planes to evolve autonomously and cooperatively, using a standardized southbound interface for their interconnection. Moreover, it inspires innovation in both planes (i.e., in software and hardware) giving MNOs unprecedented real–time programmability and fine–grained control over their RAN infrastructure of 5G networks.

The basic concept in NFV is to virtualize network functions, such as load balancers and firewalls, and migrate them from purpose–build specialized hardwares to general–purpose computing infrastructures such as clouds infrastructures. As a consequence, several logical instances of the Virtual Network Function (VNF)s can co–exist together and use the same hardware equipment, which in turn, would prolong the lifecycle of the hardware relying on software updates rather than hardware ones. Other benefits of NFV include faster time to market, multi-tenancy support, energy consumption reduction, and scalability.

In the traditional RAN, the use of proprietary hardware appliances leads to high CAPEX and OPEX, and inefficient utilization of precious network resources (e.g., frequency band) in the time when their capacity demand is growing in order to meet the ever–increasing mobile traffic requirements. NFV and SDN also play a pivotal in the realization of C–RAN by decoupling baseband processing from its hardware and virtualizing these resources at the CU pool. This enables dynamic and flexible association of basaband resources with any DU whose processing is centralized at the CU pool, therefore, exempting the need for having dedicated and unsharable baseband processing for each DU, like it is the case in the traditional RAN. Moreover, this lowers the RAN cost and enables flexible deployment of services on commodity hardwares instead of specialized ones.

Recently, significant efforts have been invested by European Telecommunications Standards Institute (ETSI) in NFV [31], including also the definition of the use cases for mobile networks [36]. In particular, the use case #6, which suggests BS virtualization, is of interest in our work. It has numerous advantages such as easing network management, providing the possibility for RAN slicing through virtualization of physical RAN resources, load balancing, etc. One of the most prominent advantages of applying NFV in mobile networks is that it opens a door for new business opportunities for MVNOs. We remind the reader that these are the operators who own no mobile network infrastructures and provide mobile services by leasing network resources such as antennas, RF bands, link capacity, from InPs. For instance, MVNOs can request a virtual BS with some characteristics (e.g., bandwidth, antenna configuration) in their desired area. The goal of the InP receiving the request would be to allocate the requested resources by mapping it to the network in the most optimal way, resulting in an efficient utilization of network resources.

2.2 RAN Evolution towards C–RAN

2.2.1 Traditional RAN Architecture

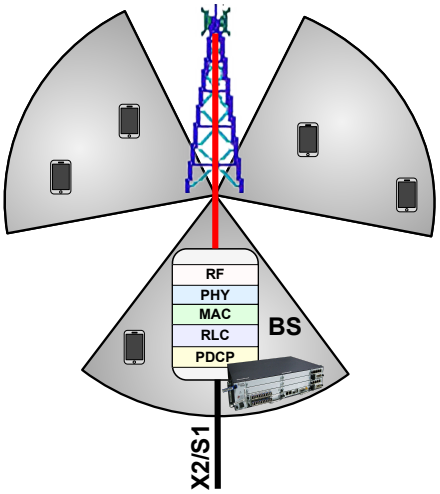
In the RAN of traditional mobile networks, the radio and the baseband processing units, which make up a BS, are co–located at shelters/cabinets deployed at the cell sites, and thick RF coaxial cables, also called feeders, are used to interconnect them with the antennas. Depending on the cell site deployment, the length of these lossy RF coaxial cables can span from 5 meters, in the case rooftop deployments, to 50 meters or even more, in the case of greenfield deployment. This may result in a significant signal attenuation before even starting its actual processing. On the other hand, the frequency bands allocated to the BSs having the described RAN architecture are not used efficiently since MNOs dimension the frequency resources in such a way as to meet the peak traffic demand at 24×7. As a consequence, due to spatio–temporally fluctuating traffic demand, the idle frequency resources are wasted during less–busy periods [3]. Figure 2.1a illustrates the RAN architecture in a traditional BS, where S1 and X1 are the links that connect BSs, respectively, with the core network and with other BSs.

2.2.2 Decentralized RAN Architecture (D–RAN)

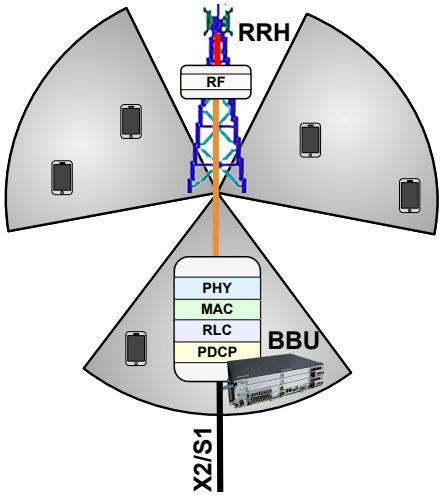
In order to mitigate the high signal losses associated with the RF coaxial cables, the radio processing unit, referred as Remote Radio Head (RRH), and the baseband processing unit, referred as Baseband Unit (BBU), are separated at the BS. While the latter is left at the cabinet, the former is moved closer to the antennas. Additionally, the RF coaxial cables are replaced with optical fiber, and a digital interface with the protocols such as the Common Public Radio Interface (CPRI) protocol [37] is typically used to carry the In–phase / Quadrature (I/Q) signals between the baseband units and the radio elements. This RAN architecture is called D–RAN. As opposed to the traditional RAN in which the distance between the radio and baseband processing units is circumscribed due to significantly high signal losses, this problem does not persist in the D–RAN architecture thanks to the characteristics of the optical fiber. Whereas like the traditional RAN, also D–RAN is characterized by inefficient frequency band and baseband processing resource usage. This is because both architectures are distributed resulting in separate baseband processing at each BS. Therefore, BSs fall short to share the idle resources with neighbor BSs. Figure 2.1b illustrates the D–RAN architecture.

2.2.3 Cloud RAN Architecture (C–RAN)

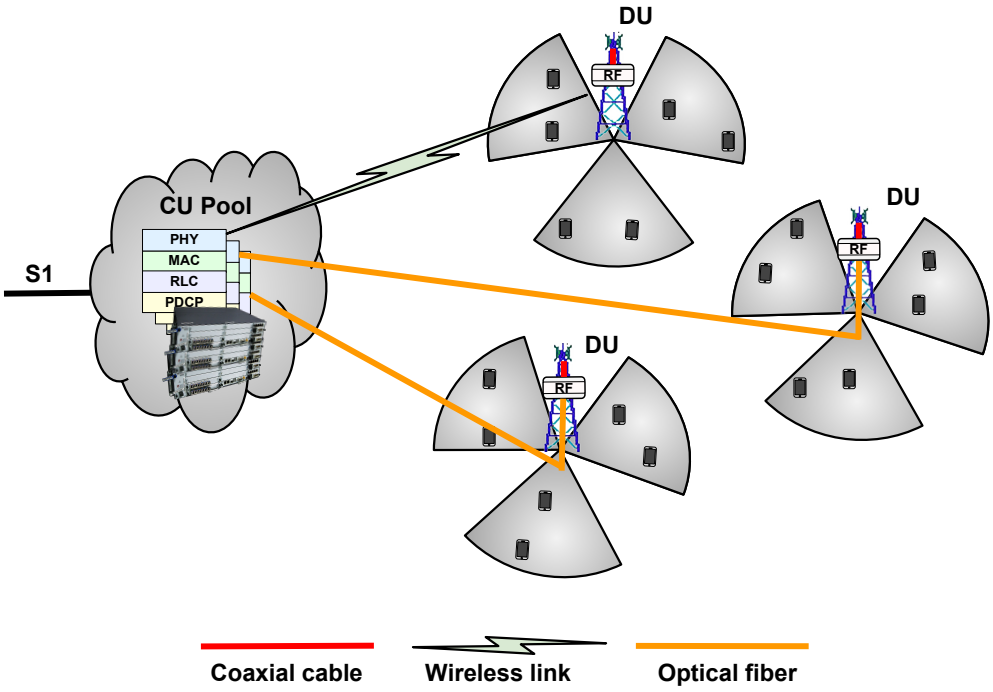
As it has been mentioned in the previous sections, both the traditional RAN and the D–RAN architectures are distributed and are, therefore, characterized by non–optimal use of radio and baseband processing resources. Fully centralized and virtualized RAN (C–RAN) architecture has increasingly attracted MNOs' attention as an approach capable of tackling the aforementioned problem and boosting the system performance. Enormous research efforts have, therefore, been invested in C–RAN by academia and industry [7, 38, 39, 40].



(a) Traditional RAN architecture.



(b) D–RAN architecture.



(c) C–RAN architecture.

Figure 2.1 – Radio Access Network (RAN) evolution in mobile networks.

The C–RAN architecture is depicted in Figure 2.1c. As opposed to D–RAN, in C–RAN, CUs of multiple BSs can be centralized in a CU pool[1], while the fronthaul, which is the link interconnecting the CU pool with DUs, can span up to 40Km [7].

[1] Notice that the 3GPP [10] terminology with a slight modification is used throughout this doctoral dissertation. However, other terminologies such as RRH and BBU pool, Remote Radio Unit (RRU) and Radio Cloud Center (RCC), and RU and DU for DU and CU can be found in the technical documents of, respectively, SCF [8], NGFI [12] and NGMN [11].

2.2.3.1 Advantages of C–RAN

The C–RAN architecture depicted in Figure 2.1c corresponds to the classical PHY–RF functional split since the entire baseband signal processing is performed in the CU pool. Therefore, the term "C–RAN" is often used in the literature to refer to the split option in the C–RAN architecture. The main pros of this particular *C–RAN split* can be summarized as follows:

Easy deployment, energy and cost efficiency: Traditional BS deployments call for a meticulous cell site survey. The physical requirements (e.g., area, cable installation) of traditional BSs confine the candidate locations for them to be deployed, very often urging the site engineers to find new locations. On the contrary, DUs, due to the fact that they perform no baseband signal processing, have many more deployment possibilities. Indeed, being compact devices, DUs can be mounted, for example, on billboards or on lamp posts. This expedites the cell rollout, from initial cell planning until their on–air status. As a consequence, not only does this lower the CAPEX spent on renting cell sites and on labor cost for the site installation and maintenance but it also considerably curtails the electricity bills share in the overall CAPEX since no cooling systems and on–site supporting equipments are required for DUs. Furthermore, due to so–called *tidal* effect, which characterizes spatio–temporal traffic variation at cell sites, aggregation of baseband units (CUs) of DUs at a CU pool reduces the overall number of CUs at the CU pool several times, providing statistical multiplexing gain, which directly translates to CAPEX savings. As a consequence, thanks to the multiplexing gain, the same tasks can be performed with less computational resources. For example, in [41], C–RAN is compared with D–RAN in terms of the number of CUs that would be required to meet the required traffic demand. Considering a certain commercial, office and residential BSs deployment scenario with their given traffic demand, the authors demonstrate that the number of CUs required to meet the traffic demand can be curtailed by four times thereby achieving multiplexing gain.

Efficient link resource utilization: Network densification through small cell deployments is currently adopted by many MNOs as an efficient strategy to boost the mobile network capacity. Due to the small size of the cells, this strategy would force users to more frequently cross the borders of different small cells, resulting in a considerable increase in the signaling overhead. Virtualization and centralization of baseband processing resources has the potential to remedy this situation thanks to a high–bandwidth and low–latency switching matrix in the CU pool that enables fast and efficient signaling/information exchange among the CUs that belong to the same CU pool, without consuming link resources i.e., S1 or X2 link resources, like in the traditional network.

Improved spectral efficiency: In C-RAN, CUs of various BSs (macro, micro or small) are aggregated in a large CU pool where they can easily share signaling, data and Channel State Information (CSI) for active users in the system. With C-RAN, it is much easier to implement algorithms to mitigate ICI and improve the spectral efficiency [42]. For instance, Coordinated Multi–Point (CoMP) algorithms, can be easily implemented within the C-RAN infrastructure.

Adaptability to non-uniform traffic: Pooling baseband processing resources in a CU pool enables MNOs to dynamically add/remove baseband processing units (CUs) based on their needs. Moreover, in the C–RAN architecture, there is no need for having a dedicated CU for each DU since the virtualization allows baseband resources to be shared among different DUs. C–RAN can, therefore, effectively adapt to non–uniform traffic by balancing it across different CUs in the CU pool.

Ease of network management and extension: C–RAN eases and expedites network upgrade and extension through centralized control. Moreover, it facilitates centralized management and operation since its architecture consolidates the CUs and site support equipments in the CU pool. Furthermore, it provides the possibility of exploiting complex multi–cell coordination techniques such as JT/JR and Dynamic Point Selection (DPS) [9]. Additionally, C–RAN eliminates the need for maintaining X2 interfaces among the small cell that are controlled by the same CU pool.

2.2.3.2 Challenges of C–RAN

Although fully centralized C–RAN yields many benefits as discussed in Section 2.2.3.1, it poses several challenges on the fronthaul links. Specifically, C–RAN sets huge fronthaul bandwidth requirement, which depends on a vast number of parameters such as the number of carriers at a cell, the number of antennas and employed LTE bandwidth. For instance, CPRI rate of ≈ 2.5 Gbps is required on the fronthaul links in order to carry signals over a 20 MHz LTE Frequency Division Duplex (FDD) channel using a $2x2$ Multiple–Input Multiple–Output (MIMO) antenna configuration. This enormous fronthaul bandwidth demand is due to the fact that raw IQ samples without any baseband processing are transmitted on the fronthaul link. Since the overall signal is forwarded in time-domain, the fronthaul rate is static and does not depend on the actual user traffic. Thus, even if there is no user attached to the cell, given the system setup, the fronthual link has to support certain bandwidth.

Yet another challenge imposed on the fronthaul link is its tight latency requirement. There are several timers defined by 3GPP within the RAN protocol stack in the Long Term Evolution (LTE) standard. These timers will ultimately define the maximum latency requirement that has to be fulfilled in order to meet the specifications of the standards. Among them, Hybrid Automatic Repeat reQuest (HARQ) procedure in the RAN protocol stack is the one posing the most stringent latency requirement. In LTE standards, it requires a positive/negative ACKnowledgment (ACK) to be received in the uplink direction within 4ms. Subtracting the signal processing time, the rest of the delay budget translates to the maximum admissible fronthaul distance of 40Km. Several approaches such as HARQ interleaving [43] exist in order to extend the fronthaul delay; however, it is achieved at the expense of a reduced peak data rate.

3 State of The Art

This chapter surveys the state–of–the–art studies conducted on the VNF placement problem, on the VNE problem, on some options for mobile network sharing and on the C–RAN architecture with different functional splits. Specifically, we first examine the VNF placement problem, providing the results which are of interest for our work. We then detail the key findings in the VNE placement problem, highlighting the main differences with our study. This is followed by a discussion on some of the mobile network sharing options. Lastly, we investigate several functional split options along with some potential fronthaul transport technologies and protocols.

3.1 Virtual Network Function Placement

Decoupling network functions such as load balancers, intrusion detection systems, firewalls from their specialized, purpose–built hardwares, NFV has the potential of curtailing CAPEX and OPEX of the network, easing the network upgrade and management while still providing a satisfying performance of the VNF [31]. While in the literature the terms VNF placement and VNE placement are often considered as synonymous, in this chapter, the VNF placement in used to refer only to the individual VNF placement. VNF placement conceptually resembles virtual machines placement in Cloud Providers' (CPs) network such as in Amazon AWS [44] and Microsoft Azure [45]. For example, clients can request one or more Elastic Compute Cloud (Amazon EC2) instances in their desires geographical areas where Amazon has data centers, called Availability Zone (AZ) in Amazon's terminology, specifying the desired Service Level Agreement (SLA). Moreover, clients are allowed to flexibly change the parameters of their requested VNFs according to their needs. One of the most attractive options for the CPs' clients is the pay–as–you–go pricing, which, as the name suggests, implies that the clients pay only for the network resources, such as memory, Central Processing Unit (CPU), storage, that are actually consumed by their VNFs/services. Therefore, the vertical/horizontal scaling of the VNF parameters, while ensuring optimal resource usage of the resources, is of great importance for CPs.

There is a vast array of work studying the VNF placement [46, 47, 48, 49, 50] problem. A survey on resource management in cloud computing environments can be found in [51]. In [46], the authors study the problem of placing virtual machine instances on physical containers in

such a way as to reduce communication overhead and latency. In [47], the authors propose a novel design for a scalable hierarchical application components placement for cloud resource allocation. The proposed solution operates in a distributed fashion, ensuring scalability while providing performances very close to that of a centralized algorithm. This work is extended in [48] where several algorithms for efficient data management of component-based applications in cloud environments are proposed. In [49], the elasticity overhead, and the trade–off between bandwidth and host resource consumption are jointly considered by the authors when formulating a VNF placement problem. An elastic VNF placement problem is introduced in [50]. The idea is to enable horizontal in/out scaling of the VNFs by installing/removing new VNF instances, migrating a VNF instance or reassigning a part of the workload to another VNF instance in response to new service requests and/or workload variations.

3.2 Virtual Network Embedding

VNF(s) can be requested as an individual network function or as a chain that forms a full service. Each of these services can be composed of different kinds and numbers of VNFs that are interconnected in a particular order, also known as a Service Function Chain (SFC). A client, for example, can request an SFC in which a WiFi hotspot is connected to a firewall that, in turn, is linked to a load balancer [52]. The embedding of an SFC in the InP's networks is known as VNE problem [15]. A huge body of work has been published on the VNE problem. Studies in this domain can be mainly classified into the following groups: online and offline VNE, coordinated and uncoordinated VNE, intra–domain and inter–domain VNE and VNE with static and dynamic resource allocation. A VNE algorithm can fall into more than one group. For example, a VNE algorithm can be online, coordinated with dynamic resource allocation.

Online and Offline VNE. VNRs dynamically arriving over time can be either unknown (online) or known (offline) prior to arrival. The amount of literature published on both online VNE [53, 54, 55] as well as on offline VNE [56, 57, 58] is considerable. In the online VNE, the network should be responsive by reactively handling VNRs. In most of the real–life situations, VNE has to be tackled as an online problem for VNRs with an arbitrary amount of lifetime. The authors of [53] present a path–based VNE model, which is based on the Primal–Dual analysis. The proposed approach achieves an optimal or a near–optimal solution as a result of an iterative process based on feedback between node and link mapping sub-problems. Online VNE algorithms are also proposed in [54]. The algorithms make use of backtracks to execute the VNE in best–fit sub–substrate networks. Specifically, for each VNR, the algorithms build a set of candidate sub–substrate networks out of the substrate nodes and links that are able to meet the minimum requested resources and have not been used in other sub–substrate networks.

Although there are just a few use cases for offline VNE, it is still of importance to be considered. For example, a Voice over IP (VoIP) service provider may require an end–to–end connectivity between central hubs, specifying their desired locations prior to launching their service. This provides InPs an opportunity to efficiently utilize network resources by planning the VNE a priori. An offline VNE problem is studied in [58]. The authors propose two variant of the VNE problem: VNE with reconfiguration and VNE without reconfiguration. The former refers to the case where the VNE can be changed, while the latter implies that the VNR embedding is for the lifetime and, therefore, cannot be changed. In order to reduce congestion and to avoid hot spots, a node/link stress ratio is defined, and the algorithms aim to simultaneously minimize the maximum of this ratio in the substrate network. A single–path and a multi–path routing schemes are examined in [56] for an offline VNE problem. The authors employ the rent–at–bulk approach, which provides discounts if physical resources are rented in bulks. Extending their work, the same authors propound exact and heuristic approaches for an offline VNE problem in [57]. In order to maximize the InPs' profit, a probabilistic model is introduced, exploiting the fact that the allocated resources are not always fully utilized.

Coordinated and Uncoordinated VNE. Coordination plays an important role in solving VNE problems. In terms of coordination, VNEs can be grouped into coordinated [59, 60, 17, 61] and uncoordinated [62, 63, 64] embeddings. In the coordinated VNE, as the name suggests, there is a coordination between the node mapping and the link mapping stages. This kind of one–stage optimization approach results in more efficient mapping solutions. A coordinated VNE problem is formulated and solved in [59]. The authors propose EdWiN algorithm that supports large lookahead values by simultaneously mapping multiple requests. VNetMapper is proposed as a coordinated VNE algorithm [17]. Different from many studies that resort heuristic algorithms in order to solve the VNE problem in practical–sized networks in a reasonably short time, VNetMapper formulates the embedding as a Mixed Integer Programming (MIP) problem and finds its optimal embedding solution within granularity of a few seconds. Security of a virtual network is of paramount importance for InPs in real–life VNE scenarios. The authors of [65] propose a comprehensive framework for the security–aware VNE problem. In their framework, VNRs can request a security plan, specifying parameters for different sort of security attacks for different levels (e.g., data center, node and link level attacks).

The uncoordinated VNE implies no coordination between the node and the link mapping stages. As a consequence, this leads to a sub–optimal embedding solution due to restricted solution space. In this case, the results of the virtual node mapping serve as an input to the virtual link mapping function. Early studies on VNE rely on the uncoordinated embedding of nodes and links. Aiming at minimizing energy consumption in InPs' network, the authors of [62] propose exact and uncoordinated heuristic algorithms to solve the VNE problem. The main intuition behind the algorithms is to reuse the working nodes while increasing the possibility of accepting VNRs, thereby producing more revenues for InPs. For the heuristic algorithm, the best–fit approach is used to map virtual nodes; whereas the worst–fit approach is used for the virtual link embedding, employing k–shortest–path algorithm. Another uncoordinated VNE is proposed in [64]. In the virtual node mapping stage, the nodes are initially sorted

in descending order according to their required CPU. They are then mapped starting from the first node in the sorted list. For the link mapping instead, k–shortest paths are selected, starting from the one that would require the least amount of bandwidth.

Intra–domain and Inter–domain VNE. While a number of studies can be found on the intra–domain VNE [66, 67], only a few works are available on the inter–domain VNE [68, 69, 70]. A single–domain robust VNE problem is investigated in [66] for mobile networks. The problem is formulated as an ILP problem whose main objective is to provide a robust optimization framework, considering the shortest–path VNE problem. As opposed to many other studies that consider deterministic traffic demand and omit users' mobility, this study takes into account users' mobility and stochastic traffic demand.

Topological attributes play an important role in VNE. A topology–aware six VNE algorithms are proposed in [67] based on seven topological attributes. The first algorithm considers nodes' degree in the node mapping as the number of direct links to their neighbors. Apart from the degree, the second algorithm ranks nodes taking into account their fairness, which is defined as the sum of their shortest distances to all the other nodes. Considering the degree and the fairness, the third algorithm further extends the node mapping through betweenness centrality, which quantifies the frequency that the node serves as a bridge along the shortest path between any other two nodes. The rest of the algorithms modify the Random Walk algorithm, implementing eigenvector centrality, Katz centrality as well as other topological characteristics.

Many practical virtual networks require an end–to–end connectivity in a wide area, which may increase the geographical footprint of a single InP's network, thus requiring connectivity across multiple networks. It is rarely the case when a single InP controls an entire end–to–end path [71]. Large–scale virtual network deployment requires the need for provisioning and operating virtual networks across multiple InPs' networks so as to offer services to end users in a wider area. In order to exempt service providers from the burden of negotiating with individual InPs for inter–domain VNE, an intermediate broker–like entity in introduced, termed Virtual Network Provider (VNP). VNRs are initially relayed to the VNP whose role is to make an efficient virtual network decomposition (with minimized virtual network provisioning cost) suitable to be embedded on different InPs' network. However, VNPs are expected to have very limited knowledge on InPs' networks since many of them are reluctant to disclose their network topology, connectivity, and resource information.

Online VNE algorithms are proposed in [72] for the inter–domain VNE problem. Different information sharing schemes are defined in order to retrieve some information on InPs' networks without disclosing the network details. The authors of [73] propose an approach for the multi–domain VNE that circumvents the need for using VNP that provides only limited information on InPs' network. Specifically, PolyWiNE is presented, a protocol that allows information between service providers and InPs to be exchanged in the distributed embedding process. Moreover, PolyWiNE enables individual InPs to develop their own embedding algorithms and policies. Unlike the studies that employ VNPs to partition virtual networks, the

approach considered in [68] offers a customer–driven virtual network partitioning strategy, which guarantees a fair competition environment among InPs through auction. Different pricing (e.g., package pricing) and auction (e.g., sealed Vickrey) models are also discussed.

VNE with Static and Dynamic Resource Allocation. Even though NFV provides a number of advantages, enabling multiple virtual networks with their diverse requirements and policies (e.g., bandwidth, delay) to coexist in the same physical node, it poses a critical challenge in terms of efficient usage of substrate resources. With respect to resource allocations, VNE can be static [74, 75, 76] or dynamic [19, 77, 78, 79]. Static resource allocation implies that a fixed amount of resources are allocated to virtual networks without being able to change it over time. For example, if two virtual networks share the same substrate node, in the case of more traffic demand in the first virtual network, it cannot use more resources than it has been allocated, despite the second virtual network may have idle resources. This leads to inefficient usage of precious substrate resources.

Static resource allocation is studied in the VNE problem presented in [76]. Considering the dynamic arrival of VNRs, admission control is applied to VNRs, giving mapping priority to the ones with the highest revenues for the given time window of collecting VNRs. In order to increase the acceptance ratio of VNRs, path splitting and path migration techniques are employed. [75] investigates the VNE problem in SDN/OpenFlow–based network environment. Different from previous works, the VNE algorithm proposed in [80] aims at reducing the computational complexity of the embedding and improving the substrate resource utilization efficiency. The processing time is shortened by introducing the substrate subgraph, containing relevant network resource records created by taking into consideration load balancing.

As opposed to the VNE with static resource allocation, in the VNE with dynamic resource allocation, the requirements of VNRs can be changed over time, and the idle resources of one VNR can be used by another VNR. In [19], a VNE problem is proposed which considers time–varying resource requirements of the VNRs that have exponentially distributed lifetime. In particular, an opportunistic resource–sharing–based framework is designed where the time-varying sub–requirements of VNRs are modeled using a probability–based modeling technique. A dynamically adaptive virtual resource allocation mechanism is designed in [77] for a VNE problem. The proposed algorithm aims to maximize the acceptance rate, and therefore, also InPs' revenue by increasing resource allocation efficiency. The basic idea is to divide the requested virtual network topology into star topologies then formulate a K–supplier problem for each star topology and solve it using an approximation bottleneck algorithm. Having an objective of minimizing the global mapping cost, a backtracking algorithm is then employed in order to reconstruct the initial virtual network topology. Another dynamic VNE algorithm is presented in [78], having the goal to minimize the embedding cost. The authors in [79] present a VNE problem with the goal of optimizing energy consumption while still providing high revenues for InPs.

Table 3.1 summarizes some of the state–of–the–art studies in each group. Specifically, the optimization objective and the optimization algorithms are specified for each referred work.

Table 3.1 – State–of–the–art VNE studies

Ref.	Optimization objective	Optimization algorithm		
		Exact		Heuristic
		Method	Tool	
Online VNE				
[53]	Min. overall flow cost	MILP	–	–
[54]	Maximize acceptance ratio of VNRs and InPs' revenue	–	–	Heavy Edge Matching
[55]	Maximize revenue and minimize embedding cost in the long run	MIP, LP	GLPK	–
Offline VNE				
[56]	Maximize profit	MILP	CPLEX	–
[57]	Maximize profit	MILP	CPLEX	Two–phase MILP
[58]	Minimize the maximum node and link stress	–	–	Greedy
Coordinated VNE				
[60]	Maximize total revenue	ILP	CPLEX	–
[17]	Maximize total number of embedded virtual networks	MIP	GUROBI	–
[61]	Minimize blocking probability maximize time–average revenue.	–	–	Greedy
Uncoordinated VNE				
[62]	Minimize the number of working and intermediate nodes.	ILP	GLPK	Best–fit/Worst–fit
[63]	Maximize InPs' revenue	–	–	Greedy
[64]	Minimize embedding cost	–	–	Greedy
Intra–domain VNE				
[66]	Minimize the total routing cost	LP	MATLAB	–
[67]	Maximize long–term average revenue and acceptance ratio	–	–	Greedy, Random Walk
Inter–domain VNE				
[68]	Min. virtual network setup cost	–	–	Vickrey auction
[69]	Minimize embedding cost	–	–	Particle Swarm Opt.
[70]	Minimize embedding cost	MIP	ILOG	Max–flow/min–cut
VNE with static resource allocation				
[74]	Min. overall flow table occupation	ILP	CPLEX	–
[75]	Min. energy costand max. revenue	ILP	GLPK	Particle Swarm Opt.
[76]	Minimize the embedding cost and maximize the revenue	–	–	Greedy
VNE with dynamic resource allocation				
[79]	Mimimize energy consumption	ILP	GLPK	EAD–VNE
[77]	Maximize acceptance rate	–	–	Approx. bottleneck
[78]	Minimize embedding cost	MLP	–	–
[19]	Max. substrate resource util.	–	–	Bin packing

3.3 CU Placement

In the C–RAN architecture, the terms BBU/CU placement and DU–CU mapping are often used to refer to the same problem, which is mainly manifested in two forms. The main question answered by the first CU placement problem is the following. **How can MNOs build C–RAN in the cheapest possible way, making sure that their traffic demand is satisfied?** In other words, given the mobile data traffic demand, the problem is to find the number of CU pools, their location and assign CUs to the optimal CU pools, optimizing the Total Cost of Ownership (TCO) of the network. The second CU placement problem answers the following question. **Given a C–RAN architecture with the locations of CU pools and DUs, how can MNOs embed VNRs received from MVNOs in such a way as to utilize the network resources in the most efficient manner?** These variants of the CU placement problem can be formulated as a VNE problem. A sizable body of work has been published on both.

An optimization algorithm is presented in [81] for the CU placement problem over Fixed/-Mobile Converged (FMC) optical networks. The authors formulate an ILP problem, which efficiently calculates the minimum number of required CU pools taking into account the maximum allowed distance between DUs and their CUs. The same authors propose an energy–efficient CU placement algorithm for FMC optical networks in [82], aiming at minimizing the Aggregation Infrastructure Power. In both studies, the authors assume that a CU pool can be deployed at any node (e.g., Optical Line Terminator (OLT)s, Optical Network Unit (ONU)s, optical switches) in the network.

An energy–efficient DU–CU mapping algorithm is proposed in [83]. Aiming at minimizing energy consumption at CU pools, computing resource requirement of DUs and inter–DU traffic exchange are considered for assigning DUs to CU pools. The authors in [84] propose an energy–efficient scheme for the optical–transport–enabled C–RANs by introducing the concept of a virtual BS and enabling resource sharing of CUs and line cards of OLTs.

A two–stage design mechanism, namely downlink Orthogonal Frequency–Division Multiple Access (OFDMA) resource allocation and power allocation, and DU–CU assignment for dynamic resource sharing, is proposed in [85]. In the first stage, an MILP problem is formulated for finding the best DU assignments and the best Physical Resource Block (PRB) allocations for the mobile users, considering the Signal to Interference plus Noise Ratio (SINR) threshold for each of them. In the second stage, by employing the results obtained in the first stage, a DU–CU assignment problem is formulated as a Multiple Knapsack Problem (MKP), taking into account the real–time traffic load in DUs. The CU placement problem is also studied in [86]. The problem is formalized and solved using exact and heuristic algorithms having the goal to minimize the total deployment cost of the mobile network while satisfying the traffic demands.

However, unlike our work presented in Chapter 4, none of the mentioned works studies the dynamic CU placement problem over a reconfigurable substrate network. Moreover, none of them considers a reconfigurable mmWave fronthaul, which based on certain criteria (e.g.,

daytime versus nighttime traffic patterns, or when a new request arrives) can dynamically re–embed VNRs with the objective of reducing the network–wide power consumption.

Traffic– and interference–aware dynamic DU–CU mapping algorithm is proposed in [87]. Mutual coupling loss, which characterizes Cross–carrier Co–channel Interference (CCI) between DUs, is taken into account in order to find the most optimal DU–CU mapping, which apart from load balancing CUs and minimizing power consumption, would also minimize the CCI between DUs. Semi–static and adaptive DU–CU switching schemes are proposed in [88]. The schemes, although have the same objective of minimizing the required number of CU pools in order to meet the traffic demand at each DU, differ in terms of the DU–CU switching interval. The same authors [89] elaborate more on the DU–CU switching plan, considering also the signaling load caused by users' handover while making DU–CU switching decisions. In all aforementioned works, however, the authors do not study how to select CU pool locations and how to assign DUs to CU pools in the case of multiple CU pools in order to get the highest resource multiplexing gain, since it is important to consider not only spatially and temporarily fluctuating traffic but also the distance between DUs and CUs.

A CU placement problem is studied in [90], considering different LTE–A configurations and investigating the impact of different CU centralization level on both CAPEX and OPEX in an optical network–supported C–RAN. The authors of [91] propose a dynamic DU–CU mapping scheme employing a borrow–and–lend approach. The key idea is to migrate the DUs assigned to a highly utilized CU to a less utilized CU, having the objective of maximizing the utilization of every single CU inside the CU pool. However, only one CU pool is considered in these studies without tackling the problem of selecting the number and the locations of the CU pools in order to cater the traffic demand of all the DUs in the network. Moreover, the authors consider no CU placement problem simply assuming that all DUs are assigned to a single CU pool via direct links.

The authors of [92] study the problem of minimizing the CAPEX for those MNOs who want to design a mobile network from scratch, adopting the C–RAN architecture. Specifically, the authors study the trade–offs between the resource multiplexing gain, which would increase by assigning more DUs to the same CU pool, and the fronthaul network deployment cost, which would reduce if more CU pools were available for DUs to be associated with. However, the authors make some simple assumptions which would not be reasonable to be applied to existing mobile infrastructures. For example, they assume that the fronthaul links are directly connected to the CU pool. They also categorize BSs into two types, office and residential, and assume that the same number of office cells are allocated to the CU pools. Nevertheless, they do not delve into the granularity of hourly traffic requirement of each cell in order to better understand the CU composition of which DUs would provide the maximum multiplexing gain, since the traffic demand in some office and some residential areas might be such that they would not provide high multiplexing gain being assigned to the same CU pool.

An optimization problem similar to ours is presented in [93]. A CU placement problem is formulated aiming to find the optimal quantity and locations of CU pools with the goal of minimizing TCO. The migration cost is computed from an optical D-RAN and a Microwave D–RAN to C–RAN considering greenfield and brownfield optical networks for macrocell, microcell and, nanocell deployments. The main difference compared to our work, however, is that the authors do not consider the real traffic demand in each cell of the traditional D–RAN, which plays a pivotal role in selecting the number and the locations of the CU pools. Different from the previous study, [94] investigates the CU placement problem for the C–RAN that employs Wavelength Division Multiplexing (WDM) aggregation technology in the fronthaul network. The authors formulate a joint and an independent CU and electronic switch placement problems, considering their placement possibilities in different parts of the Optical Transport Network (OTN) and the Overlay fronthaul transport network.

The research to date has tended to focus on building future mobile networks from scratch based on the traffic demand. A common characteristic of the aforementioned works is that none of them has studied how MNOs, owning legacy LTE networks, can upgrade the network by adopting the C–RAN architecture with the minimal cost by employing the available site locations, transmission links, the knowledge of the CU pool candidate locations, and hourly traffic demand per cell. This is a relevant problem since we believe that a few MNOs would be willing to invest a huge amount of money in building C–RAN from scratch, when they can just reuse the legacy infrastructure and, therefore, significantly curtail the required investments in order to deploy C–RAN. An approach is presented in Chapter 6 to tackle this problem.

3.4 Mobile Network Sharing

Network sharing has been evolving since the arrival of 3G networks, reaching from passive sharing such as sharing of site locations, antenna masts, to active sharing techniques such as Multi–Operator Core Network (MOCN) and GateWay Core Network (GWCN). The active network sharing has paved a way for new business opportunities, enabling InPs to host MVNOs, Over–The–Top (OTT) service providers and vertical market players over their physical network. The authors of [95] introduce an on–demand capacity broker concept, which is able to securely expose selected service features via Application Programming Interface (API), allowing InPs to allocate the required portion of their networks to MVNOs, OTTs, or vertical market players. In [52], the authors present a VNF placement problem in which InP's RAN can be shared among several MVNOs.

Being inspired by the concept of "everything" as a service (XaaS) and having the goal of allowing mobile network operators to offer a customizable end–to–end service to MVNOs, the Network–Slice–as–a–Service (NSaaS) concept is introduced in [96]. More specifically, three service models are proposed: application level, network function level, and infrastructure level. Additionally, several implementation possibilities such as network slicing only in the core network, only in the RAN or in both in the core network and in the RAN, are discussed. The last slicing option (slicing in both core network and RAN), which implies that each slice

of the RAN is connected to a core slice, along with the infrastructure level service model is of interest for our work presented in Chapter 5. In the infrastructure level service model, network resources such as PRBs, antennas, DUs, CPU cores, memory, and storage can be allocated to each of the MVNOs, guaranteeing resource isolation between them.

An extensive survey on network slicing can be found in [97], highlighting the major problems associated with network slicing and isolation between slices. Various approaches to wireless slicing are presented for different technologies such as LTE, WiMAX, and Wi–Fi. Whereas, a detailed study on the impact of network slicing in 5G RANs can be found in [98].

3.5 Functional Splits in C-RAN

The distribution of RAN functionalities between the DUs and the CU pool depicted in Figure 2.1c corresponds to full resource centralization, referred as PHY–RF split. As discussed in Section 2.2, in fully centralized C–RAN, the entire baseband signal processing is taking place at the CU pool, while DUs are equipped with dumb antennas and are in charge of basic RF functionalities (e.g., analog–to–digital and reverse conversion, signal amplification, etc.). The main concern with this split is the cost of the fronthaul network and availability of suitable fronthaul transport to meet its high bandwidth and tight latency requirements. In order to circumvent these limitations yet providing the possibility to obtain resource centralization and virtualization gains, other intermediate functional splits have been defined [99, 42].

As mentioned in Section 2.2, the C–RAN architecture has the potential to bring a number of benefits into the network, starting from reducing the TCO of the network up to enabling the use of advanced multi–cell coordination techniques. Nevertheless, the level of the benefits that can be reaped by employing the C–RAN architecture depends on the actual functional split option between the DU and the CU.

This section investigates the state–of–the–art studies conducted on C–RAN. This encompassed main functional split options and various fronthauling transport mediums/technologies/protocols that are on the focus of active research by many telcos and research institutes. We first examine the most promising functional split options, highlighting their pros and cons while presenting the key results achieved by the recent studies that consider these splits. We then delve deeper into various transport mediums, technologies, and protocols that can be potentially used in the fronthaul network in different functional split scenarios.

3.5.1 Functional Split Options

The functional split problem has attracted a great deal of attention from both the academia and the industry [12, 11, 10]. There are in fact different approaches to small cell virtualization in terms of the point at which the BS's operations are decomposed into physical and virtual. A number of factors (e.g., mobile data traffic demand, interference scenario, and latency constraints) have to be considered in order to select the optimal split point. Figure 3.1 illustrates

Table 3.2 – Fronthaul bandwidth and one–way latency requirements (absolute and relative) for the presented functional splits [8].

Functional splits	DL bandwidth	Latency	Latency class
PHY–RF Split	2.457 Gbps ($x1$)	250 μs ($x1$)	Ideal
PHY I Split	1.075 Gbps ($x2.28$)	250 μs ($x1$) 2 ms[1] ($x8$)	Ideal Near Ideal
PHY II Split	0.932 Gbps ($x2.63$)	250 μs ($x1$) 2 ms[1] ($x8$)	Ideal Near Ideal
PHY III Split	0.173 Gbps ($x14.19$)	250 μs ($x1$) 2 ms[1] ($x8$)	Ideal Near Ideal
MAC–PHY Split	0.152 Gbps ($x16.14$)	250 μs ($x1$) 2 ms[1] ($x8$)	Ideal Near Ideal
MAC Split	0.151 Gbps ($x16.24$)	6 ms ($x24$)	Sub Ideal
PDCP–RLC Split	0.15 Gbps ($x16.37$)	30 ms ($x120$)	Non Ideal

the basic signal processing functionalities in the RAN protocol stack for LTE in both uplink and downlink directions, highlighting some of the promising functional split options. All functionalities in each layer within the RAN protocol stack are divided between the DU and the CU. Specifically, the functionalities below each split line are performed in the DU, which serves as a network attachment point for users since from their perspective it is the closest network entity; whereas, the rest of the functionalities above the split line are performed at the CU. Since the PHY–RF split has been presented in detail in Section 2.2, this section discusses the rest of the functional split options outlined in Figure 3.1. The functional splits are presented with a bottom–up approach in which fewer functions are gradually centralized at the CU pool. Table 3.2 summarizes the downlink fronthaul bandwidth and one–way latency requirements (absolute and relative) derived from [8] for the presented functional splits. These requirements are calculated for a single reference user per Transmission Time Interval (TTI) (1ms in LTE/LTE–A) with 150Mbps downlink traffic, using 2×2 MIMO antenna configuration and 64QAM modulation in an LTE network with 20MHz channel bandwidth.

PHY I Split: In the uplink direction, after receiving signals, the Cyclic Prefix (CP) is removed and the signal is transformed into the frequency domain (FFT). The guard signals, which constitute around 40% in the overall received signal, are thereby removed. In essence, this leads to a downlink bandwidth reduction by a factor of 2.28 compared to the time–domain IQ forwarding in the full centralization case (i.e., PHY–RF split). Moreover, the PHY I split could have 8 times relaxed fronthaul latency requirements in comparison with the baseline PHY–RF split. This split is worth to be considered since FFT/IFFT is rather complex to be virtualized and implemented in a commodity hardware in the CU pool contrary to its specialized, purposed–built hardware in DU. However, like for the PHY–RF split, also for the PHY I split the required fronthaul bandwidth is constant, since the PRB demapping is still executed in the CU pool, and depends on the system bandwidth and other characteristics of the given BS.

[1] 2 ms latency relaxation is obtained by halving a single user achievable throughput.

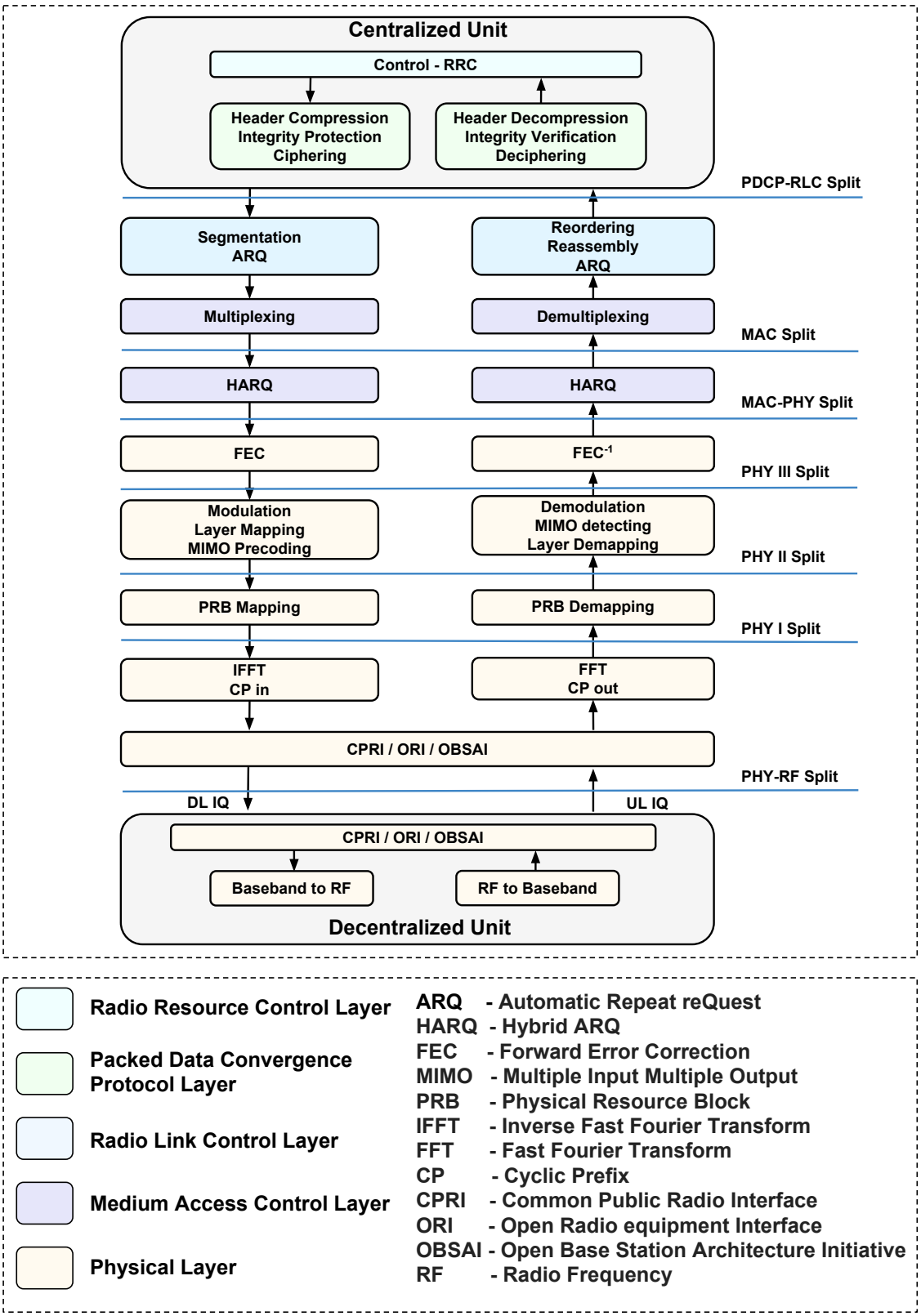


Figure 3.1 – Main functional split options at C–RAN.

PHY II Split: In the PHY I split, the signal transmitted on the fronthaul from the DU to the CU still carries information that is of no interest to the user. In the PHY II split, by further moving the PRB mapping/demapping to the DU side, the required fronthaul bandwidth can be reduced since it will scale with the actually occupied PRBs. For instance, if the user is allocated 50 PRBs out of 100 PRBs, which translates to 600 subcarriers out of 1200, then only those 600 subcarriers are transmitted from the DU to the CU that are actually used by the user. Similar to PHY–RF split and PHY I split, also this functional split implies almost no restrictions on realizing centralization gains; that is, MNOs can still take advantage of the users' centralized processing that can be jointly performed at the CU pool. Moreover, PHY II split shares the same fronthaul latency like the PHY I split. In contrast to the PHY I split, however, the required fronthaul data rate of this split and the ones above scale with users' actual data traffic, which increases with their radio access channel quality. This allows for exploiting the statistical multiplexing gain on the basis of the occupied PRBs, which, in turn, enables the subsequent splits to profit from load balancing gains. Moreover, this enables the exploitation of packet–based network to transport the fronthaul data.

PHY III Split: By further placing layer mapping/demapping, MIMO processing, and modulation/demodulation at DUs, it is possible to significantly relax the fronthaul bandwidth requirement. As it can be seen in Table 3.2, while the fronthaul latency requirement is the same like for the PHY I and PHY II splits, compared to the baseline PHY–RF split, the fronthaul bandwidth requirements can be relaxed by a factor of 14.19. Since the synchronization is performed at DUs, the fronthaul rate depends only on the modulation, and therefore, on the Signal-to-Interference-plus-Noise Ratio (SINR) experienced by users. Moreover, since in this split also the MIMO processing is taking place locally at the DUs, the fronthaul bandwidth scales with the number of spatial layers instead of the number of antennas. Notice however how these requirements are relieved at the expense of reduced resource centralization gain. For example, as opposed to the previous functional splits, PHY III split can employ Coordinated Multipoint (CoMP) features such as JT/JR with significant performance degradation [11], while it can still take advantage of joint decoding. This may deteriorate performance especially for cell–edge users, which are the ones that benefit the most from the interference reduction/cancellation features enabled by CoMP.

MAC–PHY Split: The MAC–PHY split suggests clear separation between the Medium Access Control (MAC) layer and the PHY layer. Specifically, in this case, the entire PHY–layer signal processing is handled by the DU, while the MAC and the layers above are virtualized at the CU. After Forward Error Correction (FEC) decoding, the decoder extracts the data bits from the received symbols, outputting the pure MAC payload. Hence, the fronthaul bandwidth is tightly coupled with the actual user throughput and, therefore, almost no bandwidth resource is wasted. Although joint PHY–layer processing is no longer possible for this split, techniques such as joint scheduling can still be employed. Like for the splits PHY–RF, PHY I, PHY II and PHY III, the main challenge for this split is the tight latency requirement defined by HARQ processing, which requires a NACK/ACK to be received within 4ms after a packet has been transmitted. Subtracting physical layer processing delay, the maximum fronthaul

delay budget becomes 250 μs [8]. Although HARQ interleaving, which defers the HARQ transmission/retransmission to a subsequent HARQ period by remotely sending an ACK to the user without granting new data indication (the detail of this technique can be found in TS36.321 [43]), can be applied in order to stretch this stringent fronthaul delay requirement to 2 ms, it will halve the peak data rate for users. Nonetheless, this split has attracted a great deal of attention by the industry leading to the release of nFAPI, a standardized fronthaul interface between the DU and the CU [100].

MAC Split: In this split, also the HARQ procedure is performed at the DUs while the rest of the MAC functions along with the upper layers are virtualized and consolidated at the CU. Compared to the PHY–RF split, the MAC split allows relaxation of the latency requirements by a factor of 24 and the bandwidth requirements by a factor of 16.24. In MAC–PHY split, functions such as joint decoding can no longer be exploited, however, joint scheduling and joint path selection are still possible. This split option is more preferable compared to the subsequent Radio Link Control (RLC)–MAC split (not illustrated in Figure 3.1) since it provides more benefits while having the same fronthaul bandwidth and latency requirement.

PDCP–RLC Split: Finally, in the PDCP–RLC split, all functionalities are taking place at the DUs with the exception of the PDCP layer and the Radio Resource Control (RRC) layer processing, which are virtualized at the CU. The PDCP–RLC split roughly has the same fronthaul bandwidth requirement like the MAC split, since only the MAC and RLC header information bits, which are negligible, are removed from the signal before transmitting it to the PDCP layer. Nonetheless, this split differs significantly from the other splits in terms of fronthaul latency requirement (non–ideal latency, see Table 3.2). The main advantages of PDCP–RLC split include the possibility of WiFi offloading, load balancing and the ease of multi–RAT operation through a common PDCP layer. Furthermore, the PDCP–RLC split expedites mobility management functionality especially for those UEs whose DUs are linked to the same CU pool, which no longer needs to interact with the core network in order to perform the UEs' handover [101]. The challenge, however, is that the PDCP–RLC split would call for re–sequencing at both DU and CU pool side of this split since the PDCP–RLC split requires in–order packet delivery. The re–ordering buffer would impose additional processing burden on the two endpoints, adding latency to the path. 3GPP [101] has recently standardized PDCP–RLC split. This split was considered the most straightforward option to be standardized since its fundamentals had already been standardized for LTE Dual Connectivity.

Several organizations and standardization bodies such as 3GPP [10], xRAN [102], NGMN [11] and SCF [8] are working on functional splits. RAN WG3 of 3GPP concentrates their work on the PDCP–RLC split (option 2) and the high RLC–low RLC (option 3.1) split from the higher layer functional splits, and MAC–PHY split (option 6) and the intra–PHY (option 7) split from the lower–layer functional splits. xRAN investigates two variants of a single functional split. In the first variant, called Category A, the precoding is not performed at the DU, while in the second variant, called Category B, the precoding is performed at the DU. The selection of this particular split is underpinned by its interface simplicity (user data is transmitted based on

the actually used PRBs) and the transport bandwidth scalability (as opposed to the PHY–RF split and the PHY I split in which the interface scales based on the number of antennas, in this functional split, which is not illustrated in Figure 3.1, the interface scales based on streams). While NGMN considers only the splits within and up to the PHY layer, SCF discusses a number of functional splits (some of which are not illustrated in Figure 3.1) and details their associated fronthaul bandwidth and latency requirements along with their advantages and challenges.

Functional split has also attracted significant attention by academia. A comprehensive survey on the aforementioned functional splits is presented in [42], highlighting the main findings of the recent studies conducted on both theoretical and practical level. A detailed discussion on various functional splits can also be found in [103, 104, 105, 99, 106]. Being inspired by "everything" as a service (XaaS) concept, a novel RAN–as–a–Service (RANaaS) concept, which resembles Infrastructure–as–a–Service (IaaS) model, is introduced in [103]. In RANaaS, centralization of management and processing is flexible and can be adapted to the actual service demands. It employs cloud computing technologies in order to centralize RAN functionalities with the goal of increasing the system throughput and the system utilization efficiency. Moreover, relying on joint design and optimization of access and backhaul networks whose challenges are outlined in [107], it seeks to reduce energy–per–bit and cost–per–bit compared to LTE–A technology. The authors of [104] further detail the RANaaS architecture, discussing the challenges and potential technologies to implement functional splits in the RANaaS platform.

The authors of [99] present quantitative analysis of PHY I, PHY II, MAC–PHY, MAC, and PDCP–RLC functional splits with a particular emphasis on the approaches aiming at increasing the processing time budget imposed by the HARQ procedure for the first three split cases. Moreover, the economic aspect in terms of TCO for the discussed functional splits is investigated. The study presented in [106] conducts detailed analysis on PHY II, PHY III, and MAC–PHY splits. A detailed investigation on intra–PHY later functional splits have also been conducted in [105], providing numerical results on the required fronthaul data rates for each envisioned split. As opposed to [99], however, more focus is placed on the flexible functional split selection opportunities. Additionally, the authors discuss the advantages and challenges of signal processing functionalities, such as FEC decoding, in the cloud platform.

The authors of [108] propose a graph–based framework to split and place baseband functions, studying the trade–offs between the baseband function computation cost and fronthauling cost. Specifically, the physical layer signal processing chain is represented as a directed graph, and the placement problem is formulated as a graph–clustering problem, which is then solved using a customized genetic algorithm. In [109], a theoretical study is presented that jointly considers functional split selection and scheduling policy. Optimization problems are formulated aiming at minimizing the total latency, which is computed as the sum of the scheduling latency, the processing latency at the DUs and the CU pool, and the fronthaul transmission latency over all DUs. Different from the mentioned works, a control plane / user plane split is discussed in [110], highlighting its pros and cons, while based on burstiness

of the traffic and the fact that the mobile data traffic varies depending upon the area (e.g., residential, office) and the time of a day, mathematical and simulation methods are proposed in [111] for quantifying the multiplexing gain of the PHY–RF and the PDCP–RLC functional splits. WizHaul is presented in [112], a planning tool that is able to jointly select the optimal fronthaul route and the optimal functional split in packet-based transport networks with high path diversity. An optimization problem is formulated with the goal of maximizing the centralization degree (i.e, selecting low-layer functional splits for base stations) subject to the delay requirement of the selected functional split option and the capacity of the fronthaul links. The same authors elaborate more on this problem in [113], apart from the joint fronthaul route and functional split selection, considering also the placement of Multi–access Edge Computing (MEC) functions. The objective of this problem is to minimize the network operator's cost, which encompasses the routing and computing costs, and the delay the MEC services experience, therefore, maximizing the MEC users performance.

However, to the best of our knowledge, our work presented in Chapter 5 is the first that considers the possibility of dynamically adapting a small cell functional split based on the evaluated ICI level at each small cell.

3.5.2 Fronthaul Technologies and Protocols

This section details some of most common wired and wireless technologies and discusses their applicability in the fronthaul network of the C–RAN architecture. Moreover, it presents some fronthaul protocols supporting various functional split options.

3.5.2.1 Wired Fronthaul

Wired technologies that can be used in the fronthaul network can be classified into two groups: copper–based and fiber–based technologies. They both are time–consuming to be installed, if feasible, with the former requiring huge investments. In this section, we present T1/E1, VDSL2 copper–based, and xGPON, Ethernet fiber–based technologies.

Cooper–based: Among the copper–based solutions, leased T1/E1 copper lines have been extensively used in 2G/2G+ mobile networks since they were capable of supporting voice traffic with deterministic QoS, low latency and jitter. With the arrival of 3G networks, however, they have not been considered as a candidate for the backhaul network due to their limited scalability in terms of bandwidth. This is a bottleneck also for any Digital Subscriber Line (xDSL) technology, whose recent standard, Very–high–bit–rate Digital Subscriber Line 2 (VDSL2), is able to support up to 200 Mbps in both downstream and upstream. Moreover, it is limited in the distance; for example, VDSL2 technology can efficiently reach up to 1.6 km with good performance. Thus, the VDSL2 technology is suitable to be used only in the last mile of the fronthaul network for only the functional splits that are above the PHY layer.

Fiber–based: Optical–fiber–based technologies undoubtedly provide higher data rates in comparison with wireless technologies. One of such technologies is Gigabit Passive Optical Network (GPON), whose further advancement has been undertaken by Full Service Access Network (FSAN) and International Telecommunication Union (ITU). Specifically, GPON evolution has been divided into Next Generation GPON1 (NG–PON1) and NG-PON2. The former is a mid–term upgrade of GPON mostly driven by economic reasons and technology availability, which ensures backward compatibility with the GPON technology. NG–PON1 preserves the point–to–multipoint architecture of GPON and achieves downstream data rate of 10 Gbps and upstream data rate of 2.5 Gbps within the maximum physical transmission distance of 20 km, which are, respectively, four times the downstream data rate and two time the upstream data rate achievable by the GPON technology. The latter (NG–PON2), which was standardized in 2015, further enhances the physical layer specifications reaching 40 Gbps and 10 Gbps up to distance of 40 km, respectively, for the downstream and upstream transmission, employing technologies such as WDM Passive Optical Network (WDM–PON) with possible multiplexing schemes like Course Wavelength Division Multiplexing (CWDM) or Dense Wavelength Division Multiplexing (DWDM). Finally, recent advancements in Ethernet networking technology have enhanced both the scalability and capability of Ethernet as a carrier–grade transport technology for mobile fronthauling/backhauling. IEEE 802.3ba Ethernet standard, which has been approved in June 2010, promises rates as high as 40 Gbps and 100 Gbps. Specifically, single–mode fiber can support 40GBase–LR4 (40 Gbps) and 100GBase–LR4 (100 Gbps) for the distance of 10 km, while 100GBase–ER4 of the same single–mode fiber can cover up to 40 km of distance.

Optical fiber is usually deployed in densely populated areas that have demand for high capacity. Although optical fiber is perceived to be the most promising fronthauling medium in the long run that is capable of supporting the ever–increasing fronthaul/backhaul traffic requirements even for the most ambitious functional split, it incurs relatively high CAPEX. This is the reason why fiber–based links cannot simply be provided to thousands of small cells that are expected to be deployed aiming at meeting exponentially increasing traffic demand.

3.5.2.2 Wireless Fronthaul

Wireless fronthaul/backhaul technologies can be classified according to the frequency band that they operate at. The common advantage of these technologies, compared to their wired counterparts, is their significantly faster installation. In this section, we present Free Space Optics (FSO), microwave and millimeter wave technologies, discussing their pros and cons.

Free Space Optics (FSO): FSO is a Line–of–Sight (LoS) wireless communication technology that harvests the benefits of both wireless links and wired optical links. Specifically, it uses invisible light spectrum, (e.g., 850 nm or 1550 nm wavelength) typically used in optical communication, to deliver wireless bandwidth as high as 1.25 Gbps as reported by CableFree [114]. Its main benefits over an optical fiber are easy and fast installation, thanks to its wireless nature, and lower latency, due to the fact that light transmission over the air is around 40%

faster than through the optical fiber. Better yet, as opposed to other wireless technologies, it has no need for purchasing a license in order to exploit the spectrum, thereby lowering its cost and accelerating the time to market. Conversely, like mmWave wireless technology, it uses a rather narrow light beam, which makes it immune to interference. The main challenge with the FSO technology, however, is that it suffers heavily during snowfall and fog, thus limiting its length and deployment scenarios. Nonetheless, its benefits make FSO a viable technology to be used in the fronthaul network for the functional splits higher from the PHY layer.

Microwave Fronthaul: Microwave transmission by far is the most prevalent backhaul technology widely used in mobile communication especially in the areas where installation of wired transmission links is problematic. Currently, state–of–the–art microwave technologies, operating at frequencies between 6 GHz and 42 GHz, deliver from a few hundred Mbps up to 1 Gbps data rate, which is expected to increase according to the roadmaps of this technology vendors. Therefore, depending on the functional split option, it could be a viable and cost–efficient alternative to the optical fiber to be used in the fronthaul network. For instance, the present–day microwave technology, due to its insufficient link capacity, cannot satisfy the fronthual bandwidth requirement of the classical PHY–RF split; whereas, it could be a suitable fronthualing option for higher–layer splits such as MAC–PHY or PDCP–RLC split, satisfying the fronthual bandwidth requirement of the split and enabling MNOs to garner its benefits.

Millimetre Wave (mmWave) Fronthaul: Any communication within 30–300 GHz frequency band is referred as a mmWave communication. Recently, mmWave technology started gaining momentum as an efficient technology capable of delivering multi–Gbps capacity on the fronthual and backhaul network due to wider bandwidth compared to microwave technologies [115]. In general, mmWave spectrum is divided into four bands: V–band, E–band, W–band and D–band with total of, respectively, 7–9 GHz (40–75 GHz frequency range), 10 GHz (60–90 GHz frequency range), 17.85 GHz (92–114.5 GHz frequency range) and 31.8 GHz (130–174.8 GHz frequency range) bandwidth. A discussion on mmWave communication on all four bands for both fronthaul and backhaul network is presented in [116]. Among these bands, E–band communication, thanks to its peculiar signal propagation characteristics, is considered to be a viable technology to be used in the fronthaul network, supporting its high data rate demands [117]. Indeed, E–band communication can provide data rate as high as 10 Gbps with high security and excellent immunity to interference thanks to its narrow focused beams. Compared to fiber–based technologies, mmWave technology is cheaper, quicker and simpler to be installed due to small form factor. Its main technical challenges, however, lie in the clear LoS requirement, severe propagation loss and high transceiver design complexity for massive MIMO systems. Nowadays, there are many commercial mmWave CPRI–based products operating in the E–band. For instance, EB–link FrontLink™58 supports 7.5 Gbps of data rate in the distance of up to 4 Km, while CableFree mmWave radio supports 2.5 Gbps CPRI data rate up to the distance of 20 Km. Thus, depending on the functional split option, mmWave technology can be a cost–efficient alternative to optical fiber to be used especially for the last mile fronthaul communication. The higher–layer is the functional split option, the more justifiable is the use of mmWave technology from the fronthaul bandwidth perspective.

There is a huge body of work investigating different technologies to be used in the fronthaul network for various functional split cases [118, 116, 119, 120, 121]. A comprehensive study on various fronthauling options is presented in [118]. The authors of [116] study the applicability of mmWave technology in the fronthaul network as a complement to optical fiber for intra–PHY and PDCP–RLC functional splits. An analytical model is derived in [119] for finding the optimal ratio between optical fibers and microwave links, which would reduce the CAPEX required to build the fronthaul network and, at the same time, meet the traffic requirement at each cell site. Several wired/wireless transport fronthauling technologies, as well as the associated bandwidth and latency requirements for different functional splits, are explored in [120]. The possibility of employing xDSL, mmWave and fiber technologies on the fronthaul network in different functional split cases are investigated in [106]. A case–study analysis is presented in [121] for several PHY–layer functional splits, considering a DSL, microwave and optical fiber transport as fronthaul technologies. The authors conclude that among the different functional split, the PHY–RF split with optical fiber fronthaul option is the most profitable one, though it incurs the highest deployment costs.

3.5.2.3 Fronthaul Protocols

CPRI is the legacy constant–bitrate protocol employed in the fronthaul network [37]. It is used to transport users' I/Q data, control, management as well as synchronization information between DUs and CU pools. Specifically, Time Division Multiplexing (TDM) is used to package and transmit I/Q data of different RF front–ends over the transmission line. CPRI traditionally supports only the PHY–RF functional split option whose stringent fronthaul bandwidth and latency requirements have so far kept away the use of packet–switched technologies in the fronthaul network. Although, several CPRI compression techniques are developed with compression rates as high as 30% by elimination redundant spectrum bandwidth [122], the economics of scale and the exploration of other functional splits within the RAN protocol stack between DUs and CU pools with their relaxed requirements drive the need for using packet–based fronthaul technologies such as Ethernet. Significant efforts have, therefore, been made towards the packet–based fronthaul networks [12, 123, 124, 125]. NGFI, for example, strongly advocates the need for using Ethernet as the fronthaul data transmission solution and, apart from the PHY–RF split, examines all intermediate functional splits between the DUs and the CU pools (see Figure 3.1) [12]. Not only does Ethernet accelerate the system rollout by exploiting existing Ethernet networks, but it also offers the benefits of packet–based transmission such as statistical multiplexing, flexible routing, and better scalability. Nonetheless, it also poses challenges in terms of delay, jitter and synchronization since packet–based networks are non–deterministic and lack precise time synchronization, while wireless payloads are rather sensitive to these parameters. Specifically, in comparison with traditional CPRI, packet–based NGFI interface will increase the fronthaul delay due to required packet switching. With respect to jitter, it is required to implement data caching on both sides in order to compensate the jitter. The greater the jitter, the larger buffering space would be required. Whereas in terms of achieving synchronization between different stations, one of

the candidates is Precision Time Protocol (PTP)/ IEEE 1588v2 [126] that is able to provide frequency, phase and time accuracy with nanoseconds precision. Besides, IEEE 802.1CM project [127] of Time–Sensitive Network Task Group is currently working on several standards with the goal of allowing time–synchronized, low–latency and high–bandwidth services over Ethernet networks.

One of the main protocols considered for the packet–based fronthaul network is eCPRI [124]. It sits on top of the transport network layer, and therefore, enables efficient data transmission over any packet–based fronthaul network such as IP or Ethernet. As opposed to CPRI specification that only supports the PHY–RF functional split, eCPRI protocol can be implemented for the PHY II and PHY III splits. IEEE 1914.3 working group defines a standard, called Radio over Ethernet (RoE), for RoE encapsulation and mapping [123]. RoE can be manifested in three forms: I/Q data transmission over Ethernet (Native RoE packet mapper), structure–aware as well as structure–agnostic mappers, for example, for CPRI/OBSAI. Currently, IEEE 1914.3 standard supports the PHY–RF split and the PHY I split. xRAN [125] presents the technical specification of the control plane, user plane and synchronization plane protocols for the fronthaul network with an intra–PHY layer functional split [102]. The protocols can employ either eCPRI or IEEE 1914.3 encapsulation mechanisms, and like the former supports both Ethernet and Internet Protocol (IP)/User Datagram Protocol (UDP) encapsulation types. SCF has standardized nFAPI as a fronthaul interface for the MAC–PHY split between the DU and the CU [100].

Different PHY–layer functional splits are studied in [128] for the packet–based fronthaul networks. In particular, symbol–based, slot–based and subframe–based packetization methods are proposed, and their impact on the fronthaul delay and the fronthaul bandwidth is studied. By performing system–level simulations, the maximum number of supported DUs in the fronthaul network is estimated for different PHY–layer functional splits, considering standard and jumbo frame formats. The same authors describe FlexCRAN [129], a unified DU/CU framework that supports the PHY–RF split and an intra–PHY layer split over Ethernet-based fronthaul network. Three groups of KPIs are analyzed, namely: Fronthaul–related KPIs such as Round Trip Time (RTT) and throughput at the fronthaul links, endpoint–related KPIs such as CPU utilization and memory usage at the DU and CU pool, and data–plane KPIs such as the QoS of the data plane and its delay. By considering the mentioned splits, it is concluded that they can achieve results comparable with the traditional D–RAN deployments in terms of users experience. PDCP–RLC and MAC–PHY functional splits in an experimental LTE network with an Ethernet–based fronthaul are studied in [130]. The splits are implemented and evaluated using experimental and simulation methods. Specifically, different transport protocols are evaluated, and it is demonstrated that the UDP protocol exhibits the best performance.

4 Millimeter Wave Fronthaul in C–RAN

This chapter investigates the use of mmWave technology as an alternative to the optical fiber in the fronthaul network in different functional split scenarios, having an ultimate goal of reducing the fronthaul deployment cost while optimizing the utilization of network resources. Specifically, employing ILP techniques, we first formulate and solve a dynamic CU placement problem for mobile networks with the goal of minimizing the fronthaul bandwidth consumption, and therefore, also minimizing the network–wide power consumption. In the considered network, CU pools are placed at the edges of the network, and a reconfigurable mmWave wireless fronthaul links are used in order to provide DUs with connectivity. We then study the impact of different functional splits on the placement cost and the acceptance ratio using different substrate networks. Lastly, we propose and evaluate a CU placement heuristic using a numerical simulator.

4.1 Overview

Mobile network operators are currently facing a tremendous increase in the level of data traffic. Although cell size reduction is one of the most common ways used to accommodate such traffic demand, densely deployed small cells also dramatically increase the level of ICI. By centralizing baseband signal processing at powerful computing infrastructures (CU pools), C–RAN enables advanced coordination algorithms to be employed in dense small–cell networks, aiming to reduce/cancel the ICI or use it in a constructive manner. In C–RAN, due to stringent bandwidth and latency requirements at the fronthaul links, the optical fiber, thanks to its bandwidth and latency characteristics, continues to be the most viable fronthaul medium option. Its high deployment cost, however, hurdles its widespread adoption. MNOs are therefore seeking for other cost–efficient fronthauling solutions and, apart from classical PHY–RF split, are investigating other functional splits that would allow them to harvest some of the benefits of resource centralization and virtualization without sacrificing all of them.

In this chapter, we employ ILP techniques to formulate and solve a CU placement problem, which is modelled as a VNE problem, having the objective of minimizing the fronthaul bandwidth utilization with an ultimate goal of reducing the network–wide power consumption. We also develop a CU placement heuristic algorithm in order to tackle the scalability issue of the ILP formulation, addressing the larger instances of the problem. Different functional splits are considered in the InP's C–RAN network, referred as a substrate network. Virtual networks, composed of virtual CUs and DUs, are requested by MVNOs and embedded by the InP in their substrate network, where CU pools are placed at the edges of the network, possibly co–located with macro cells and/or distributed clouds, while a reconfigurable mmWave fronthaul is used in order to provide DUs with fronthaul connectivity. The mmWave fronthaul network leverages on steerable directional antennas in order to adapt its topology to different usage scenarios. The reconfigurability of the substrate network allows VNRs to be re–embedded in case of changing traffic patterns (e.g., daytime versus nighttime traffic, when new requests arrive). This allows the traffic of low–utilized mmWave fronthaul links to be aggregated, shutting down the unnecessary mmWave interfaces, and therefore, reducing the power consumption at C–RAN.

4.2 Network Model

The reference network architecture is depicted in Figure 4.1. The lower part of the figure shows a traditional C–RAN deployment with the PHY–RF split where the CUs are centralized in powerful computing facilities, and CPRI optical fronthaul links are used to interconnect DUs with CU pools. The upper part of the figure instead shows the architecture envisioned in this work, where the CU pools and macro cells are co–located and connected to DUs through reconfigurable mmWave fronthaul links. The reconfigurability of the mmWave transmission links allows the interconnections between DUs to be reconfigured, aiming to minimize the number of active mmWave connections, which, in turn, translates to minimization of power consumption. For instance, when the traffic on a mmWave link is low then, while embedding a new VNR, the algorithm may try to re–use the same mmWave link, therefore, avoiding to power up a new mmWave connection or, the algorithm may decide to shut down a low–utilized mmWave link and aggregate its traffic in another more–utilized mmWave link. A traditional S1 link is used for connecting CU pools with the core network. Compared to the traditional (expensive) CPRI optical links, the reconfigurable mmWave fronthaul, although provides less fronthaul capacity, reduces the deployment and the power consumption costs of the fronthaul, while still allowing for improved control and coordination across the small cells.

In this section, we will first detail the notations used for the substrate and virtual network models. We will then introduce the optimal ILP formulation and a scalable heuristic for the dynamic CU placement problem. It is worthwhile to note that in this dynamic CU placement problem only mmWave wireless fronthaul network is considered. Thus, the CU placement problem over the converged wireless/optical network is left as future work.

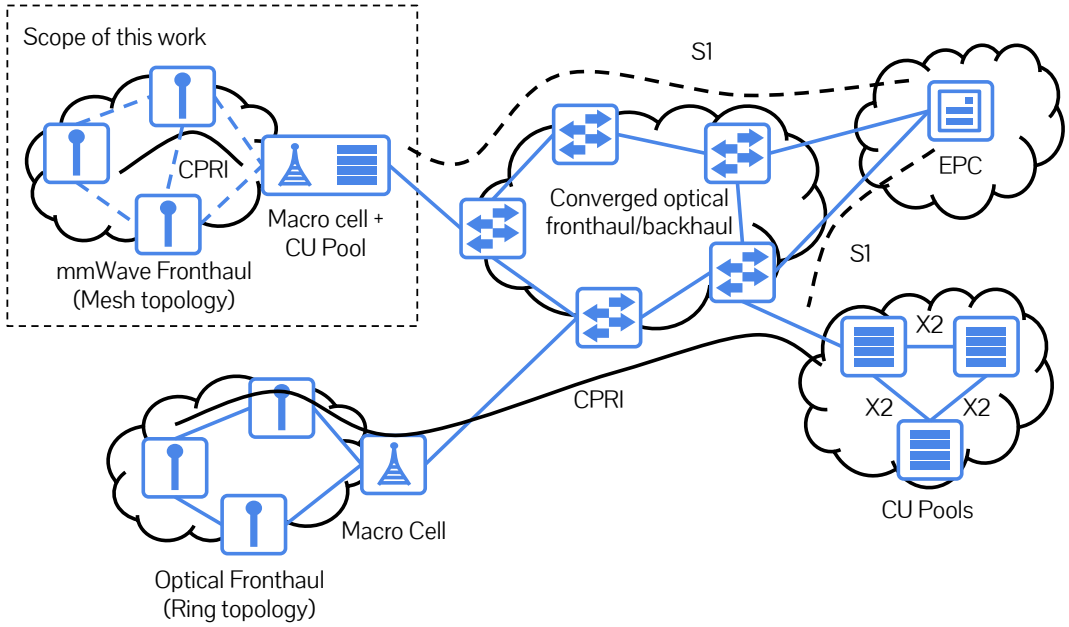


Figure 4.1 – The network architecture used for this CU placement problem. The black solid lines represent the CPRI links while the black dashed lines represent the S1 links. CU pools and the macro cells are co–located, therefore, improving computational locality and shortening the fiber–based CPRI links.

4.2.1 Substrate Network Model

The considered substrate network is composed of computational as well as networking resources. The former consists of micro datacenters, possibly co–located with macro cells. The latter consists of a reconfigurable wireless fronthaul and multiple DUs. The fronthaul network is composed of mmWave routers equipped with a variable number of mmWave interfaces and highly directional steerable antennas. Finally, each DU co–located with mmWave routers is equipped with a variable number of RF front–ends and with processing capabilities.

Let $G_s = (N_s, E_s)$ be an *undirected* graph modelling the substrate network, where $N_s = N_s^{du} \cup N_s^{cu}$ is the set of DUs sites/mmWave Relays and CU pools, and $E_s \subseteq N_s^{du} \times N_s^{cu}$ is the set of fronthaul links. Notice how N_s^{du} nodes[1] in the substrate network can act as both DU and mmWave Relays (i.e., they can both serve end–user terminals over, for example, an LTE air interface and act as relays in mmWave fronthaul), while N_s^{cu} nodes (i.e., the ones co–located with macro cells) can act only as CU pools. A wireless edge $e^{nm} \in E_s$ if and only if a line–of–sight connection exists between the nodes $n, m \in N_s$.

Three weights, $\omega_s^{ant}(n)$, $\omega_s^{int}(n)$ and $\omega_s^{prc}(n)$, are assigned to each node $n \in N_s : \omega_s^{ant,int,prc}(n) \in \mathbb{N}^+$ representing, respectively, the number of RF front–ends, the set of mmWave interfaces and the processing capacity available at the node n. Notice that, since different functional splits are considered for different embedding scenarios, the processing capacity of each substrate

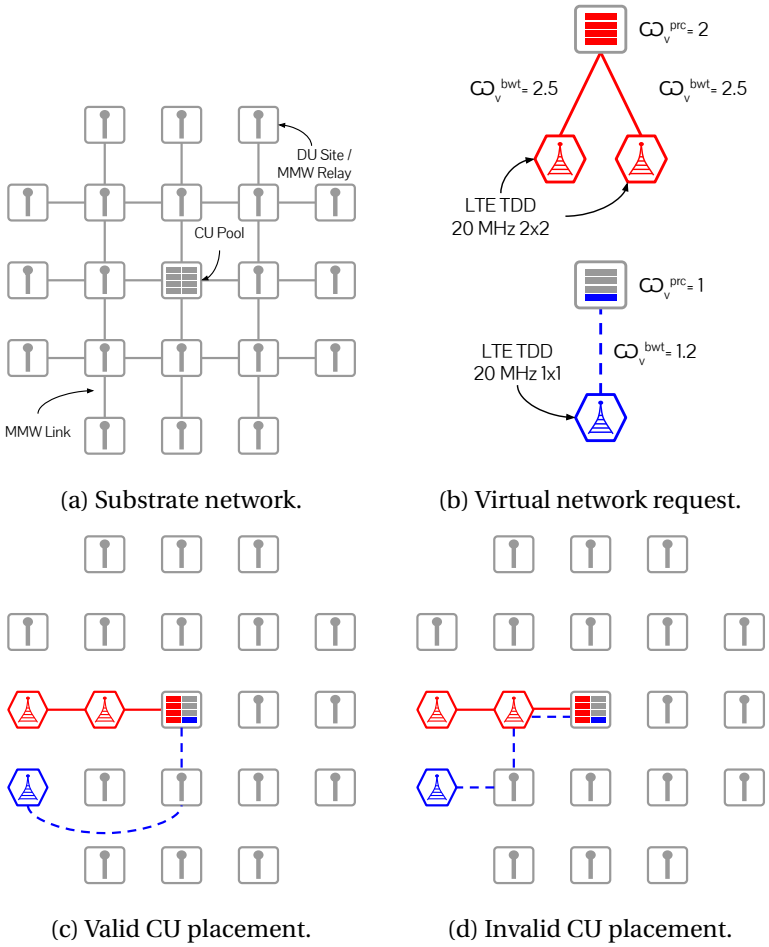
[1] Node is a general term used to refer both CU pools and DUs.

Table 4.1 – Substrate network parameters

Variable	Description
G_s	Substrate network graph.
N_s	Set of substrate nodes in G_s.
N_s^{du}	Set of substrate DUs in G_s.
N_s^{cu}	Set of substrate CU pools in G_s.
E_s	Set of substrate links in G_s.
$\omega_s^{ant}(n)$	Number of RF front–ends available at the node $n \in N_s$.
$\omega_s^{int}(n)$	Set of mmWave interfaces available at the node $n \in N_s$.
$\omega_s^{prc}(n)$	The processing capacity of the node $n \in N_s$.
$\omega_s^{bwt}(e^{nm})$	Capacity of the mmWave link $e^{nm} \in E_s$ (in Gbps).
$\delta(n)$	Coverage radius of the DU $n \in N_s^{du}$ (in meters).
$loc(n)$	Geographical location of the node $n \in N_s$ (x, y).
Λ_{int}	Cost for using the mmWave interface $i \in \omega_s^{int}(n)$ of the node $n \in N_s$.
Λ_{bwt}	Cost for each Gbps of the bandwidth the substrate link.

node (e.g., a DU, a CU pool) depends on the functional split option. We remind the reader that in the C–RAN architecture, a BS is decomposed into two parts, a DU and a CU, and the latter is centralized in a CU pool. The processing capacity at DUs and CU pools for the considered functional splits is reported in Table 4.5. Notice that the reported values are relative to the overall processing capacity of the substrate small cell n. For example, if the PHY–RF split is considered in the substrate network then the DUs do not possess processing capacity; whereas, if the rest of the splits are considered, the DUs do possess processing capacity, which increases at the DUs and decreases at the CU pools when fewer upper–layer functionalities are centralized at the CU pools. Nevertheless, the overall processing capacity (the sum of the processing capacities at the CU pool and at the DUs) of the substrate network is equal to the number of substrate small cells and remains the same regardless of the considered functional split option (see Table 4.5). Each substrate node $n \in N_s$ is associated with a geographic location $loc(n)$, as x, y coordinates, while each DU $n \in N_s^{du}$ is also associated with a coverage radius $\delta(n)$, in meters, indicating the coverage area of the small cell centered on DU n. Another weight $\omega_s^{bwt}(e^{nm})$ is assigned to each link $e^{nm} \in E_s : \omega_s^{bwt}(e^{nm}) \in \mathbb{N}^+$ representing the capacity (in Gbps) of the link connecting the two nodes $n, m \in N_s$. Finally, let P_s be the set of all loop–free substrate paths and $P_s(s,t)$ represent the shortest path between $s, t \in N_s$. Table 4.1 summarizes the substrate network parameters.

A sample substrate network is depicted in Fig. 4.2a. This network composed of 21 nodes is representative of a scenario where DUs are deployed at road intersections in a dense Manhattan–like urban area. The considered topology is solely an example. The problem formulation is generic and can be used for other types of network topologies.



(a) Substrate network.

(b) Virtual network request.

(c) Valid CU placement.

(d) Invalid CU placement.

Figure 4.2 – Sample substrate network, virtual network request and CU placement. Notice how the CU placement in Fig. 4.2c is valid since the interface constraint is satisfied on all relay DUs. Conversely, the CU placement in Fig. 4.2d is invalid since it would require four interfaces in relaying DUs while only two are actually available.

4.2.2 Virtual Network Model

Virtual network requests (VNRs) are modeled as *undirected* graphs $G_v = (N_v, E_v)$, where $N_v = N_v^{du} \cup N_v^{cu}$ is the set of virtual DUs and virtual CU pools, and $E_v \subseteq N_v^{du} \times N_v^{cu}$ is the set of virtual fronthaul links. Note that VNRs do not contain mmWave Relays. An edge $e^{nm} \in E_v$ if and only if the virtual DU n is connected to the virtual CU pool m. Thus, contrary to the substrate network model, edges in the VNRs represent the logical mapping between virtual DUs and their corresponding virtual CU pools. Additionally, we require that each virtual CU pool be mapped to one substrate CU pool. Conversely, different virtual CU pools can be mapped to the same substrate CU pool. This formulation allows MVNOs to specify requests in which a group of virtual CU pools is mapped to the same substrate CU pool, enabling advanced interference control algorithms such as JT/JR.

Table 4.2 – Virtual network request parameters

Variable	Description
G_v	Virtual network graph.
N_v	Set of virtual nodes in G_v.
N_v^{du}	Set of virtual DUs in G_v.
N_v^{cu}	Set of virtual CU pools in G_v.
E_v	Set of virtual links in G_v.
$\omega_v^{ant}(n)$	Number of RF front–ends required by the node $n \in N_v$.
$\omega_v^{prc}(n)$	The processing capacity required by the node $n \in N_v$.
$\omega_v^{bwt}(e^{nm})$	Requested bandwidth for the link $e^{nm} \in E_v$ (in Gbps).
$loc(n)$	Desired geographical location for the DU $n \in N_v^{du}$ (x, y).

Nodes in the VNRs have two weights $\omega_v^{prc}(n)$ and $\omega_v^{ant}(n)$ denoting, respectively, the processing capacity and the number of RF front–ends requested by the node $n \in N_v$. Notice that the overall processing requirement of the small cell m in the VNR is the sum of the processing resource requirements of its components virtual DU $m_{du} \in N_v^{du}$ and virtual CU pool $m_{cu} \in N_v^{cu}$: $\omega_v^{prc}(m) = \omega_v^{prc}(m_{du}) + \omega_v^{prc}(m_{cu})$. Each virtual DU $n \in N_v^{du}$ in the VNRs has a geographic location $loc(n)$ as x, y coordinates, which along with the location and the coverage radius of the substrate DUs is used to find the candidate substrate DUs for each requested virtual DU. Lastly, each virtual link $e^{nm} \in E_v$, which is the link connecting the two nodes $n, m \in N_v$ in VNRs, has bandwidth requirement $\omega_v^{bwt}(e^{nm})$, which can be easily derived considering the characteristics of the cells and their employed functional split option. For example, given a 20 MHz FDD LTE cell with a $2x2$ MIMO antenna configuration, and assuming that the cell employs the PHY–RF split, the required CPRI bitrate is $\approx$ 2.5 Gbps in order achieve up to 150 Mbps of users traffic [8]. Table 4.2 summarizes the VNR parameters.

Fig. 4.2b displays an example of a VNR. The request is composed of three small cells. The number of requested small cells in the VNR is equal to the number of virtual DUs. Notice that the red ones require their CUs be co–located with the same CU pool, while the blue one does not impose such a constraint. Notice also that the blue small cell requests a single CU and a lower CPRI bandwidth compared to the red small cells.

4.2.3 Fronthaul Deployment Cost Analysis

In this section, we will perform a fronthaul deployment cost analysis studying the tipping point in terms of economic viability between an optical and a mmWave wireless fronthaul.

TCO of MNOs has been growing in the last few years in conjunction with exponentially increasing data traffic demands in mobile networks. The fronthaul network deployment cost constitutes the huge share of the TCO and, therefore, requires careful design. Although, there are commercial products which provide more than 7.5 Gbps wireless CPRI links using E–band

frequency range, the optical fiber, regardless of its cost, keeps leading the market as the best fronthaul medium in terms of its bandwidth and latency characteristics. Apart from the cost factor, however, there are a number of other parameters, such as time–to–market, the physical feasibility of running wired/wireless links, which have to be taken into account when selecting the fronthaul medium. Therefore, future mobile networks are likely to leverage on both wired and wireless fronthauls.

Let us consider the network depicted in Fig. 4.2a and assume that each small cell provides coverage within a radius of 250 m. Let us also assume that a mmWave link can provide similar capacity as an optical fiber up to a distance of 500 m. The substrate network in this example has a total of 54 mmWave interfaces for delivering connectivity from DUs to the CU pool; whereas, 18 km of optical fiber would be required to build the fronthaul network, like Checko et al [131, 41], assuming that the optical fiber links are buried as straight lines from the CU pool to the DUs. Dividing the former value (54 mmWave interfaces) by the latter one (18 km) we can obtain the cost ratio starting from which deploying the mmWave links is more cost–effective than deploying fiber links. In this particular example, mmWave links must be 3 times cheaper than one km of the optical fiber in order to make the mmWave fronthaul economically viable. This cost factor reduces with an increase in the distance between the DUs and the CU pool.

4.3 CU Placement

One of the characteristics of mmWave communication is that it requires highly directional antennas. This along with the fact that mmWave Relays can possess only a few mmWave interfaces makes only a subset of substrate links as viable to be used at a given time. As an example, let us consider the CU placement shown in Fig. 4.2c. In this case, three small cells are embedded onto the substrate network. The red small cells are assumed to require two CPRI option 3 links (5 Gbps in total), which, assuming that each mmWave interface has 5 Gbps capacity, results in a single mmWave interface being required on the CU pool in order to serve the request, while two interfaces (i.e., one for serving the local small cell and one for relaying the CPRI link of the other small cells) being required on the relay DU. On the other hand, it is assumed that the blue small cell requires just a CPRI option 2 link (1.2 Gbps in total); therefore, a longer mmWave link can be used, minimizing the number of relaying nodes required to serve the request. We will further discuss the relationship between CPRI capacity and mmWave length in the evaluation section. Notice that in the alternative CU placement shown in Fig. 4.2d, the constraint on the maximum number of interfaces of the relaying DU is violated since the mapping would require four interfaces while only two are actually available.

4.3.1 Problem Statement

The CU placement problem presented in this chapter can be formally stated as follows:

Given: grid–shaped and random substrate topologies in different functional split scenarios, where each node has processing capacity, fronthaul bandwidth, and RF front–ends.

Find: the optimal mapping of the virtual nodes and links in the VNRs to, respectively, substrate nodes and links in the substrate network.

Objective: jointly minimize fronthaul bandwidth utilization and the number of active mmWave interfaces, which, translates to minimizing the network–wide power consumption in C–RAN.

Upon receiving a VNR, the InP that owns the substrate network has to decide whether to accept and map the request or to reject it. The CU placement process is composed of two steps: the node embedding and the link embedding. In the node embedding step, each node in the VNR is mapped to a substrate node; while in the link embedding step, each link is mapped to a single substrate path. In both steps, nodes and links constraints must be satisfied.

4.3.2 Dynamic ILP–based Placement Algorithm

Before formulating the ILP problem, we shall first find the candidate substrate DUs for each virtual DU in the VNR. This can be done by considering the location $loc(n')$ of the virtual DU $n' \in N_v^{du}$ along with the location $loc(n)$ and the coverage radius $\delta(n)$ of every substrate DU/Relay $n \in N_s^{du}$. For each virtual DU n', a cluster of candidate DUs $\Omega(n')$ can then be defined as follows:

$$\Omega(n') = \left\{ n \in N_s^{du} | dis(loc(n), loc(n')) \leq \delta(n) \right\} \tag{4.1}$$

It is important to mention that the signal propagation model, although important, takes a secondary role in this CU placement problem. Firstly, this is because the CU placement problem only considers VNRs whose characteristics are specified by the MVNOs making the request; thus, the actual users of MVNOs are not considered in the CU placement problem. Secondly, considering complex signal propagation models will unnecessarily complicate the embedding problem without adding any significant benefit. Therefore, the signal propagation of a cell is characterized by only its coverage radius.

This ILP formulation aims at computing the optimal CU placement by considering the available computational, fronthaul bandwidth and radio resources under a certain cost function with the ultimate goal of minimizing the power consumption in the fronthaul network. The chosen objective function is:

$$minimize \quad \sum_{e \in E_s} \sum_{e' \in E_v} \Lambda_{btw} \omega_v^{bwt}(e') \Phi_e^{e'} + \sum_{n \in N_s} \sum_{e \in E_s(n)} \Lambda_{int} \Phi_e^i \tag{4.2}$$

Table 4.3 – Binary decision variables {0, 1}

Variable	Description
$\Phi_n^{n'}$	Shows if the virtual DU $n' \in N_v^{du}$ has been mapped to the substrate DU $n \in \Omega(n')$.
Φ_e^i	Shows if the mmWave interface $i \in \omega_s^{int}(n)$ of the substrate link $e \in E_s(n)$ of the node $n \in N_s$ has been used in the virtual link mapping.
$\Phi_e^{e'}$	Shows if the virtual link $e' \in E_v$ has been mapped to the substrate link $e \in E_s$.

Table 4.3 summarizes all binary decision variable used in the ILP formulation. The first argument of the objective function minimizes the overall bandwidth consumption across all substrate fronthaul links. In other words, it minimizes the number of hops required to embed the virtual links onto the substrate paths. Whereas, the second argument minimizes the overall number of active mmWave interfaces required to host the VNRs. It is important to mention that the cost for using the fronthaul bandwidth resources (Λ_{bwt}) is negligible compared to the cost of using mmWave interfaces of the substrate nodes (Λ_{int}). This is because our goal is to reduce the power consumption in C–RAN by curtailing the number of active mmWave interfaces.

The substrate network can embed VNRs as long as the host substrate nodes and links have enough resources to support the requests:

$$\sum_{n' \in N_v^{du}} \omega_v^{prc}(n') \Phi_n^{n'} \leq \omega_s^{prc}(n) \quad \forall n \in N_s^{du} \tag{4.3}$$

$$\sum_{n' \in N_v^{cu}} \omega_v^{prc}(n') \Phi_n^{n'} \leq \omega_s^{prc}(n) \quad \forall n \in N_s^{cu} \tag{4.4}$$

$$\sum_{e' \in E_v} \omega_v^{bwt}(e') \Phi_e^{e'} \leq \omega_s^{bwt}(e) \quad \forall e \in E_s \tag{4.5}$$

$$\sum_{n' \in N_v^{du}} \omega_v^{ant}(n') \Phi_n^{n'} \leq \omega_s^{ant}(n) \quad \forall n \in N_s^{du} \tag{4.6}$$

Specifically, Constraints (4.3) and (4.4) ensure that the processing resource requested by the virtual DUs and the virtual CU pools are at most equal to the processing resource of, respectively, the host substrate DUs and CU pools. Constraint (4.5) makes sure that the overall fronthaul capacity of each substrate link is not exceeded. Finally, Constraint (4.6) deals with the RF front–ends, making sure that the number of RF front–ends required at the host substrate node is at most equal to the number of RF front–ends available at that substrate node.

Each node in the VNR must be mapped only once:

$$\sum_{n \in N_s} \Phi_n^{n'} = 1 \quad \forall n' \in N_v \tag{4.7}$$

Each virtual DU in the VNR must be mapped only on a candidate substrate DU:

$$\sum_{n \in N_s^{du} \setminus \Omega(n')} \Phi_n^{n'} = 0 \quad \forall n' \in N_v^{du} \tag{4.8}$$

At each substrate node, the sum of the substrate mmWave interfaces that are used to carry the traffic of the virtual links must be less or equal to the number of mmWave interfaces available at that node:

$$\sum_{e^{ij} \in E_v} \Phi_{e^{nm}}^{e^{ij}} + \sum_{e^{ij} \in E_v} \Phi_{e^{mn}}^{e^{ij}} \leq |\omega_s^{int}(n)| \quad \forall n, m \in N_s, \quad \forall e^{nm} \in E_s \tag{4.9}$$

The following constraint enforces for each virtual link $e^{nm} \in E_v$ to be a continuous path between the pair of physical nodes on top of which the virtual nodes $n, m \in N_v$ have been mapped:

$$\sum_{e \in E_{\mathbf{s}}^{\star i}} \Phi_e^{e^{nm}} - \sum_{e \in E_{\mathbf{s}}^{i\star}} \Phi_e^{e^{nm}} = \begin{cases} -1 & \text{if } i = n \\ 1 & \text{if } i = m \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in N_s, \quad \forall e^{nm} \in E_v \tag{4.10}$$

where $E_s^{\star i}$ is the set of the fronthaul links that originate from any node and directly arrive at the node $i \in N_s$, while $E_s^{i\star}$ is set of the fronthaul link that originates from the node $i \in N_s$ and arrive at any node directly connected to i.

Finally, in order to minimize the number of mmWave interfaces (i.e., second argument in the objective function) that are being used to deliver connectivity from the host DUs to the host CU pools, the last constraint guarantees that the mmWave interfaces can be reused. Notice that the interface $i \in \omega_s^{int}(n)$ of the substrate link $e \in E_s(n)$ of the node $n \in N_s$ is used in mappings (i.e., $\Phi_e^i = 1$) if at least one virtual link has been mapped onto the substrate path that is using the interface i.

$$\sum_{e' \in E_v} \Phi_e^{e'} - \mu_b \Phi_e^i \leq 0 \quad \forall n \in N_s, \quad \forall e \in E_s(n), \quad \forall i \in \omega_s^{int}(n) \tag{4.11}$$

where μ_b is a big positive number. If the interface i has not been used to map a virtual link ($\sum_{e' \in E_v} \Phi_e^{e'} = 0$) then that interface will not be selected ($\Phi_e^i = 1$ is ruled out) since the objective function, apart from minimizing the fronthaul bandwidth consumption, also aims at minimizing the number of active mmWave interfaces.

4.3.3 Scalable Dynamic Placement Heuristic

The introduced ILP–based placement algorithm has limited scalability, and therefore, it is not applicable to big–sized networks. For example, it may take around one day to embed a VNR that has 4 nodes (1 CU pool and 3 DUs) over a grid–shaped substrate network composed of 49 nodes (2 CU pools and 47 DUs) on Intel Core i7 laptop (3.0 GHz CPU, 16 Gb RAM) using the Matlab® ILP solver (intlinprog). In this section, a heuristic is presented that is able to embed similar requests and find near–optimal embedding solutions in less than 10 milliseconds.

The proposed heuristic consists of three steps (see pseudocode in Alg. 1). Let $m_{du} = |N_s^{du}|$ and $m_{cu} = |N_s^{cu}|$ be the number of, respectively, substrate DUs and substrate CU pools with $m = m_{du} + m_{cu}$. Similarly, let $n_{du} = |N_v^{du}|$ and $n_{cu} = |N_v^{cu}|$ be the number of, respectively, virtual DUs and virtual CU pools. Lastly, let $k = |E_s|$ be the number of edges available in the substrate network. In the first step, the heuristic selects a list of candidate substrate DUs and candidate substrate CU pools, respectively, for each virtual DU $n \in N_v^{du}$ and each virtual CU pool $m \in N_v^{cu}$ in the VNR. Specifically, the candidate substrate DUs are selected based on the locations of the substrate and the virtual DUs, and based on the availability of the required RF front–ends and the processing resources at the substrate DUs (lines from 3 to 12 in the pseudocode). Whereas, the candidate substrate CU pools are selected by considering only the processing resource requirement of the virtual CU pools (lines from 13 to 19 in the pseudocode). Step 1 requires $O(n_{du}m_{du} + n_{cu}m_{cu})$ time.

Algorithm 1 Nodes and links assignment

```
1:  procedure EmbedRequest(G_s, G_v)
2:      Step 1: Compute list of candidates.
3:      for n ∈ N_v^du do                                   ▷ Virtual DUs.
4:          for p ∈ N_s^du do                               ▷ Substrate DUs.
5:              d ← dis(loc(n), loc(p))                     ▷ Distance in meters.
6:              if d ≤ δ(p) then
7:                  if ω_v^ant(n) ≤ ω_s^ant(p) and ω_v^prc(n) ≤ ω_s^prc(p) then
8:                      candidates(n) ← p
9:                  end if
10:             end if
11:         end for
12:     end for
13:     for m ∈ N_v^cu do                                   ▷ Virtual CU pools.
14:         for q ∈ N_s^cu do                               ▷ Substrate CU pools.
15:             if ω_v^prc(m) ≤ ω_s^prc(q) then
16:                 candidates(m) ← q
17:             end if
18:         end for
19:     end for
```

20: *Step 2: Perform CU placement.*
21: **for** $n \in N_v^{cu}$ **do** ▷ Virtual CU pools.
22: **for** $i \in N_s$ **do** ▷ Initialize mapping cost array.
23: $m_c(i) \leftarrow 0$
24: **end for**
25: **for** $p \in candidates(n)$ **do**
26: **for** $m \in neighbors(n)$ **do**
27: $c_{curr} \leftarrow +\infty$
28: **for** $q \in candidates(m)$ **do**
29: $c_{new} \leftarrow \sum_{e \in P_s(p,q)} \omega_v^{bwt}(e^{nm})$
30: $c_{curr} \leftarrow min(c_{curr}, c_{new})$
31: **end for**
32: $m_c(p) \leftarrow c_{curr}$ ▷ Accumulate mapping cost.
33: **end for**
34: **end for**
35: $p \leftarrow argmin(m_c(p))$
36: $mapped(n) \leftarrow p$
37: **end for**
38: *Step 3: Perform RF front–ends embedding.*
39: **for** $n \in N_v^{cu}$ **do** ▷ Virtual CU pools.
40: $p \leftarrow mapped(n)$
41: **for** $m \in neighbors(n)$ **do**
42: **for** $i \in N_s$ **do** ▷ Initialize mapping cost array.
43: $m_c(i) \leftarrow 0$
44: **end for**
45: **for** $q \in candidates(m)$ **do**
46: $m_c(q) \leftarrow \sum_{e \in P_s(p,q)} \omega_v^{bwt}(e^{nm})$
47: **end for**
48: $q \leftarrow argmin(m_c(q))$
49: $mapped(m) \leftarrow q$
50: *Allocate path* $P_s(p,q)$
51: **end for**
52: **end for**
53: **end procedure**

In the second step, for each candidate substrate CU pool $p \in candidates(n)$ of each virtual CU pool $n \in N_v^{cu}$, the heuristic considers all neighbor nodes (i.e., the rest of the nodes in the VNR) of the virtual node n. Then, the heuristic computes the cost of embedding each virtual node pair n, m. In essence, this is the cost of embedding the virtual link e^{nm} onto the path between the candidate CU pool $p \in candidates(n)$ and the candidate DU $q \in candidates(m)$. The heuristic then assigns the virtual node n to the substrate node p with the lowest mapping cost (lines from 21 to 37 in the pseudocode). Here, the goal is to place the virtual CU pool on the substrate CU pool that can support all its virtual DUs at the minimal cost. This process requires $O(n_{cu} m_{cu} n_{du} (m_{du} - 1) k \log_{10} m)$ time.

In the last step, the heuristic loops over the virtual CU pools, considers the neighbor nodes $neighbor(n)$ (i.e., the virtual DUs) of each virtual CU pool $n \in N_v^{cu}$ then maps each virtual DU $m \in neighbor(n)$ to its candidate substrate DU $q \in candidates(m)$ that has the lowest mapping cost. Once the virtual DU has been mapped, using Dijkstra's shortest path algorithms, the heuristic allocates the shortest path $P_s(p,q)$ between the nodes $p,q \in N_s$, which are the ones that have hosted, respectively, the virtual CU pool n and the virtual DU m (lines from 39 to 50 in the pseudocode). This results in the virtual nodes of the VNR being placed close to each other in the substrate network, and therefore, entails an efficient utilization of the substrate resources. Step 3 requires $O(n_{cu}n_{du}(m_{du}-1)k\log_{10} m)$ time. Thus, the overall time complexity of the heuristic is $O(n_{du}m_{du}+n_{cu}m_{cu}+[n_{du}n_{cu}(m_{du}-1)k\log_{10} m](1+m_{cu}))$.

Finally, it is important to mention that, in order to ensure the correctness of the results, we pass all the solutions found by the heuristic through the same constraints defined for the ILP formulation.

4.4 Evaluation

In this section, we compare[2] the performance of the ILP–based placement algorithm with the performance of the heuristic using different synthetic substrate networks and different VNRs. Initially, we describe the used simulation environment. We then report on the outcomes of the numerical simulations conducted in a discrete event simulator implemented in Matlab.

4.4.1 Simulation Environment

The simulation parameters and the substrate network characteristics are derived from a number of studies on mmWave communications. Rappaport et al [132] suggests that optimum coverage be achieved by having 200 m of distance between each DU, while Pi et al [133] estimates 1 km to be the typical coverage radius for mmWave links in line–of–sight conditions. Finally, Ghosh et al [134] and Rangan et al [135] rely on empirical measurements to show that bitrates as high as 10 Gbps can be achieved in the mmWave band with an outage probability of $\approx 11\%$, while 5 Gbps of bitrate can be achieved with an outage probability of $\approx 3\%$.

The ILP–based algorithm and the heuristic are evaluated using grid–shaped and random substrate network topologies. The former is similar to the one depicted in Fig. 4.2a and is composed of 21 nodes with a uniform inter–node distance of 500 m. Whereas the latter, as the name implies, is generated by randomly positioning the same number of nodes in the area of 4 km^2. As opposed to the grid–shaped substrate network topology, in the random substrate network topology the nodes may have more/fewer neighbors, and therefore, may also have more/fewer link embedding opportunities. The random network is generated with the goal of mimicking the real–life mobile network deployment scenarios such as the ones deployed in office or residential areas. In order to make a fair comparison between those two topologies,

[2] Note that we do not compare the proposed heuristic algorithm with any of the state-of-the-art algorithms since the problem formulations are different and the comparison would be unfair.

Table 4.4 – CPRI link bandwidth per option.

CPRI option	CPRI rate	I/Q sampling data rate	LTE configurations
1	600 Mbps	400 Mbps	10 MHz, $1x1$SISO
2	1.2 Gbps	0.9 Gbps	20 MHz, $1x1$SISO
3	2.5 Gbps	1.8 Gbps	20 MHz, $2x2$MIMO
5	5 Gbps	3.6 Gbps	20 MHz, $4x4$MIMO

we keep the total number of mmWave interfaces constant across the two types of substrate network topologies. In both cases, the following functional splits are considered: PHY–RF split, PHY split[3], MAC split, and PDCP–RLC split[4]. A detailed description of the considered functional splits is provided in Section 3.5.1.

When generating the substrate topologies, it is assumed that there is a LoS between substrate nodes and that up to 500 m of inter–node distance the link can support up to 2.5 Gbps of traffic, while up to 1000 m of inter–node distance the link can deliver up to 1.2 Gbps of traffic. Table 4.4 summarizes some of the most common CPRI options for the PHY–RF split. It can be observed that the aforementioned assumptions correspond to a CPRI option 3 and to a CPRI option 2 configurations, respectively.

Each node in the substrate networks is either a DU/Relay or a CU pool. Depending upon the functional split used in the substrate network, however, both of them may possess processing units. In the case of the PHY–RF split in the substrate network, for example, the DU $n_{du} \in N_s^{du}$ of the substrate small cell n does not possess processing capacity, while in the case of the PHY split or the MAC split, the DU does possess processing capacity $\omega_s^{prc}(n_{du}) = 0.5 \cdot \omega_s^{prc}(n)$ and $\omega_s^{prc}(n_{du}) = 0.7 \cdot \omega_s^{prc}(n)$, respectively, where $\omega_s^{prc}(n)$ is the overall processing capacity of the substrate small cell n. Note that, for example, in the case of the PHY split, it is assumed that the half of the processing capacity is allocated to the DUs while the other half is allocated to the CU pools. This is because the most processor–hungry procedure (i.e., FFT/IFFT) is taking place in the PHY layer. The processing requirement increases at the DUs and decreases at the CU pools when fewer layers (e.g., PHY layer, MAC layer) are centralized at the CU pools (see Table 4.5). The overall processing capacity of each substrate network is computed considering the number of embeddings, the average number of DUs in a VNR and the functional split option employed in the substrate network.

The number of CU pools, which are randomly deployed over the substrate networks, varies between 1 and 4. For the sake of fair comparison, the same quantity of CU pools with the same locations are selected for virtual network embedding problems for different substrate networks with different functional splits. DU relays at the edges of the network are equipped with a single mmWave interface and 4 RF front–ends. Whereas, the rest of the DUs are equipped with

[3]Note that the PHY split corresponds to the PHY II split in Figure 3.1

[4]For the sake of improving readability, only the results of the first three splits are reported. The results of the PDCP–RLC split resemble the ones of MAC split and are thus omitted.

Table 4.5 – Fronthaul bandwidth and processing requirements for the DU and the CU pool of the small cell n for the functional splits considered in this work.

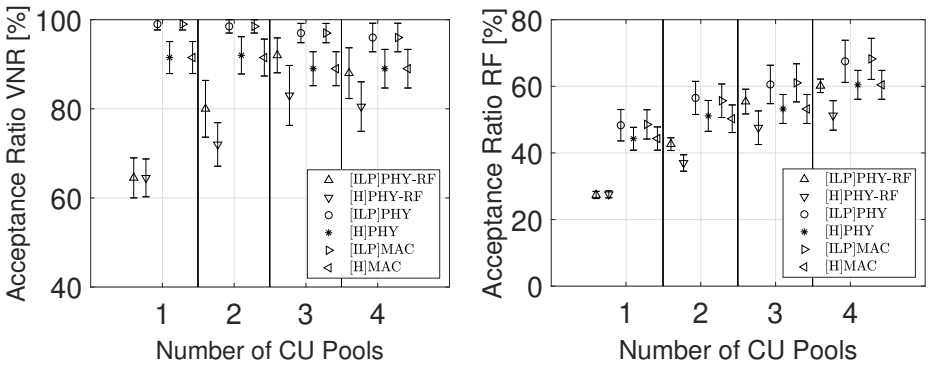
Resources / Splits	Processing capacity		DL bandwidth	
	DU	**CU**	**1x1SISO**	**2x2MIMO**
PHY–RF Split	$0 \cdot \omega_v^{prc}(n)$	$1 \cdot \omega_v^{prc}(n)$	1.23 Gbps	2.46 Gbps
PHY Split	$0.5 \cdot \omega_v^{prc}(n)$	$0.5 \cdot \omega_v^{prc}(n)$	0.46 Gbps	0.93 Gbps
MAC Split	$0.7 \cdot \omega_v^{prc}(n)$	$0.3 \cdot \omega_v^{prc}(n)$	0.07 Gbps	0.15 Gbps
PDCP–RLC Split	$0.9 \cdot \omega_v^{prc}(n)$	$0.1 \cdot \omega_v^{prc}(n)$	0.07 Gbps	0.15 Gbps

4 mmWave interfaces and 8 RF front–ends. Lastly, CU pools are equipped with 10 mmWave interfaces and do not possess RF front–ends.

VNRs consist of star–shaped networks similar to the ones shown in Fig. 4.2b. The number of small cells per VNR and the number of RF front-ends per small cell are randomly selected within the set of $\{1,2,3,4\}$ and $\{1,2\}$, respectively. The fronthaul bandwidth requirement instead depends on the functional split of the VNR. For example, if the PHY–RF split is considered, each of the RF front–ends may require either a CPRI option 2 or a CPRI option 3 link. The overall processing requirement of a VNR is equal to the number of requested DUs. Whereas, the processing requirements of individual nodes (i.e., a virtual DU or a virtual CU pool) depend upon the considered functional split. For example, if the PHY split or the MAC split are considered, the processing requirements of the virtual DU $n_{du} \in N_v^{du}$ would be, respectively, $\omega_v^{prc}(n_{du}) = 0.5 \cdot \omega_v^{prc}(n)$ and $\omega_v^{prc}(n_{du}) = 0.7 \cdot \omega_v^{prc}(n)$ where $\omega_v^{prc}(n)$ is the overall processing requirement of the virtual small cell n. Whereas, the remaining $\omega_v^{prc}(n_{cu}) = 0.5 \cdot \omega_v^{prc}(n)$ and $\omega_v^{prc}(n_{cu}) = 0.3 \cdot \omega_v^{prc}(n)$ would be the processing requirements of the virtual CU pool $n_{cu} \in N_v^{cu}$, respectively, in the PHY split and the MAC split cases. The fronthaul bandwidth requirements for different antenna configurations as well as the processing resource requirements at both DUs and CU pools for the considered functional splits can be found in Table 4.5. Note that the fronthaul bandwidth requirements are derived for a 20 MHz LTE channel. More details can be found in [8].

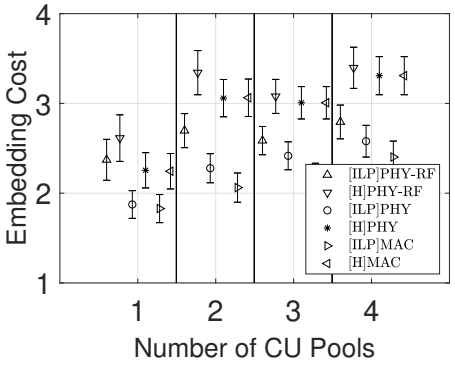
In this study, it is assumed that a fixed number of VNRs are embedded sequentially. It is important to mention that with the arrival of a new VNR, all the previously embedded requests are re–computed along with the new request (global network optimization is performed). The re–embedding can also be triggered based on the daytime versus the nighttime traffic requirement variation. Thus, a dynamic virtual network embedding is considered.

In order to make a fair comparison between the different splits, the same VNRs are used for all the functional split cases. Specifically, 200 (10 simulations each with 20 embeddings) random VNRs are generated and used by both the ILP–based algorithm and the heuristic for different functional split cases in the grid–shaped and the random substrate networks. Reported results are the average of 10 simulations.

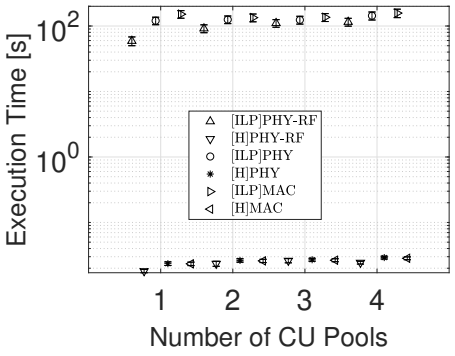


(a) Acceptance Ratio [VNR].

(b) Acceptance Ratio [RF].



(c) Embedding Cost.



(d) Execution Time.

Figure 4.3 – Performance of the ILP–based algorithm and of the heuristic with a different number of substrate CU pools for the considered functional splits.

4.4.2 Simulation Results

Figure 4.3 displays the acceptance ratio, the embedding cost and the execution time of the ILP–based CU placement algorithm and of the placement heuristic for different functional splits in the grid–shaped [5] substrate network. The acceptance ratio is computed in two ways: based on the number of accepted VNRs illustrated in Fig. 4.3a and based on the number of hosted RF front–ends illustrated in Fig. 4.3b. We remind the reader that VNRs can have a variable number of DUs each requesting a variable number of RF front–ends. As expected, in both figures, the ILP–based algorithm achieves a higher acceptance ratio compared to the one of the heuristic in all functional split cases.

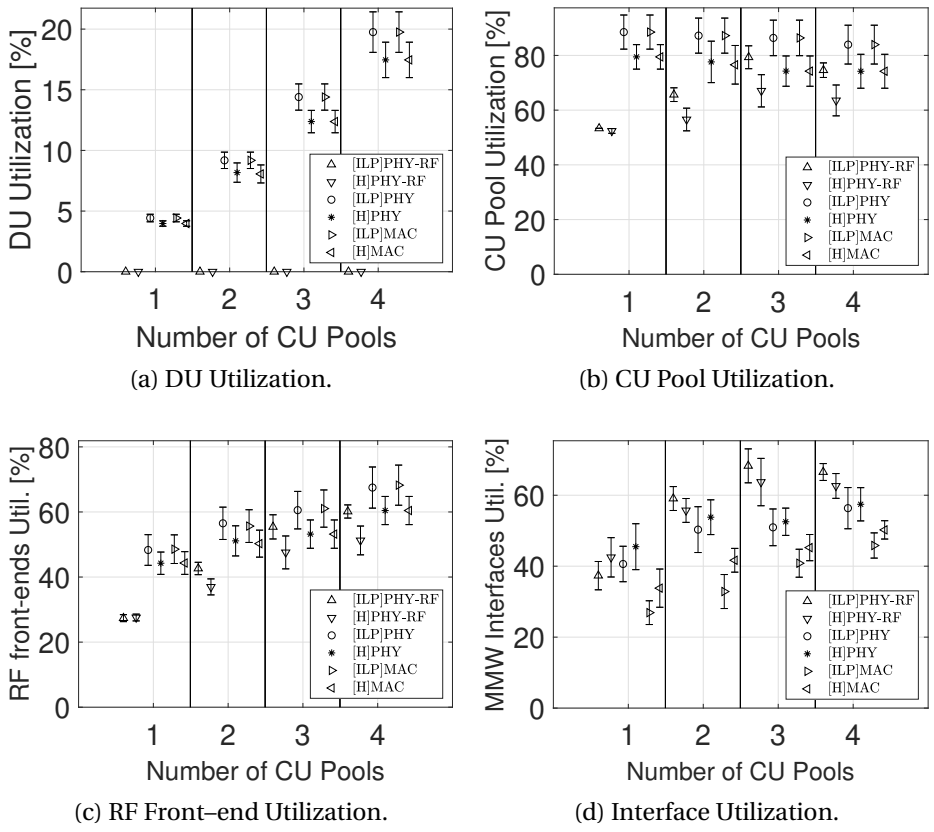
More specifically, a saturation point in the acceptance ratio can be observed for the PHY–RF split in Fig. 4.3a when there are three CU pools in the substrate network. Initially, when there is one CU pool in the substrate network, the acceptance ratio for both algorithms is low (65%) for the PHY–RF split since compared to the rest of the splits it has much higher fronthaul bandwidth requirement (see Table 3.2), which becomes a bottleneck for accepting

[5] The performance results of the random topology are similar to the ones of the grid topology and are, therefore, omitted.

more VNRs. Since, as it is mentioned in Section 4.4.1, CU pools have more mmWave interfaces than DUs, deploying more CU pools in the substrate network increases the network–wide fronthaul bandwidth capacity, which leads to a higher acceptance ratio. We can observe, however, that the acceptance ratio for the PHY–RF split reduces when the number of CU pools is four. This can be explained by the fact that at this point the RF front–ends start to become a bottleneck for accepting more request. Because of the same reason, we can also observe that the acceptance ratio for the rest of the splits reduces when there is more than one CU pool in the substrate network. Notice that as opposed to the PHY–RF split, the fronthaul capacity never becomes a bottleneck for the PHY and MAC splits in order to accept more VNRs. This is justified by the fact that compared to the PHY–RF split those splits have significantly lower fronthaul bandwidth requirements (see Table 3.2).

It is interesting to note that, even though the acceptance ratio for the PHY–RF split (see Fig. 4.3a) decreases when four CU pools are available in the substrate network, the acceptance ratio in terms of the on–boarded RF front–ends (see Fig. 4.3b) increases for the same split for the same number of CU pools in the substrate network. This is because substituting more DU nodes with CU pools curtails the overall number of DU nodes in the substrate network, which in turn curtails the number of RF front–ends that can host requested RF front–ends since in the considered scenario only DUs possess RF front–ends. As a result, the remaining substrate DUs become heavily utilized, and therefore, the acceptance ratio of the RF front–ends increases across the entire substrate network. Notice also that there is a significant difference between the acceptance ratios in Fig. 4.3a and Fig. 4.3b. This is because the substrate DU nodes possess redundant RF front–ends in order to support the extreme case in which each virtual DU of the virtual networks might request two RF front–ends, which is the maximum number of RF front–ends that a virtual DU can request. As it has been mentioned, however, each virtual DU selects the number of RF front–ends randomly between one and two.

Figure 4.3c plots the average embedding cost that is defined as the average number of mmWave interfaces that are used to map a virtual link onto a substrate path. It can be observed that the ILP–based embedding algorithm can on–board more requests than the heuristic while ensuring a lower embedding cost. This means that the ILP–based algorithm employs the mmWave interfaces more efficiently than the heuristic. It can also be observed that for all the functional splits, the embedding cost manifests an increasing trend with more number of CU pools in the substrate network. This is due to the fact that compared to the DUs, CU pools possess more mmWave interfaces, which leads to more link embedding opportunities, therefore increasing the embedding cost. Lastly, it can be seen that the embedding cost for the PHY–RF split is always higher compared to the cost of the rest of the splits, regardless of the quantity of the CU pools in the substrate network. Similarly, the embedding cost for the PHY split is always higher than the one of the MAC split. This stems from the fact that the higher–layer is the functional split, the less is the fronthaul bandwidth requirement, and therefore, more virtual links are mapped to the same mmWave interface. Thus, the lower–layer is the functional split, the cheaper is the mmWave links to be used.



(a) DU Utilization.

(b) CU Pool Utilization.

(c) RF Front–end Utilization.

(d) Interface Utilization.

Figure 4.4 – Substrate resource utilization of the ILP–based algorithm and of the heuristic.

Figure 4.3d illustrates the average amount of time required in order to embed a *single* VNR by using the ILP–based placement algorithm and the placement heuristic. It can be seen that the embedding time for the ILP–based algorithm is significantly higher compared to the embedding time of the heuristic. The problem with the ILP–based algorithm is that it becomes computationally intractable when a substrate network with a few tens of nodes is considered, while the heuristic is able to effectively map complex VNRs on the substrate network with a few hundreds of nodes in a limited amount of time. Even though for the sake of finding the optimal CU placement MNOs may readily wait even several weeks, we argue that the proposed heuristic, although less efficient than the ILP–based placement algorithm, allows a faster service on–boarding time, while the ILP–based placement algorithm could be employed to periodically optimize network configuration.

Figure 4.4 displays the resource utilization of the substrate network for the ILP–based algorithm and the heuristic. Specifically, Fig. 4.4a and Fig. 4.4b illustrate the processing resource utilization of substrate DUs and CU pools, respectively. In both figures, it can be observed that the resource utilization of the ILP–based algorithm is higher than the one of the heuristic. While the processing resource utilization at the CU pools (see Fig. 4.4b) resembles the acceptance ratio (see Fig. 4.3a), the picture is totally different in the processing resource utilization at the DUs (see Fig. 4.4a). For the PHY–RF split, the DU utilization is zero since in that case the

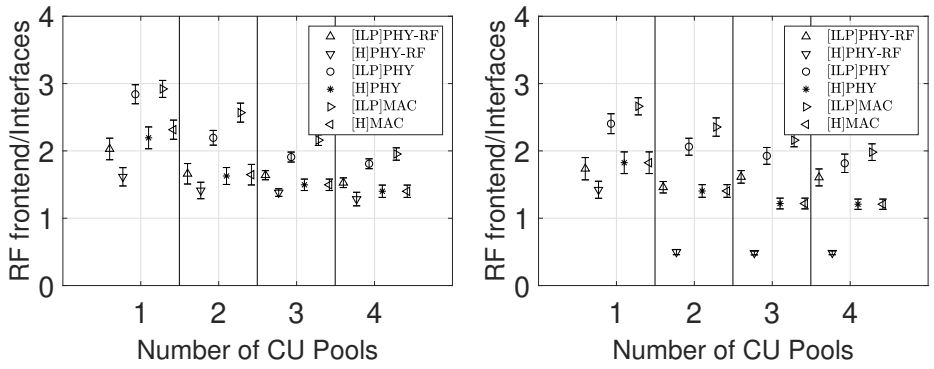
DUs do not possess processing resource. For the other two splits instead, the DU utilization as well as RF front–end utilization (see Fig. 4.4c) increases with an increase in the number of CU pools in the substrate network. This is justified by the fact that increasing the number of CU pools in the substrate network curtails the overall number of DU nodes in the network; therefore, the remaining DUs become much more loaded.

Figure 4.4d shows the network–wide mmWave[6] interface utilization. It can be observed that the utilization increases for all the splits for both algorithms with the increase in the number of CU pools. It can also be observed that compared to the heuristic algorithm, in the case of using the ILP–based algorithm, the mmWave interface utilization for the PHY and MAC splits is lower, regardless of the number of CU pools. This is an evidence of the fact that the ILP–based algorithm exploits the mmWave interfaces in a more efficient manner than the heuristic. The picture is similar for the PHY–RF split only when there is one CU pool in the substrate network. When there are more than one CU pool in the substrate network, the mmWave interface utilization is more in the case of the ILP–based algorithm compared to its heuristic counterpart since, as it can be seen in Fig. 4.3a, the ILP–based algorithm accepts a significantly more number of VNRs. It is worthwhile to note that although the acceptance ratio increases when fewer resources are centralized (i.e., from the PHY–RF split towards the MAC split), the mmWave interface utilization reduces due to the fact that already used mmWave interfaces are reused more frequently, avoiding to power up new mmWave interfaces.

In order to compare different substrate topologies, let us analyze Fig. 4.5a and Fig. 4.5b that show the RF front–end and mmWave interface utilization ratios for, respectively, the grid–shaped and the random substrate networks. The higher is the ratio, the more beneficial is the embedding; in other words, more interfaces are being reused to map VNRs. The superiority of the ILP–based algorithm over the heuristic can be observed in terms of a higher ratio in both grid and random substrate topologies. The results in both plots are somewhat similar in terms of their values and fluctuating behavior. On the one hand, increasing the number of CU pools in the substrate network increases the utilization of the RF front–ends because of the lack of DUs. On the other hand, it increases the interface utilization up to its saturation point (i.e., when the number of CU pools is three). Therefore, the optimal RF front–ends/mmWave interfaces ratio varies and depends on the actual functional split used at the RAN.

In order to better understand how the substrate resources are utilized during the embedding process, let us now analyze a single iteration of the simulation in detail. For the sake of improving the readability, only two splits are shown: the PHY split and the MAC split for the DU utilization figures, while the PHY–RF split and the MAC split for the rest of the figures. We remind the reader that each iteration consists of 20 randomly generated VNRs. As it can be seen in Fig. 4.6, the utilization of the substrate resources in the case of employing the ILP–based placement algorithm is higher compared to the resource utilization of the heuristic, regardless of the number of CU pools in the substrate network. This is justified by the fact that the ILP–based placement algorithm is able to embed more requests than the heuristic.

[6]The evaluation of the mmWave connection is out of the scope of this work.



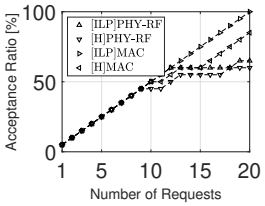
(a) RF Front–ends/mmWave Interfaces Ratio [grid].

(b) RF Front–ends/mmWave Interfaces Ratio [random].

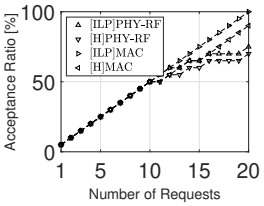
Figure 4.5 – The utilization ratio of the RF front–ends and mmWave interfaces of the ILP–based placement algorithm and the placement heuristic for the grid and the random networks with different number of substrate CU pools.

Figure 4.6a, 4.6b, 4.6c and 4.6d show the acceptance ratio for a different number of CU pools. This ratio is calculated as the percentage of the accepted requests out of 20 VNRs (a single iteration). As expected, the acceptance ratio of the PHY–RF split is smaller than the acceptance ratio of the MAC split for both algorithms, regardless of the number of CU pools in the substrate network. This is because the latter split has 16.5 times smaller fronthaul bandwidth requirement compared to the former one (see Table 3.2). Thus, the higher–layer splits, such as the MAC split, are always beneficial in terms of fronthaul resource requirements, and therefore, are more energy–efficient and enable more VNRs to be embedded in the substrate network. We can observe that for all the splits the ILP–based algorithm accepts more VNRs than its heuristic counterpart.

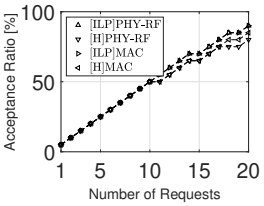
Figures 4.6e, 4.6f, 4.6g and 4.6h resemble Fig. 4.3b and Fig. 4.4a. The DU utilization increases with an increase in the number of CU pools. As we have already mentioned, the reason for this is that the number of CU pools is increased at the expense of the number of DUs in the substrate network. Figures 4.6i, 4.6j, 4.6k and 4.6l show the utilization of the CU pools. It is worthwhile to note that, for example, when there is one CU pool in the substrate network, both the ILP–based placement algorithm and the placement heuristic embed all the VNRs up to 10 for all the splits. Therefore, the DU and the CU pool utilizations are the same for all of them, though the CU pool processing requirements of the PHY–RF split and the MAC split are different. This is because the same requests, with different processing requirements, are used for different functional split cases, and the processing resources of the CU pools are picked based on the split option that the embedding is considered for. We can observe that the CU pool utilization pattern resembles the acceptance ratio of the VNRs (see Fig. 4.3a). Initially, when there is only one CU pool in the substrate network, the CU pool utilization for the MAC split for both algorithms is much higher compared to the ones of the PHY–RF split. This can be explained by the fact that compared to the PHY–RF split, when the MAC split is employed, many more VNRs are accepted by the substrate network. As the number of CU pools increases



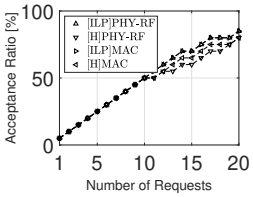
(a) Acceptance Ratio (1 CU pool).



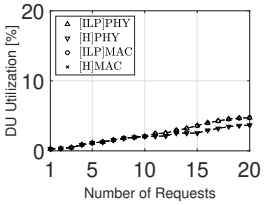
(b) Acceptance Ratio (2 CU pool).



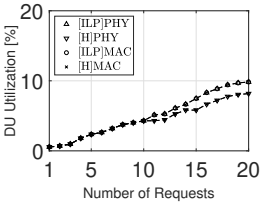
(c) Acceptance Ratio (3 CU pool).



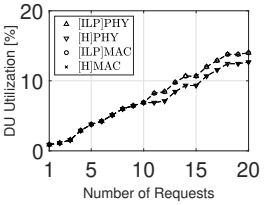
(d) Acceptance Ratio (4 CU pool).



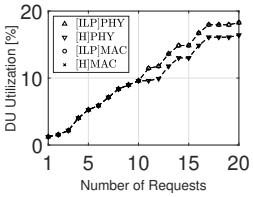
(e) DU Utilization (1 CU pool).



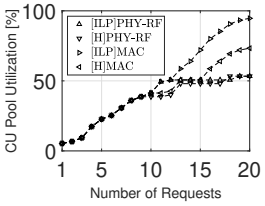
(f) DU Utilization (2 CU pool).



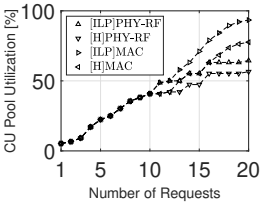
(g) DU Utilization (3 CU pool).



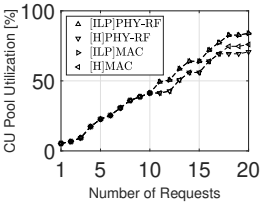
(h) DU Utilization (4 CU pool).



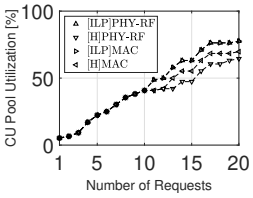
(i) CU Pool Utilization (1 CU pool).



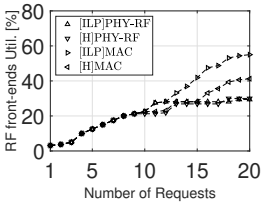
(j) CU Pool Utilization (2 CU pool).



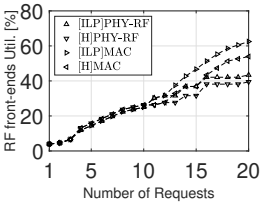
(k) CU Pool Utilization (3 CU pool).



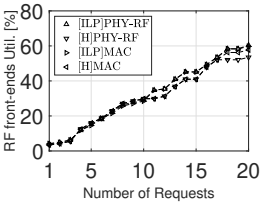
(l) CU Pool Utilization (4 CU pool).



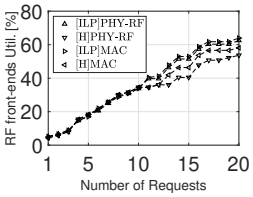
(m) RF Front–end Utilization (1 CU pool).



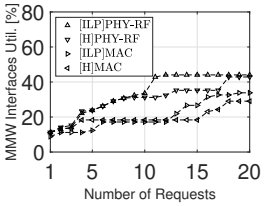
(n) RF Front–end Utilization (2 CU pool).



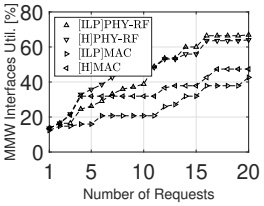
(o) RF Front–end Utilization (3 CU pool).



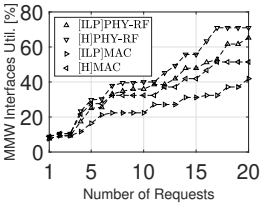
(p) RF Front–end Utilization (4 CU pool).



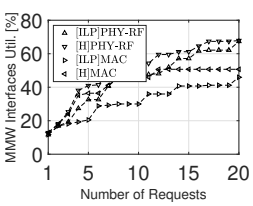
(q) Interfaces Utilization (1 CU pool).



(r) Interfaces Utilization (2 CU pool).



(s) Interfaces Utilization (3 CU pool).



(t) Interfaces Utilization (4 CU pool).

Figure 4.6 – Performance of the ILP–based placement algorithm and of the placement heuristic with a different number of substrate CU pools for one simulation (20 embeddings) for the grid–shaped substrate network.

up to three, the gap in the CU pool utilization between the MAC split and the PHY–RF split shrinks for both algorithms. This stems from the fact that the acceptance ratio increases for the PHY–RF split and decreases for the MAC split when two or three CU pools are deployed in the substrate network. Whereas, when there are four CU pools in the substrate network, this gap increases since apart from the MAC split, the acceptance ratio also reduces for the PHY–RF split as a result of insufficient substrate RF front–ends. A similar behavior can be observed in the RF front–ends utilization graphs (see Figures 4.6m, 4.6n, 4.6o and 4.6p).

Finally, Figures 4.6q, 4.6r, 4.6s and 4.6t plot the utilization of mmWave interfaces for one iteration. When the number of CU pools in the substrate network is one or two, the difference in the acceptance ratio for both splits for the ILP–based algorithm is significantly higher than for the heuristic. Therefore, in those cases, the ILP–based algorithm has a higher mmWave interface utilization ratio than the heuristic. As opposed to the previous case, when the number of CU pools in the substrate network is three or four, the difference in the acceptance ratio of both splits between the ILP–based algorithm and the heuristic is negligible. Therefore, in those cases, the ILP–based algorithm has a lower mmWave interface utilization ratio than the one of the heuristic. This advocates that the ILP–based algorithm is able to utilize the mmWave interfaces more efficiently than its heuristic counterpart.

5 Flexible Functional Split in C–RAN

This chapter discloses the motivation for the flexible functional split selection approach at small cells and discusses its benefits and challenges. Specifically, we first propose CU placement algorithms to flexibly select the appropriate functional split for each small cell. The CU placement problem is formulated using ILP techniques, having the objective of jointly minimizing the ICI and the fronthaul bandwidth utilization by dynamically selecting the appropriate functional split. Specifically, dynamic and static ILP–based algorithms are proposed to find the optimal solutions, while dynamic and static heuristics are proposed to address the scalability problem of the ILP–based algorithms in dense and ultra–dense mobile networks, respectively. We then draw a comparison between flexible functional split and the classical PHY–RF split in C–RAN.

5.1 Overview

The C–RAN and D–RAN architectures are two extreme concepts, both with pros and cons. In fact, while D–RAN requires relatively low backhaul capacity, it does not allow for joint signal processing. Conversely, C–RAN enables joint signal processing techniques, such as CoMP algorithms, however, at the expense of stringent backhaul requirements (e.g., bandwidth, latency). In order to tackle the aforementioned challenges, a number of intermediate functional splits, each characterized by a different demarcation point between DUs and CUs, have been proposed. Various criteria have to be considered in order to select the appropriate functional split. Following the current galloping pace in the mobile data traffic demand, it is our standpoint that implementing a certain fixed functional split is not a viable solution in the long run. Therefore, considering the mobile traffic demand and the daily traffic variations, the flexibility of dynamically choosing the optimal functional split is essential in order to efficiently employ the fronthaul bandwidth and baseband processing resources.

In Chapter 4, we study a CU placement considering different functional splits in the networks. Static functional splits, however, are considered in that work. In other words, once a small cell started employing a particular split option, there is no possibility for the small cell to employ another split. As it has been already mentioned, this approach has its limitations, especially in the long run. The work presented in this chapter instead addresses this issue, examining the possibility of flexible functional split change at the small cells.

Flexible functional split allows reaping the benefits of different functional splits by enabling MNOs to flexibly change the split option based on their needs. For example, since different functional splits are characterized by significantly different fronthaul bandwidth and latency requirements, MNOs can support specific QoS settings for each offered service (e.g., low latency, high throughput). Additionally, flexible functional split enables MNOs to support specific user density and load demand in each geographical area.

The main contribution of the work presented in this chapter is fourfold:

- First, we formalize a CU placement problem as a VNE problem in 5G networks supporting different functional split options. ILP techniques are used to formulate the VNE problem, in which VNRs are received from MVNOs and are embedded by the InP, having an objective of jointly minimizing the network–wide ICI *and* the fronthaul bandwidth utilization by dynamically selecting the appropriate functional split option at each small cell.

- Second, we propose dynamic and static ILP–based algorithms, and scalable dynamic and static heuristics to solve the CU placement problem, discussing the pros and cons and the applicability of each algorithm in different mobile network scenarios.

- Third, we apply the flexible functional split approach to a star, a ring and a tree fronthaul topologies, highlighting the topology–dependent benefits of the proposed approach.

- Fourth, we compare the flexible functional split with the classical static PHY–RF split, discussing the advantages and the challenges of the former approach.

5.2 Network Model

This section details the substrate and the virtual network models. Figure 5.1 depicts the reference network architecture used in this work. The main idea of this figure is to show that different functional splits can co–exist at the same network and can be dynamically changed. In the lower left part of the figure we can see the traditional D–RAN architecture in which the DU and the CU are deployed in close proximity. As opposed to the D–RAN case, for all the other functional splits, the DUs are decoupled from the CUs, and an optical[1] fronthaul is used for their interconnection. It is important to say that, regardless of the split option being employed at the given time, apart from the CU pool, also all the DUs possess processing capabilities since we consider a network in which the functional split options can be dynamically changed. For example, although in the case of employing the PHY–RF split there is no baseband signal processing at the DUs, the DUs are still required to have baseband signal processing capabilities in order to be able to support the other splits if necessary.

[1] Optical fiber, as the most common fronthauling option, has been selected as the fronthaul medium. However, other fronthauling options such as millimeter wave wireless links or copper links are also possible [120].

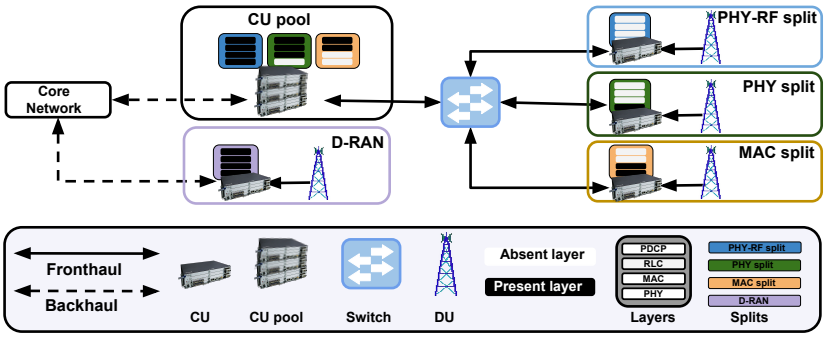


Figure 5.1 – This figure shows the coexistence of different functional splits at the CU pool. Notice that, apart from the CU pool, also the DUs possess processing capabilities, regardless of the functional split option being employed at the considered time.

5.2.1 Substrate Network Model

Let $G_s = (N_s, E_s)$ be an *undirected* graph modeling the InP's physical network, where $N_s = N_s^{du} \cup N_s^{cu}$ is the set of $n_1 = |N_s^{du}|$ DUs and $n_2 = |N_s^{cu}|$ CU pools, and $E_s \subseteq N_s^{du} \times N_s^{cu}$ is the set of fronthaul links. An edge $e^{nm} \in E_s$ if and only if a connection exists between $n, m \in N_s$. Three weights, $\omega_s^{ant}(n)$, $\omega_s^{prb}(n)$ and $\omega_s^{prc}(n)$, are assigned to each node $n \in N_s : \omega_s^{ant,prb,prc}(n) \in \mathbb{N}^+$ representing, respectively, the number of RF front–ends, the set of PRBs whose cardinality depends on the employed carrier[2], and the processing capacity supported by the node. Each substrate node is also associated with a geographic location $loc(n)$, as x, y coordinates, and a coverage radius $\delta(n)$, in meters, indicating the coverage area of the small cell centered on the node $n \in N_s$. Another weight $\omega_s^{bwt}(e^{nm})$ is assigned to each link $e^{nm} \in E_s : \omega_s^{bwt}(e^{nm}) \in \mathbb{N}^+$ representing the capacity (in Gbps) of the link connecting the two nodes. Table 5.1 summarizes the substrate network parameters.

Notice how, it is our assumption that the CU pool is equipped with enough computational capacity to support all the DUs employing the lowest possible functional split i.e., the PHY–RF split, which requires all baseband signal processing to take place at the CU pool. Whereas, the DUs are equipped with enough computational capacity to process the signals with the highest possible functional split, i.e. the MAC split. We also assume that fronthaul links have enough capacity to support the PHY–RF split and that only DUs are equipped with RF front–ends [8].

5.2.2 Virtual Network Model

There are different approaches to model VNRs, from resource–based [136, 137] models to throughput–based models [138]. In this work, we use a resource–based model in which MVNOs can request one or more small cells with a particular antenna configuration and a fixed amount of PRBs to be allocated to their small cells. This model provides no throughput guarantees to the MVNO's users whose performance can be affected by their distribution and by the time–varying nature of their wireless channel.

[2]Carrier is an RF bandwidth (in MHz) that is allocated to a small cell. It is assumed that the same carrier is used at all small cells. This corresponds to the worst case scenario.

Table 5.1 – Substrate network parameters

Variable	Description
G_s	Substrate network graph.
N_s	Substrate nodes in G_s.
N_s^{du}	Substrate DUs in G_s.
N_s^{cu}	Substrate CU pools in G_s.
E_s	Substrate links in G_s.
$\omega_s^{ant}(n)$	Number of RF front–ends available at the node $n \in N_s$.
$\omega_s^{prb}(n)$	Set of PRBs available at the node $n \in N_s$.
$\omega_s^{prc}(n)$	The processing capacities of the node $n \in N_s$.
$\omega_s^{bwt}(e^{nm})$	Capacity of the link $e^{nm} \in E_s$ (in Gbps).
$loc(n)$	Geographical location of the node $n \in N_s$.
$\delta(n)$	Coverage radius of the node $n \in N_s$ (in meters).
$R_n^{n'}(m)$	Fronthaul bandwidth needed on the link of node $n \in N_s$ that uses the m^{th} split to host virtual node $n' \in N_v$ (in Gbps).
Λ_{bwt}	Cost for each Gbps of bandwidth resource.
Λ_{itf}	Cost for each interfering PRB.

Virtual network requests are modeled as *undirected* graphs $G_v = (N_v, E_v)$ where $N_v = N_v^{du} \cup N_v^{cu}$ is the set of virtual DUs $n_1 = |N_v^{du}|$ and virtual CU pools $n_2 = |N_v^{cu}|$, and $E_v \subseteq N_v^{du} \times N_v^{cu}$ is the set of virtual fronthaul links. Each node $n \in N_v$ in the VNRs has two weights, $\omega_s^{ant}(n)$ and $\omega_s^{prb}(n)$ indicating, respectively, the number of RF front–ends and the number of PRBs, while each DU $n \in N_v^{du}$ is also associated with a geographic location $loc(n)$, as x, y coordinates. This information together with the substrate DU location and its coverage radius is used to express how far a virtual DU $n \in N_v^{du}$ can be placed from the preferred location specified by $loc(n)$. Table 5.2 summarizes the VNR parameters.

Notice that MVNOs do not request processing resource at the CU pools or DUs. Neither do they request fronthaul bandwidth. Given the chosen functional split of a virtual small cell, which is the one of the host substrate small cell, and considering its requirements ($\omega_v^{ant}(n)$, $\omega_v^{prb}(n)$), the fronthaul bandwidth required to support the given virtual small cell is calculated by the InP [8]. For example, if the virtual DU $n \in N_v^{du}$ requests $\omega_v^{ant}(n) = 2$ (i.e., 2×2 MIMO configuration) RF front–ends and $\omega_v^{prb}(n) = 50$ PRBs then this translates to the fronthaul bandwidth of $\omega_v^{bwt} = 2.46$ Gpbs, $\omega_v^{bwt} = 0.54$ Gpbs and $\omega_v^{bwt} = 0.046$ Gbps, respectively, in the cases of the PHY–RF split, the PHY split[3] and the MAC split, assuming the transport block size of 21384 bits in an LTE network with 20 MHz bandwidth. Notice that as opposed to the PHY split and the MAC split, the fronthaul bandwidth for the PHY–RF split does not depend on the requested number of PRBs.

[3]Note that the PHY split considered in this chapter corresponds to the PHY II split in Figure 3.1.

Table 5.2 – Virtual network request parameters

Variable	Description
G_v	Virtual network request graph.
N_v	Virtual nodes in G_v.
N_v^{du}	Virtual DUs in G_v.
N_v^{cu}	Virtual CU pools in G_v.
E_v	Virtual links in G_v.
$\omega_v^{ant}(n)$	Requested number of RF front–ends at the node $n \in N_v$.
$\omega_v^{prb}(n)$	Requested number of PRBs at the node $n \in N_v$.
$\omega_v^{prc}(n)$	Processing capacity required at the node $n \in N_v$.
$\omega_v^{bwt}(e^{nm})$	Capacity required for the link $e^{nm} \in E_v$ (in Gbps).
$loc(n)$	Desired geographical location for the DU $n \in N_v^{du}$.

5.3 CU Placement

In this section, the CU placement problem is introduced followed by the dynamic and static ILP–based algorithms (*ILP–DM* and *ILP–ST*), and by the scalable dynamic and static embedding heuristics (*HEU–DM* and *HEU–ST*).

5.3.1 Problem Statement

The CU placement problem studied in this chapter can be formally stated as follows:

Given: a star–shaped substrate topology, similar to the one depicted in Fig. 5.5a, where each node has its geographical location, processing capacity, fronthaul link capacity, RF front–ends and bandwidth expressed in terms a number of PRBs.

Find: the resource allocations for each virtual node/link in VNRs (e.g., PRB, RF front–end, fronthual bandwidth) and the functional split option employed at each substrate/virtual node.

Objective: minimize the level of ICI and select the optimal functional split option for each substrate/virtual node in such a way as to also minimize the fronthaul bandwidth required to embed the VNRs and to suppress the ICI level by using techniques/algorithms enabled by the selected functional split.

Upon arrival of a new VNR, the substrate network must decide whether it is to be accepted or rejected. The embedding process consists of two steps: node embedding and link embedding. In the node embedding step, each virtual node in the request is mapped to a substrate node. In the link embedding step, each virtual link is mapped to a single substrate path. In both cases, some constraints must be satisfied.

5.3.2 Dynamic ILP–based Placement Algorithm (*ILP–DM*)

In this subsection, the *ILP–DM* algorithm is presented. *ILP–DM* employs a dynamic embedding strategy, meaning that with the arrival of a new VNR, the request along with the ones that have been previously embedded are re–embedded. Thus, with every embedding, the optimal embedding solution is found for all the requests.

Before presenting the ILP formulation, let us define a cluster of candidate substrate DUs for each virtual DU. Every virtual DU $n' \in N_v^{du}$ in a request has a desired location $loc(n')$. Whereas, every substrate DU $n \in N_s^{du}$ has both a location $loc(n)$ and a coverage radius $\delta(n)$. For each virtual DU $n' \in N_v^{du}$, the cluster of candidate DUs $\Omega(n')$ can then be defined:

$$\Omega(n') = \Big\{ n \in N_s^{du} | dis(loc(n), loc(n')) \leq \delta(n) \Big\} \tag{5.1}$$

The objective of this formulation is to minimize the ICI at each small cell and, at the same time, minimize the fronthaul bandwidth required to serve the VNRs. The chosen objective function is:

$$\begin{aligned} minimize\Big(\sum_{n' \in N_v^{du}} \sum_{n \in \Omega(n')} \sum_{n^\star \in \Omega(n)} \sum_{p \in \omega_s^{prb}(n), p^\star \in \omega_s^{prb}(n^\star)}^{p=p^\star} \Lambda_{itf} \Phi_p^n \Phi_{p^\star}^{n^\star} \\ + \sum_{n \in N_s^{du}} \sum_{n' \in N_v^{du}} \sum_{m \in \{1,2,\ldots,|R_n^{n'}|\}} \Lambda_{bwt} R_n^{n'}(m) \Phi_{n,m}^{n'} \Big) \end{aligned} \tag{5.2}$$

where (with a slight abuse of notation) we use $n^\star \in \Omega(n)$ to indicate that the node $n^\star \in N_s^1 (n^\star \neq n)$ has overlapping radio coverage with the candidate substrate node $n \in \Omega(n')$ (i.e. an interfering node). Table 5.3 summarizes all binary decision variable used in the ILP problem formulations.

The first term in the objective function aims at minimizing the number of overlapping PRBs at each host small cell, while the second term minimizes the fronthaul bandwidth demand by taking into account the first term and selecting the most optimal functional split for host small cells. The optimal split selection depends on the ICI level since different splits can enable different interference management techniques to be employed aiming to reduce/cancel the ICI [9]. This defines a trade–off between the fronthaul bandwidth demand and the level of acceptable interference in the system. It is important to mention that we select the cost of each Gbps of bandwidth resource large enough from the cost of each interfering PRB ($\Lambda_{itf} \ll \Lambda_{bwt}$) in order to make sure that second term in the objective function is significantly larger than the first one. Nevertheless, the first term, although negligible, has to be present in the objective function since the goal of the InP is (i) to avoid allocating overlapping PRBs to the virtual small cells that have been hosted by neighbor substrate small cells, and (ii) once the overlapping PRB allocation becomes inevitable, considering the level of ICI, to select the optimal functional split option for each host small cell with the goal of minimizing the fronthaul bandwidth demand and, thanks to the selected functional split, be able to employ ICI management techniques aiming to suppress the ICI.

In order to minimize the ICI, we first need to quantify it. For each small cell, a collision domain is selected, containing all the small cells with which the considered small cell has an overlap in its coverage area. In essence, these would be the small cells whose transmitted signals in the uplink/downlink direction may interfere with the ones of the considered small cells. The ICI[4] can then be estimated based on the information on the PRB chunks allocated to VNRs. Note that although more accurate interference estimation is possible by modeling users per MVNO and taking into account their channel quality information along with the PRB allocation, it is out of the scope of this work.

Note that the objective function contains a quadratic term $\Phi_p^n \Phi_{p^\star}^{n^\star}$ that results in a standard (non–convex) quadratic formulation. To linearize this term, we define a variable $\Phi_{p,p^\star}^{n,n^\star}$ and substitute it to the quadratic term in the objective function:

$$\Phi_p^n \Phi_{p^\star}^{n^\star} \approx \Phi_{p,p^\star}^{n,n^\star} = \begin{cases} 1 & if \quad \Phi_p^n = \Phi_{p^\star}^{n^\star} = 1 \\ 0 & otherwise \end{cases} \tag{5.3}$$

We will now detail the constraints used in the ILP formulation. The following constraint ensures that if the same PRB is being used by two or more small cells that are in the same collision domain then a penalty is applied to each of the small cell that uses the PRB:

$$\Phi_p^n + \Phi_{p^\star}^{n^\star} - \Phi_{p,p^\star}^{n,n^\star} \leq 1 \quad \forall n^\star \in \Omega(n),\ \forall n \in \Omega(n'),\ \forall n' \in N_v^{du}, \forall p = p^\star\ p \in \omega_s^{prb}(n), p^\star \in \omega_s^{prb}(n^\star) \tag{5.4}$$

In essence, $\Phi_{p,p^\star}^{n,n^\star} = 1$ indicates the presence of ICI at $n^\star \in \Omega(n)$ and $n \in \Omega(n')$ substrate DUs. The penalty (i.e., ICI) increases with an increase in the number of overlapping PRBs used by the small cells belonging to the same collision domain. The following constraints deal with, respectively, the required number of RF front–ends and PRB resources, making sure that they are at most equal to the resources available at the substrate nodes:

$$\sum_{n' \in N_v^{du}} \omega_v^{ant}(n') \Phi_n^{n'} \leq \omega_s^{ant}(n) \quad \forall n \in N_s^{du} \tag{5.5}$$

$$\sum_{n' \in N_v^{du}} \omega_v^{prb}(n') \Phi_n^{n'} \leq |\omega_s^{prb}(n)| \quad \forall n \in N_s^{du} \tag{5.6}$$

It is worthwhile to note that the processing resources at the virtual DUs/CU pools as well as the fronthaul bandwidth resources are not requested by MVNOs. Those resources depend on the actual functional split of the host substrate small cells and are computed by the InPs after having the optimal mapping solution of the VNRs.

[4] In this study, the interference estimation is based on the worst case assumption; that is, the PRBs allocated to MVNOs' VNRs are always in use.

Table 5.3 – Binary decision variables $\{0,1\}$

Variable	Description
$\Phi_{p}^{n}, \Phi_{p^\star}^{n^\star}$	Show if the PRBs $p, p^\star$ are in use at the substrate nodes n and $n^\star$, respectively.
$\Phi_{p,p^\star}^{n,n^\star}$	Shows the presence of ICI at $n^\star \in \Omega(n)$ and at $n \in \Omega(n')$ substrate DUs.
$\Phi_{n}^{n'}$	Shows if the virtual DU $n' \in N_v^{du}$ has been mapped to the substrate node $n \in \Omega(n')$.
$\Phi_{n,m}^{n'}$	Shows if the virtual DU $n' \in N_v^{du}$ has been mapped to the m^{th} functional split option of the substrate DU $n \in \Omega(n')$.
$\Phi_{n,p}^{n'}$	Shows if the PRB $p \in \omega_s^{prb}(n)$ of the substrate DU $n \in N_s^{du}$ has been allocated to the virtual DU $n' \in N_v^{du}$.

Each requested virtual node $n' \in N_v$ must be mapped only once (5.7), while each virtual DU must be mapped only on a substrate DU that belongs to its cluster of candidates (5.8):

$$\sum_{n \in N_s} \Phi_{n}^{n'} = 1 \quad \forall n' \in N_v \tag{5.7}$$

$$\sum_{n \in N_s^{du} \setminus \Omega(n')} \Phi_{n}^{n'} = 0 \quad \forall n' \in N_v^{du} \tag{5.8}$$

The next constraint prevents multi–allocation of the same PRB, making sure that each PRB is assigned to one VNR at most:

$$\sum_{n' \in N_v^{du}} \Phi_{n,p}^{n'} \leq 1 \quad \forall n \in N_s^{du}, \quad \forall p \in \omega_s^{prb}(n) \tag{5.9}$$

Virtual DU embedding and PRB allocation must be consistent, meaning that if a virtual DU has been mapped to a given substrate DU then only the PRBs of that substrate DU must be allocated to the virtual DU:

$$\sum_{p \in \omega_s^{prb}(n)} \Phi_{n,p}^{n'} - \omega_v^{prb}(n') \Phi_{n}^{n'} = 0 \quad \forall n' \in N_v^{du}, \quad \forall n \in \Omega(n') \tag{5.10}$$

In order to compute the fronthaul bandwidth requirement for the virtual DU $n' \in N_v^{du}$, it has to be mapped to the functional split of the host substrate DU $n \in \Omega(n')$:

$$\Phi_{n}^{n'} - \sum_{m \in \{1,2,\ldots,|R_n^{n'}|\}} \Phi_{n,m}^{n'} = 0 \quad \forall n' \in N_v^{du}, \quad \forall n \in \Omega(n') \tag{5.11}$$

Finally, the last constraint handles the functional split selection for each substrate small cell ensuring that all the virtual small cells that have been mapped to the same substrate small cells have selected the same single functional split of the substrate small cell:

$$\sum_{n^\star \in \Omega(n)} \sum_{p \in \omega_s^{prb}(n), p^\star \in \omega_s^{prb}(n^\star)}^{p=p^\star} \Phi_{p,p^\star}^{n,n^\star} \leq I(m) \sum_{n^\star \in \Omega(n)} |\omega_s^{prb}(n^\star)| \quad \forall n' \in N_v^{du}, \quad \forall n \in \Omega(n'), \quad \forall m \in \{1,2,...,|R_n^{n'}|\} \tag{5.12}$$

where $I(m)$ represents the acceptable ICI level for each m^{th} functional split option (see Table 5.4). This constraint effectively puts an upper bound on the number of acceptable overlapping PRB allocations. For example, for a PHY–RF split we are willing to accept as many PRB allocation overlaps as the number of PRBs in a collision domain. This essentially results in a reuse factor of 1, which is acceptable since the PHY–RF split enables several advanced interference mitigation techniques to be employed. Conversely, as the functional split moves up in the protocol stack we reduce the maximum number of allowed PRB overlaps.

5.3.3 Static ILP–based Placement Algorithm (*ILP–ST*)

The *ILP–ST* algorithm resembles the *ILP–DM* algorithm in terms of the fact that both have the same objective function and that all the constraints defined for the *ILP–DM* algorithm are held also for the *ILP–ST* algorithm. However, the difference between them is that in the case of the *ILP–ST* algorithm, as opposed to the *ILP–DM* one, VNRs are embedded sequentially. In other words, with the arrival of a new VNR, only that request is embedded. Thus, as the name of the algorithm implies, a static embedding is considered. As we will see in Sec. 5.4, this algorithm can be used to solve larger embedding problems since the static embedding is significantly faster compared to its dynamic counterpart. It is important to mention that in dynamic placement scenarios, G_v represents the composition of all the virtual network graphs, including the one of the new VNR, that are to be mapped with the arrival of a new VNR. Whereas in the static embedding scenarios, G_v represents the virtual network graph that characterizes only the new request.

5.3.4 Scalable Dynamic Placement Heuristic (*HEU–DM*)

The *ILP–DM* CU placement algorithm becomes computationally intractable as the size of the substrate network and/or of the VNRs increases. For example, the *ILP–DM* algorithm takes one week on Intel Core i7 laptop (3.0 GHz CPU, 16 Gb RAM) using the Matlab ILP solver (intlinprog) to map a request, having 20 virtual nodes, to a substrate network with 20 nodes. In order to address this scalability issue, we also propose the *HEU–DM* heuristic, which is able to embed the same VNR in a limited amount of time. Like in the case of the *ILP–DM* algorithm, in this case also a dynamic embedding is considered. It is worth noticing that, in order to ensure the correctness of the solutions, we pass all the solutions found by the heuristic through the same constraints defined for the ILP formulation in Subsection 5.3.2.

Algorithm 2 *HEU–DM*

procedure *Input:*(G_s, G_v)
  Step 1: Find the list of candidates and their neighbors.
  for $n' \in N_v^{du}$ **do**
    for $n \in N_s^{du}$ **do**
      $d \leftarrow dis(loc(n'), loc(n))$
      if $d \leq \delta(n)$ **then**
        $candidates(n') \leftarrow n$
      end if
    end for
    for $cand \in candidates(n')$ **do**
      for $n \in N_s^{du}$ **do**
        $d \leftarrow dis(loc(cand), loc(n))$
        if $n \neq cand$ and $d \leq 2\delta(n)$ **then**
          $neighbor(n')(cand) \leftarrow n$
        end if
      end for
    end for
  end for
  Step 2: Find all possible combinations of candidates.
  $combs\_cands \leftarrow \emptyset$
  for $n' \in N_v^{du}$ **do**
    $combs\_cands \leftarrow combvec(combs\_cands, candidates(n'))$
  end for
  Step 3: Find the best combinations of candidates (minimum interference).
  $min\_intf \leftarrow \infty$
  for $comb \in combs\_cands$ **do**
    $substrate\_resources\_copy \leftarrow substrate\_resources$
    $intf\_all \leftarrow 0$
    for $n' \in N_v^{du}$ **do**
      $cand \leftarrow comb(n')$
      $intf(cand) \leftarrow 0$
      if $\omega_s^{ant}(cand) < \omega_v^{ant}(n')$ or $|\omega_s^{prb}(cand)| < \omega_v^{prb}(n')$ **then**
        $intf\_all \leftarrow \infty$
        break
      end if
      for $neigh \in neighbor(n')(cand)$ **do**
        for $prb\_idx \in \omega_s^{prb}$ **do**
          if $neigh(prb\_idx) = cand(prb\_idx) = 1$ **then**
            $intf(cand) = intf(cand) + 1$
          end if
        end for
      end for
      $intf\_all = intf\_all + intf(cand)$
      Update $substrate\_resources\_copy$
    end for
    if $intf\_all \leq min\_intf$ **then**
      $min\_intf \leftarrow intf\_all$
      $best\_cands \leftarrow comb$
    end if
  end for

Step 4: Allocate resources & select functional splits.
for $n' \in N_v^{du}$ **do**
  $mapped(n') \leftarrow best\_cands(n')$
  $sorted(best\_cands(n')) \leftarrow sortprb(intf(best\_cands(n'))\uparrow)$
  $alloc\_prb(n') \leftarrow sorted(best\_cands(n'))[1:\omega_v^{prb}(n')]$
  for $s \in \{1,2,...,|R_n^{n'}|\}$ **do**
    if $intf(best\_cands(n')) \in intf\_bounds(s)$ **then**
      $split \leftarrow s$
      break
    end if
  end for
  $fh\_band(n') \leftarrow compute\_band(\omega_v^{prb}(n'), \omega_v^{ant}(n'), split)$
end for
end procedure

The basic design intuition behind *HEU–DM* is as follows. Initially, for each virtual DU in the VNR, it computes a list of candidate substrate DUs and a list of neighbor DUs for each candidate DU. Then, *HEU–DM* generates a matrix that contains all possible combinations of candidate DUs for each virtual DU in the request (i.e., one candidate DU per virtual DU in a single combination). This is followed by considering all possible combinations and selecting the one that would introduce the lowest level of ICI in the network. Lastly, each virtual DU is mapped to its corresponding substrate DU in the selected candidate combination, the requested number of PRBs is assigned, the functional split is selected for the host DU and the fronthaul bandwidth is allocated to the virtual DU, considering its PRB requirement, RF front–end requirement and the functional split option of the host DU.

Before describing the details of how *HEU–DM* works, let us make the following notations. Let $m_1 = |N_v^{du}|$ and $m_2 = |N_s^{du}|$ be the number of, respectively, virtual and substrate DUs, while $n_1 = |\Omega(n')|$ and $n_2 = |\Omega(n)|$ be the maximum number of, respectively, candidate substrate DUs per virtual DU $n' \in N_v^{du}$ and neighbor DUs per candidate DU $n \in \Omega(n')$ of the virtual DU n' . Finally, let $s = |R_n^{n'}|$ be the number of functional split options, and $p = |\omega_s^{prb}|$ be the number of PRBs at each substrate small cell.

HEU–DM is composed of four steps. In the first step, a cluster of candidate DUs $candidates(n')$ is selected for each virtual DU $n' \in N_v^{du}$ by considering its desired location. Then, for each candidate substrate DU $cand \in candidates(n')$ of each virtual DU $n' \in N_v^{du}$, a neighbor list is created $neighbor(n')(cand)$. This list contains all the substrate DUs that have an overlapping coverage area with the candidate DU $cand \in candidates(n')$. This process takes $O(m_1 m_2(n_1+1))$ time. In the second step, by using $combvec()$ function in Matlab, a combination matrix $combs\_cands$ is created, covering the entire search space of the candidates' combinations for all the virtual DUs. Each column of this matrix represents one combination of candidate substrate DUs i.e. one candidate DU per virtual DU. This process requires $O(m_1)$ time.

The third step aims at finding the best combination of candidates (i.e., the best candidate column vector); that is, the one that after mapping the virtual DUs upon would introduce the lowest level of ICI in the network. Specifically, each candidate vector $comb$ is considered, and for each candidate substrate DU $cand$ of each virtual DU $n' \in N_v^{du}$ in the candidate vector $comb$, RF front–end and PRB resource availability is checked in order to find out whether the considered substrate DU is capable of supporting the resource requirements of the virtual DU. The total ICI is then estimated by accumulating the ICI at each candidate substrate DU in the $comb$ vector, which is computed by checking the usage of each pair of PRB (i.e., on the candidate $cand$ and its each neighbor $neigh$) in the PRB set ω_s^{prb}. At the end of this step, the combination vector $best\_cands$ is picked that would introduce the lowest level of network–wide ICI. Step 3 requires $O(m_1^2 n_1 n_2 p)$ time.

In the last step, each virtual DU $n' \in N_v^{du}$ is mapped to its corresponding candidate substrate DU in the combination vector $best\_cands(n')$. The PRBs of the host DUs are sorted in the ascending order of likelihood in terms of interference, and then the requested number of PRBs $\omega_v^{prb}(n')$ is allocated starting from the one that would introduce the lowest level of ICI in its collision domain. After the PRBs have been allocated, the overall ICI level at the host DU is estimated, and the appropriate functional spit is selected. This is followed by computing the fronthaul bandwidth required to host the virtual DU by using the $compute\_band()$ function, providing inputs the requested number of PRBs, the RF front–ends and the split option of the host substrate DU. The last step takes $O(m_1 s)$ time. Thus, the overall time complexity of the *HEU–DM* heuristic is $O(m_1[m_2(n_1+1) + m_1 n_1 n_2 p + s + 1])$.

5.3.5 Scalable Static Placement Heuristic (*HEU–ST*)

The scalability might become a problem also for the *HEU–DM* heuristic when big–sized substrate/virtual networks with a few hundreds of substrate/virtual nodes are considered. In order to address this problem, we also propose a real–time *HEU–ST* heuristic. As the name of the heuristic implies, *HEU–ST* embeds statically the VNRs. In other words, with the arrival of a new VNR, only that request is embedded.

The basic design intuition behind *HEU–ST* is as follows. Initially, *HEU–ST* considers the first virtual DU in the VNR, creating a list of its candidate DUs. It then loops over candidate DUs and for each creates a list of neighbor DUs, which is then considered in order to find the candidate DU that after mapping the virtual DU upon would introduce the lowest level of ICI in its collision domain. Lastly, the considered virtual DU is mapped to the selected candidate substrate DU, the requested number of PRBs is assigned, the functional split is selected for the host DU and the fronthaul bandwidth is allocated to the virtual DU, considering its PRB requirement, RF front–end requirement, and the functional split option of the host substrate DU. The substrate resources are then updated and the described process is repeated for the rest of the virtual DUs in the request.

Algorithm 3 *HEU–ST*

procedure *Input:*(G_s, G_v)
  Step 1: Compute a list of candidates.
  for $n' \in N_v^{du}$ **do**
    $candidates(n') \leftarrow \emptyset$
    for $n \in N_s^{du}$ **do**
      $d \leftarrow dis(loc(n'), loc(n))$
      if $d \leq \delta(n)$ **then**
        if $\omega_v^{ant}(n') \leq \omega_s^{ant}(n)$ and $\omega_v^{prb}(n') \leq |\omega_s^{prb}(n)|$ **then**
          $candidates(n') \leftarrow n$
        end if
      end if
    end for
    Step 2: Select the small cell at which the ICI is minimum.
    $min\_intf \leftarrow \infty$
    for $cand \in candidates(n')$ **do**
      for $n \in N_s^{du}$ **do**
        $d \leftarrow dis(loc(cand), loc(n))$
        if $n \neq cand$ and $d \leq 2\delta(n)$ **then**
          $neighbor(cand) \leftarrow n$
        end if
      end for
      $intf(cand) \leftarrow 0$
      for $neigh \in neighbor(cand)$ **do**
        for $prb\_idx \in \omega_s^{prb}$ **do**
          if $neigh(prb\_idx) = cand(prb\_idx) = 1$ **then**
            $intf(cand) = intf(cand) + 1$
          end if
        end for
      end for
      if $intf(cand) \leq min\_intf$ **then**
        $min\_intf = intf(cand)$
        $best\_cand \leftarrow cand$
      end if
    end for
    Step 3: Allocate resources & select functional splits.
    $mapped(n') \leftarrow best\_cand$
    $sorted(best\_cand) \leftarrow sortprb(intf(best\_cand)\uparrow)$
    $alloc\_prb(n') \leftarrow sorted(best\_cand)[1 : \omega_v^{prb}(n')]$
    for $s \in \{1, 2, ..., |R_n^{n'}|\}$ **do**
      if $intf(best\_cand) \in intf\_bounds(s)$ **then**
        $split \leftarrow s$
        break
      end if
    end for
    $fh\_band(n') \leftarrow compute\_band(\omega_v^{prb}(n'), \omega_v^{ant}(n'), split)$
    Update substrate resources
  end for
end procedure

We will now describe the details of *HEU–ST*, using the same notations for both heuristics in order to estimate the embedding time complexity. *HEU–ST* is composed of three steps. In the first step, a cluster of candidate DUs is selected for each $n' \in N_v^{du}$ virtual DU, considering its requirements in terms of the antenna configuration, the number of PRBs and the desired location. This step takes $O(m_1 m_2)$ time. In the second step, a neighbor list $neighbor(cand)$ is created for each candidate DU of each virtual DU $n' \in N_v^{du}$. The relative distance between the potential interfering DUs is considered for populating the neighbor list. The heuristic then measures the interference coming from each DU in the neighbor list. At the end of this step, the best candidate DU $best\_cand$ is picked; that is, the one that would introduce the lowest level of network–wide ICI. The second step takes $O(m_1 n_1 (m_2 + n_2 p))$ time.

In the last step, virtual DU $n' \in N_v^{du}$ is mapped to the best substrate DU $best\_cand$. The PRBs of the host substrate DU are sorted in the ascending order of likelihood in terms of interference, and then the requested number of PRBs $\omega_v^{prb}(n')$ is allocated starting from the one that would introduce the lowest level of ICI in its collision domain. After the PRBs have been allocated, the overall ICI level at the host DU is estimated, and the appropriate functional split is selected. Lastly, the fronthaul bandwidth, required to host the virtual DU, is computed by using the $compute\_band()$ function, providing inputs the requested number of PRBs, the RF front–ends and the split option of the host substrate small cell. This is followed by updating the substrate resources and repeating the steps for all the virtual DUs in the VNR. This step takes $O(m_1 s)$ time. Thus, the overall time complexity of *HEU–ST* is $O(m_1[n_1(m_2 + n_2 p) + m_2 + s])$.

5.4 Evaluation

The goal of this section is to compare the ILP–based algorithms with the heuristics. We shall first describe the simulation environment. We will then report on the outcomes of the numerical simulations conducted in a discrete event simulator implemented in Matlab.

5.4.1 Simulation Environment

The reference substrate network has a star–shaped topology with 8 DUs directly connected to a single CU pool via optical fronthaul links (10 Gbps), providing mobile coverage in an area of 2 km^2. This is a very conservative assumption, and in a more realistic scenario, a ring or a tree topology may be used to connect DUs with CU pools. However, the focus of this study is on the flexible functional split concept, rather than on the fronthaul topology. A discussion on different fronthaul topologies is presented in Sec. 5.5.1. The inter–DU distance is 800 meters, and it is assumed that each DU possesses 6 omni–directional[5] antennas, providing radio coverage with the radius of 500 meters. This means that in some zones there will be 200 meters of area covered by more than one DU, which, in turn, means that if the users are located in that area and are connected to different DUs being scheduled at the same PRBs,

[5]Notice that also sectoral antennas at the small cells can be easily considered. In this work we, however, we consider only omni–directional antennas since sectoral antennas, although more prevalent, would complicate the model without bringing any significant benefit.

Table 5.4 – Relative processing capacities of the DU and the CU of the small cell n and the acceptable ICI level for the considered functional splits.

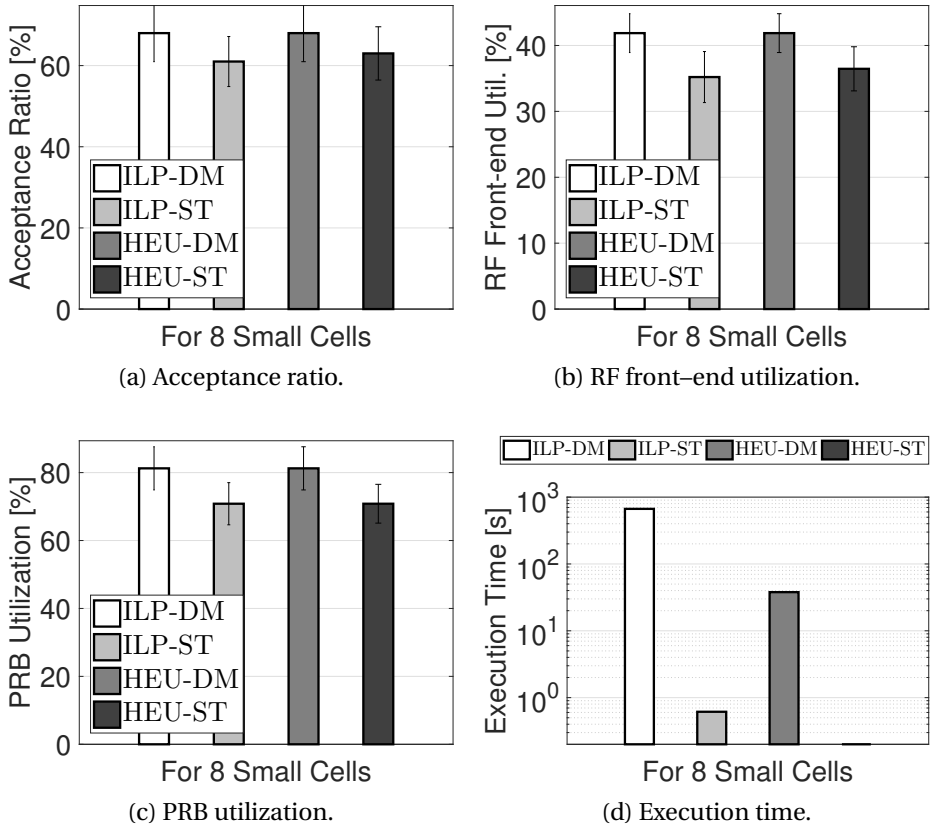
Functional Split	**ICI level** (I)	**Processing Capacity**	
		DU	**CU**
PHY–RF split	1	$0 \cdot \omega_s^{prc}(n)$	$1 \cdot \omega_s^{prc}(n)$
PHY split	0.6	$0.5 \cdot \omega_s^{prc}(n)$	$0.5 \cdot \omega_s^{prc}(n)$
MAC split	0.3	$0.7 \cdot \omega_s^{prc}(n)$	$0.3 \cdot \omega_s^{prc}(n)$

they will then create interference on one another, irrespective of the MVNO that users belong to. Lastly, it is assumed that each DU employs the same LTE 20 MHz bandwidth (100 PRBs). One of the most prominent advantages of the PHY–RF split is that through better inter–cell coordination it enables complex interference cancellation/avoidance algorithms such as JT to be employed. In some cases, however, when there is no ICI in the network, or when it is very low, employing the PHY–RF split in the C–RAN architecture would result in a waste of resources (e.g., fronthaul bandwidth) without bringing any additional benefit. For instance, when DUs are well–separated, meaning that they have no overlapping coverage area, or when DUs do have an overlapping area, but the users are scheduled at different PRBs. Depending upon the level of ICI, different functional splits would be appropriate to be employed. In our model, there are three categories of interference and three corresponding functional splits (see Table 5.4). Notice that as opposed to the PHY–RF split, in the case of the PHY/MAC split, some part of the baseband signal processing is taking place at DUs. For example, in the case of the PHY split, it is assumed that the half processing capacity is allocated to the DUs and the other half to the CU pool. This is because the most processor–hungry procedure (i.e., FFT/IFFT) is performed in the PHY layer. The processing requirement increases at the DUs and decreases at the CU pools when fewer layers (e.g., PHY layer, MAC layer) are centralized at the CU pools.

In this study, we assume that a fixed number of VNRs are embedded sequentially. The reported results are the average of 10 simulations each with 10 embeddings with 95% confidence intervals. During each embedding process, the number of virtual DUs, RF front–ends and PRBs are randomly selected for each request in the set of, respectively, $\{1,2\}, \{1,2\}$ and $\{30,60\}$.

5.4.2 Simulation Results

Figure 5.2 shows the acceptance ratio, the RF front–end utilization, the PRB utilization and the execution time of all the exact algorithms and the heuristics. As expected, both *ILP–DM* and *HEU–DM* dynamic embedding algorithms achieve better performance in terms of acceptance ratio, RF front–end and PRB utilization compared to their static counterparts (see Fig. 5.2a, 5.2b and 5.2c). We can see that both dynamic placement algorithms have accepted equal number of VNRs. It can be observed that also the RF front–end utilization (see Fig. 5.2b) and the PRB utilization (see Fig. 5.2c) is equal for those algorithms. These equalities advocate that the *HEU–DM* heuristic achieves near optimal solutions.



(a) Acceptance ratio.

(b) RF front–end utilization.

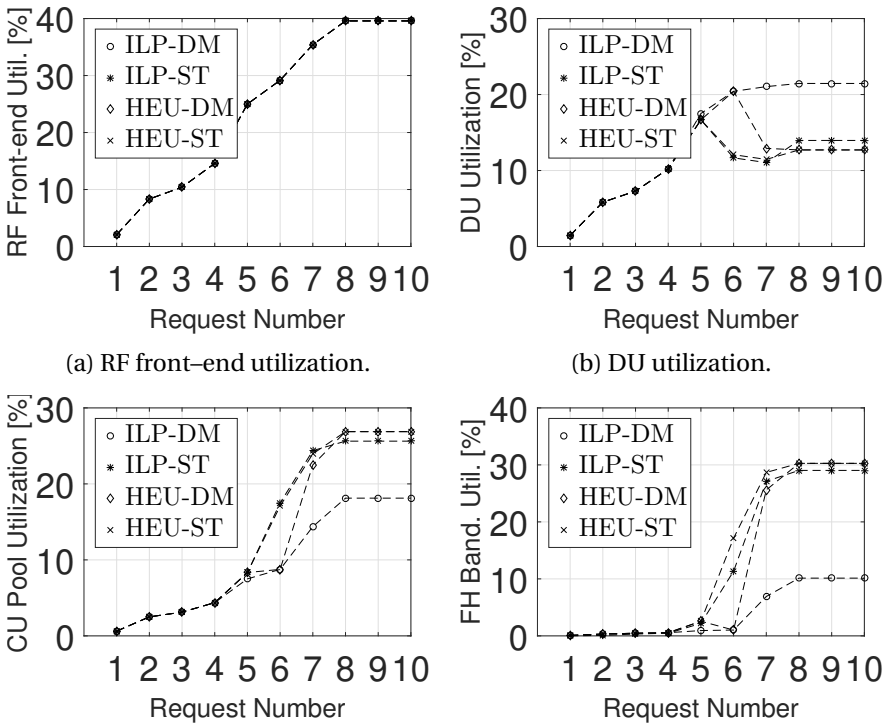
(c) PRB utilization.

(d) Execution time.

Figure 5.2 – Acceptance ratio, RF front–end utilization, PRB utilization and execution time for the ILP–based algorithms and the heuristics.

With regard to the static embedding algorithms, Fig. 5.2a shows that the *HEU–ST* heuristic has accepted slightly more VNRs than the *ILP–ST* algorithm. This negligible difference can be seen also in terms of the RF front–end utilization. Whereas, it can be observed that their PRB utilization is equal. This again witnesses about the *HEU–ST* heuristic achieving a good approximation compared to its *ILP–ST* counterpart. However, as we will see in Fig. 5.3, this does not mean that the *HEU–ST* heuristic finds more optimal solutions in comparison with the *ILP–ST* algorithm. The rationale behind *HEU–ST* performing slightly better in terms of the acceptance ratio and the RF front–end utilization is that, since for those algorithms a static embedding is taking place, it cannot be claimed that the performance of the *ILP–ST* algorithm after all embeddings must *always* be better compared to the *HEU–ST* heuristic since they have different VNR embedding logic. Moreover, by performing a static embedding, with the arrival of a new VNR, the previously embedded request cannot be re–embedded. Thus, the performance of the static embedding algorithm/heuristic merely depends on their previous mappings and on the requirement of the new requests in terms of the desired location, number of PRBs and RF front–ends.

The *ILP–DM* and *ILP–ST* exact algorithms become computationally intractable when networks with, respectively, a few tens and a few hundreds of substrate/virtual nodes are considered.



(a) RF front–end utilization.

(b) DU utilization.

(c) CU pool utilization.

(d) Fronthaul bandwidth utilization.

Figure 5.3 – RF front–end, DU processing resource, CU pool processing resource and fronthaul (FH) bandwidth utilizations for the ILP–based algorithms and the heuristics.

Depending on the size of the substrate and virtual networks (e.g., ultra–dense networks with hundreds of small cells), also the *HEU–DM* heuristic might encounter a scalability problem in terms of the required time to map VNRs. With the sole purpose of tackling this problem, a static embedding heuristic (*HEU–ST*) has also been proposed. Figure 6.4d displays the total time taken to embed 10 VNRs (a single iteration). Note that during this iteration all the algorithms/heuristics have embedded an equal number of VNRs (see Fig. 5.3). It can be observed that among the dynamic embedding algorithm/heuristic, *HEU–DM* has taken 20 times less time to embed those requests compared to its exact counterpart. Whereas among the static embedding algorithm/heuristic, *HEU–ST* has embedded the VNRs twice faster than the *ILP–ST* algorithm. The embedding time for the ILP-based algorithms will increase exponentially with a larger substrate and virtual networks.

In order to get a better insight into how the resources of the substrate network are exploited, we will now examine the same single iteration (i.e., 10 embeddings). Figure 5.3 depicts the RF front–end utilization, the processing resource utilization of the DUs and the CU pool, and the overall fronthaul bandwidth utilization for *ILP–DM*, *ILP–SM*, *HEU–DM* and *HEU–ST*. In Fig. 5.3a, it can be seen that all algorithms/heuristics have embedded an equal number of VNRs and have therefore equally utilized the RF front–end resources. In particular, all algorithms/heuristics have successfully embedded the VNRs up to the 8^{th} embedding while rejected the last two VNRs.

As for the DU processing resource utilization (see Fig. 5.3b), we can observe that the *ILP–DM* algorithm keeps gradually increasing the processing resource utilization and, therefore, exhibits the best performance among all the algorithms/heuristics. In essence, with the arrival of VNRs, this increase means that due to the optimal embedding of all the VNRs, only MAC and PHY layer splits are being employed by the substrate small cells/DUs, since among the considered splits, only those splits require partial baseband signal processing at the DUs. It can also be observed that the second best performance achieves the *HEU–DM* heuristic, since as opposed to the static embedding algorithm/heuristic, which reduce the DU processing resource utilization from the 6^{th} embedding, it reduces the DU processing resource utilization from the 7^{th} embedding. This essentially means that in the case of the *HEU–DM* heuristic, most of the substrate nodes employ either MAC or PHY layer splits up to the 6^{th} embedding. While from the 7^{th} embedding some of the substrate nodes start changing the splits from the PHY/MAC split to the PHY–RF split in order to be able to exploit advanced algorithms aiming at reducing/canceling the ICI.

Regarding the static embedding algorithm/heuristic, it can be seen that their performances resemble each other in terms of the DU processing resource utilization. However, we can see that the *ILP–ST* algorithm ultimately achieves slightly higher processing resource utilization at the DUs. This proves that the *ILP–ST* algorithm has found optimal embedding solutions and has, therefore, led to more substrate nodes employing higher–layer functional splits. In essence, this means that the ICI level at those nodes is lower from the ICI threshold starting from which the PHY–RF split should be employed.

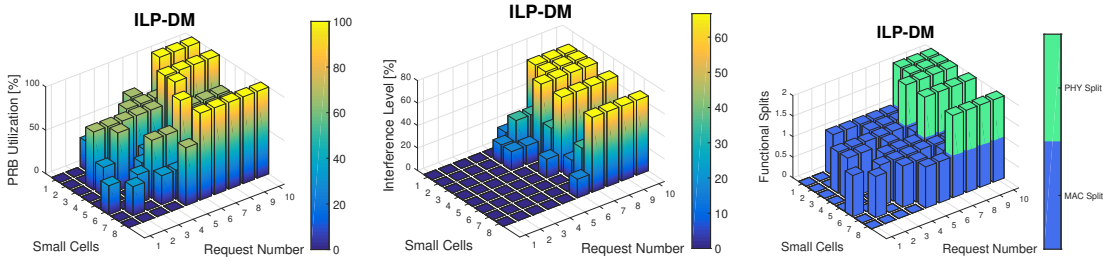
The picture is totally different for all the algorithms/heuristics in terms of the processing resource utilization at the CU pool (see Fig. 5.3c). The first observation is that for all the algorithms/heuristics the processing resource utilization at the CU pool increases, regardless of the split options employed by the substrate DUs. This stems from the fact that depending on the ICI level at the substrate small cells, which increases with the arrival of new VNRs, the functional splits at the small cells change from the higher–layer splits toward the lower–layer splits and, with this change, the processing resource utilization increases and decreases, respectively, at the CU pool and at the DUs. Since in the case of the *ILP–DM* algorithm, only the MAC split or the PHY split is employed by the substrate small cells (we will see this in Fig. 5.4c), the processing resource utilization at the CU pool is the lowest compared to the rest of the algorithm/heuristics. This is because, for those splits, the processing resource requirement at the CU pool is lower compared to the processing resource requirement at the CU pool for the PHY–RF split in which all baseband signal processing is taking place only at the CU pool. In the cases of *ILP–ST*, *HEU–DM* and *HEU–ST*, it can be observed that they achieve higher processing resource utilization at the CU pool as opposed to ones at the DUs. Moreover, *ILP–ST* and *HEU–ST* increase the processing resource utilization at the CU pool before the *HEU–DM* heuristic. Thus, the more are the substrate DUs that employ lower–layer the PHY split or the PHY–RF split, the more is the processing resource utilization at the CU pool and the less is the processing resource utilization at the DUs.

The fronthaul bandwidth utilization for all the algorithms (see Fig. 5.3d) somewhat resembles the processing resource utilization at the CU pool. This is because apart from the processing resource utilization at the CU pool, also the fronthaul bandwidth utilization increases with the increase in the number of substrate nodes that employ the PHY split or the PHY–RF split. We can observe that the fronthual bandwidth utilization for *ILP–DM* and *HEU–DM* is very low until the 6^{th} embedding with a small spike at the 5^{th} embedding for the *HEU–DM* heuristic. This is because at the 5^{th} embedding one of the substrate nodes starts using the PHY split, while at the 6^{th} embedding, the number of the host substrate nodes increases, and they all use the MAC split (this can be seen in Fig. 5.4i). We can also observe that for the static embedding algorithms the fronthaul bandwidth utilization is very low up to the 4^{th} embedding since at this point all the host substrate nodes use the MAC split, which has very low fronthaul bandwidth demand; while from the 5^{th} embedding, the fronthaul bandwidth utilization starts increasing exponentially. This huge difference in the fronthaul bandwidth utilization is due to the significant difference in the fronthaul bandwidth demands of the splits (see Table 3.2).

We will now examine the PRB utilization, the ICI level and the functional split at all the substrate small cells for all the algorithms/heuristics for a single iteration in order to better understand their relationship (see Fig. 5.4). It can be observed that the PRB utilization of individual substrate small cells for *ILP–DM*, *ILP–ST*, *HEU–DM* and *HEU–ST* varies due to different mapping decisions (see Fig. 5.4a, 5.4d, 5.4g and 5.4j). However, the network–wide PRB utilization (the sum of the PRB utilization of the small cells) is the same for all the algorithms/heuristics during each embedding. This is justified by the fact that all the algorithms/heuristics have identically accepted or rejected the VNR (see Fig. 5.3a).

Figures 5.4b, 5.4e, 5.4h and 5.4k display the ICI level in the cases of employing, respectively, *ILP–DM*, *ILP–ST*, *HEU–DM* and *HEU–ST*. It can be observed that both dynamic embedding algorithm successfully embed up to the 4^{th} VNR without creating ICI in the network. Whereas, from the 5^{th} embedding the *ILP–DM* algorithm creates a little amount of ICI at two substrate nodes, while the *HEU–DM* heuristic creates ICI at three substrate nodes. Moreover, it can be seen that at two of them the level of ICI is much higher compared to the case of employing the *ILP–DM* algorithm. After all the embeddings, we can observe that the total ICI in the network in the case of employing the *ILP–DM* algorithm is much lower than the one in the case of employing the *HEU–DM* heuristic (notice the difference in the scales). This again proves that the mapping efficiency of the *ILP–DM* algorithm is higher compared to the one of *HEU–DM*.

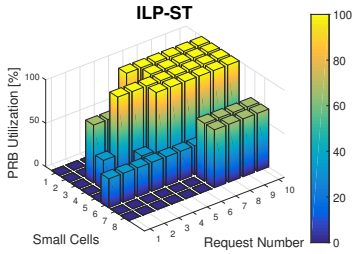
The picture is different in the cases employing static embedding algorithm/heuristic. Initially, the *HEU–ST* heuristic is as efficient as both dynamic embedding algorithm/heuristic, since like them, *HEU–ST* successfully embeds up to the 4^{th} VNR without creating ICI in the network. Nevertheless, from the 5^{th} VNR mapping, it starts introducing ICI at some of the substrate small cells. Whereas, the *ILP–ST* algorithm starts creating ICI from the 3^{th} embedding, and ultimately, results in a higher network–wide ICI compared to the *HEU–ST* heuristic. As it has been already mentioned, in this case, this is a consequence of the *HEU–ST* heuristic being slightly more efficient than the *ILP–ST* algorithm.



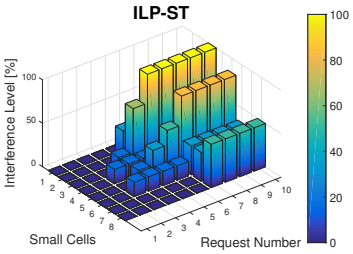
(a) PRB utilization [ILP–DM].

(b) ICI level [ILP–DM].

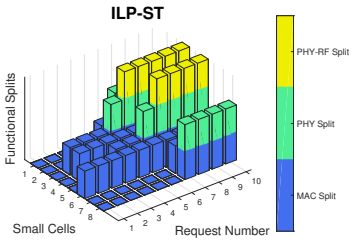
(c) Functional split [ILP–DM].



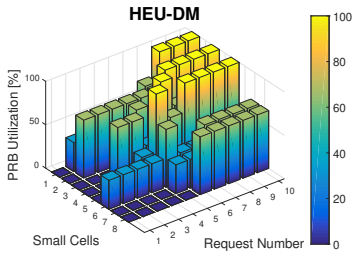
(d) PRB utilization [ILP–ST].



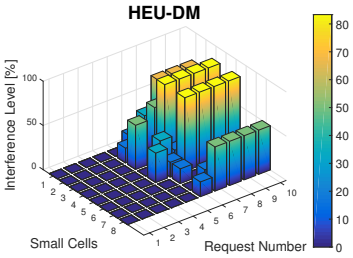
(e) ICI level [ILP–ST].



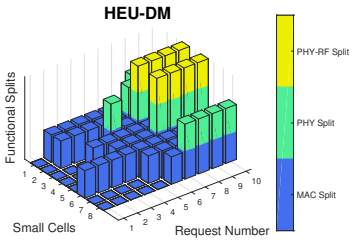
(f) Functional split [ILP–ST].



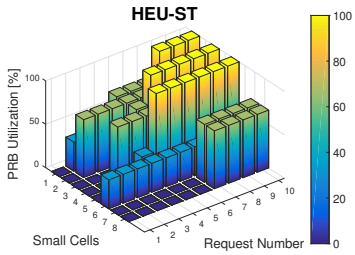
(g) PRB utilization [HEU–DM].



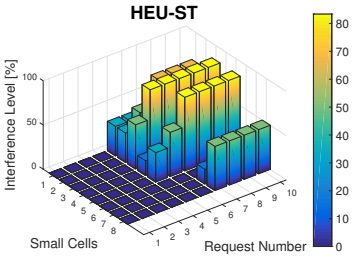
(h) ICI level [HEU–DM].



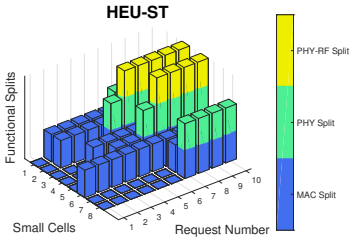
(i) Functional split [HEU–DM].



(j) PRB utilization [HEU–ST].



(k) ICI level [HEU–ST].



(l) Functional split [HEU–ST].

Figure 5.4 – PRB utilization, ICI level and functional splits of the ILP–based algorithms and of the heuristics.

Finally, let us analyze how the functional splits change at the substrate small cells as a function of changing ICI. Figures 5.4c, 5.4f, 5.4i and 5.4l illustrate the functional splits per substrate small cell for a single iteration (10 embeddings) for, respectively, *ILP–DM*, *ILP–ST*, *HEU–DM* and *HEU–ST*. In general, the lower is the ICI level, the higher–layer is the selected split, leading to a more efficient fronthaul bandwidth utilization. Among all the algorithms/heuristics, the superiority of the *ILP–DM* algorithm is obvious since, thanks to fact that it is always able to find the optimal mapping for all the VNRs, the substrate small cells only employ either the MAC split or the PHY split. Thus, being able to keep the level of ICI low from the threshold starting from which the PHY–RF split must be employed, no substrate small cell employs the PHY–RF split. It is also obvious that the *HEU–DM* heuristic exhibits the second best performance. Whereas, the performance of *ILP–ST* resembles the performance of *HEU–ST*. The reason for this is twofold. First, if a substrate small cell has been used for mapping then by default, in the considered scenario, it employs the MAC split even if there is no ICI. We can see that up to the 4^{th} embedding there is no ICI in the network in the case of employing the *HEU–ST* heuristic (see Fig. 5.4h). However, as it can be seen in Fig. 5.4i, the substrate nodes, which have been used to map the VNRs up to the 4^{th} embedding, employ the MAC split. Second, although the level of ICI at some substrate nodes are higher in the case of *ILP–ST* algorithm compared to the one of *HEU–ST* heuristic, they use the same functional split option. This is because, for each of the splits, there is an ICI range[6] defined (see Table 5.4), and if the level of ICI at those nodes is within the same range then regardless of the difference in the level of ICI, the same corresponding functional split is to be employed by those substrate nodes.

The proposed algorithms/heuristics provide MNOs with various options to select between promptness, mapping optimality and scalability. We have demonstrated that the *ILP–DM* algorithm, thanks to its dynamic embedding strategy, yields the optimal mapping solutions for all VNRs. However, it comes at the expense of significantly long embedding time (see Fig. 6.4d), which makes this algorithm not applicable to dense mobile networks with a few tens of small cells in the substrate/virtual networks. On the other hand, the *HEU–DM* heuristic, although less efficient, approximates the optimal mapping solutions found by the *ILP–DM* algorithm. Moreover, it is significantly faster in embedding VNRs, which makes it more scalable compared to *ILP–DM*. *HEU–DM*, however, is not applicable to ultra–dense mobile networks with a few hundreds of small cells in the substrate/virtual networks. If it is decided to employ a dynamic embedding algorithm/heuristic, one must also take into account the possible downsides (e.g., service interruption of the users of MVNOs) of the re–embedding of the VNRs.

In order to address the scalability problems of the exact dynamic placement algorithm and the dynamic placement heuristic, an exact static embedding algorithm (*ILP–ST*) and a static embedding heuristic (*HEU–ST*) are also proposed. We have seen that the static embedding algorithm/heuristic is less efficient in embedding VNRs and is, therefore, less efficient in employing the network resources compared to their dynamic counterparts. We have also seen that *ILP–ST* and *HEU–ST* are not comparable since their embedding decisions depend on

[6]The ICI value I for each functional split defined in Table 4 puts an upper bound to the acceptable ICI range and is used in the inequality constraint (5.12) in the ILP formulation.

their previous embedding, and they use different strategies to embed the VNRs. Thus, one must consider the availability of the network resources and the embedding time requirement in order to select the embedding algorithm that would be the most suitable for their network.

5.5 PHY-RF Split versus Flexible Split

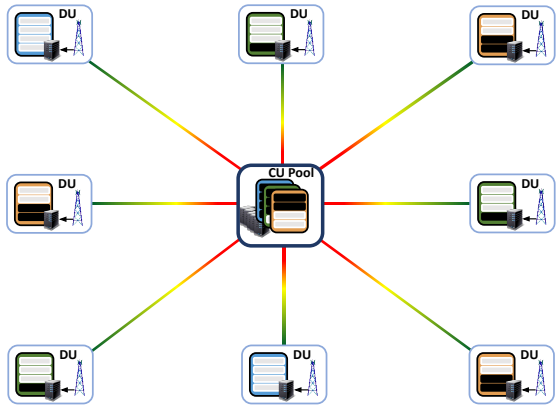
In this section, we will compare the PHY–RF split with the flexible functional split for the physical network topologies depicted in Fig. 5.5. Since the pros and cons of the PHY–RF split is well investigated in the literature [7, 99], we will highlight the pros and cons of employing the flexible functional split compared to the PHY–RF split.

5.5.1 Pros of the Flexible Functional Split over the PHY–RF Split

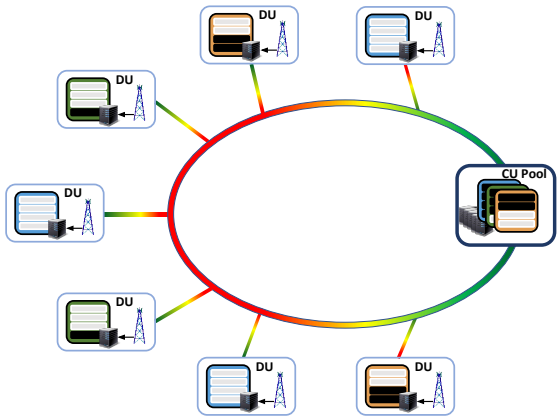
The pros of flexibly selecting the functional split options at small cells compared to the static PHY–RF split are manifold. One of the most prominent advantages of employing the flexible functional split approach is the efficient dimensioning and utilization of the fronthaul bandwidth resources. In the case of a star fronthaul network topology, in which the DUs are connected to the CU pool by means of direct optical links (see Fig. 5.5a), no multiplexing gain can be obtained on the fronthaul links since each of them is used only by a single DU attached to it. Whereas in the cases of a ring and a tree fronthaul network topology, in which the DUs are connected to the CU pool, respectively, through a high–capacity optical fiber ring and through the high–capacity optical fiber links via the anchor BSs, the aggregate fronthaul links (i.e., the ring link in Fig. 5.5b and the links from the anchor BSs to the CU pool in Fig. 5.5c) can be dimensioned based on the fronthaul bandwidth demand of the DUs that use the same fronthaul link, yielding to the fronthaul multiplexing gain. It is worth to emphasize that multiplexing gain on the fronthaul link can only be obtained if the functional split options of most of the DUs that use the same fronthaul link differ from each other at the considered time. This results in significant savings in terms of CAPEX and OPEX in the optical fronthaul network deployment, which incurs the highest portion of the total investment required to deploy the C–RAN architecture.

In order to illustrate the fronthaul multiplexing gain, let us apply the flexible functional split approach to a tree[7] fronthaul network topology, like the one depicted in Fig. 5.5c, composed of 50 DUs that are connected to a single CU pool via an anchor BS. The fronthaul network for each DU is composed of two parts: the link from the DU to the anchor BS, referred as *FH1*, and the link from the anchor BS to the CU pool, referred as *FH2*. The latter is of interest for us since this is where the multiplexing gain can be obtained. It is assumed that optical fiber fronthaul links are used and that the *FH1* links have enough capacity to support the bandwidth demand of the PHY–RF split; while the single *FH2* link has enough capacity to support the fronthaul bandwidth demand of the DUs in the case they all employ only the PHY–RF split. It is also assumed that WDM–PON technology is used and that each of the lightpaths/wavelengths

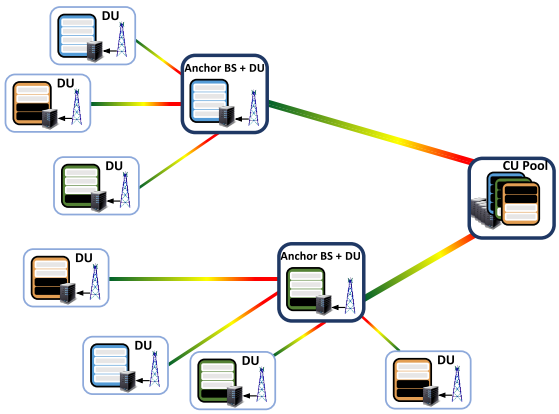
[7] The result is the same also for the ring topology where the fronthaul bandwidth estimation is for the ring link.



(a) Star topology.



(b) Ring topology.



(c) Tree topology.

Figure 5.5 – Star, ring and tree fronthaul network topologies. Optical fiber links with WDM–PON technology is used in the fronthaul network. The functional splits at the DUs and at the CU pool identified by their color correspond to the ones depicted in Fig. 5.1.

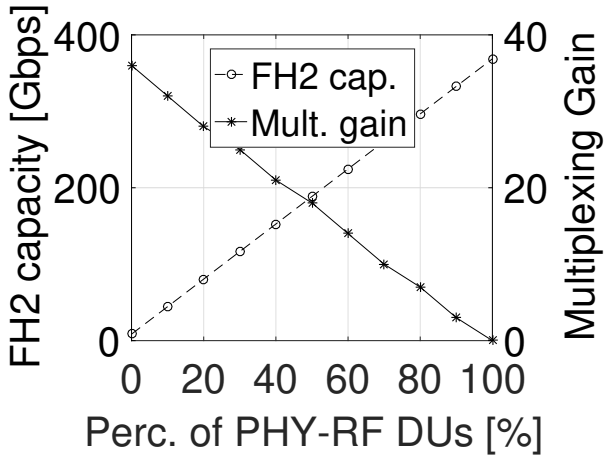


Figure 5.6 – *FH2* link capacity requirement and the multiplexing gain as a function of DUs that employ different functional splits in the tree substrate topology.

supports 10Gbps traffic. Thus, since we want to study the impact of the flexible functional split on the fronthaul deployment cost, this cost for us is just the cost of setting up lightpaths based on the fronthaul bandwidth demand of the considered DUs. Lastly, it is assumed that each DU is composed of 3 sectors each of which supporting 2×2 MIMO configuration employing LTE 20MHz bandwidth that is fully utilized by the users.

Bearing the above considerations in mind, let us analyze Fig. 5.6, which shows the *FH2* link capacity requirement and the multiplexing gain for DUs that may flexibly use any of the considered functional splits at the given time. The multiplexing gain is expressed in terms of the required number of lightpaths in order to support the *FH2* link capacity requirement. This can be easily translated to CAPEX by multiplying the multiplexing gain value by the cost of creating a lightpath. The x–axis shows the percentage of the DUs that are employing the PHY–RF split, while the rest of the DUs are employing either the PHY split or the MAC split. The trade–off between the *FH2* link capacity and the multiplexing gain is obvious. The more is the number of DUs employing the PHY–RF split at the given time, the more is the *FH2* link capacity requirement but also the less is the multiplexing gain and, therefore, so is the CAPEX savings. An important consideration that should be taken into account is that having only a few DUs that employ the PHY–RF split, even though might provide a huge multiplexing gain, might result in an underutilization of some of the lightpaths. Whereas, having many DUs that employ the PHY–RF split, although would increase the utilization of the *FH2* link, might result in a less multiplexing gain, consequently, in less savings. Therefore, one needs to find the *right* proportion of DUs that use different functional splits (e.g., the PHY–RF split, the PHY split or the MAC split) and whose traffic aggregation would yield the highest multiplexing gain while appropriately utilizing the *FH2* link.

5.5.2 Cons of the Flexible Functional Split over the PHY–RF Split

The mentioned advantage of flexibly employing functional splits comes at the expense of the network design complexity. In the case of only employing the PHY–RF split, the design of DUs is very simple and cheap. Being compact units, DUs can easily be deployed, for example, on street furniture such as on lamp posts or on billboards. The flexible functional split offsets this advantage since it has to be able to flexible support in our case the PHY–RF split, the PHY split and the MAC split. Thus, both DUs and CU pools have to support all the functionalities of the considered splits. This would require site rental cost, and therefore, increase the CAPEX.

Another challenge of flexible functional splits is that it requires a new transport protocol design that can flexibly support different splits. A universal frame format and data plane need to be designed, which can construct frames and transmits over the FH network, regardless of the selected functional split option. However, we believe that the benefits yielded by the possibility of dynamically selecting the optimal functional split option at the small cells and the fronthaul network cost savings will payoff the challenges posed by the flexible functional split approach.

6 Cost-efficient C-RAN Migration Strategy

In this chapter, we propose an approach for MNOs to adopt the C–RAN architecture with minimal investment by using the available infrastructure (e.g., site locations, transmission links). Specifically, we propose a DU–CU mapping algorithm, which effectively selects the quantity and the locations of the CU pools and assigns the CU of each DU to the appropriate CU pool. We then compare the traffic aggregation gains of C–RAN and D–RAN. Lastly, in order quantify the TCO savings that can be obtained by employing the legacy network infrastructure while migrating to C–RAN, we compare infrastructure–aware and infrastructure–unaware C–RAN migration scenarios. In both scenarios, the mapping algorithms are formulated as VNE problems using ILP techniques, and a case study is conducted on an operational LTE–A mobile network in order to assess the efficiency of the proposed algorithm.

6.1 Overview

By leveraging the fully–centralized and virtualized C–RAN architecture over densely deployed small cells, MNOs are expected to meet the ever–increasing coverage and capacity demands. A discussion on the merits and challenges of both approaches in presented in Chapter 3. Given the slow increasing pace of Average Revenue Per User (ARPU) in contrast to the traffic demand, the main goal for the MNOs, who want to harvest the benefits of the C–RAN, would be to adopt the C–RAN architecture with minimal investment. It is our standpoint that utilizing the legacy mobile network infrastructure is a viable approach to spur the realization of this goal. Towards this end, finding the optimal number, and locations of CU pools by using the network knowledge and the traffic demand statistics, and centralizing the baseband units (CUs) of eNBs at the optimal CU pools plays a pivotal role in curtailing the required investments in order to migrate from the legacy RAN architecture to the C–RAN architecture.

The previous chapters are focused more on revealing the pros and cons of different functional split options. Specifically, Chapter 4 examines mmWave wireless technology in C–RAN in various functional split scenarios, while Chapter 5 seeks to unveil the merits of flexible functional split approach. This chapter instead focuses on the C–RAN architecture design with the classical PHY–RF functional split, answering to the following question. *How to migrate from the legacy D–RAN to C–RAN with minimal investment?*

The contribution of the work presented in this chapter is threefold.

- First, we propose a DU–CU mapping algorithm in order to foster MNOs to transit from their legacy D–RAN architecture of LTE/LTE–A networks to the C–RAN architecture. The algorithm computes the number of CU pools and determines their locations by taking into account the information about the available inter–eNB transmission links, the distance between the DUs and the potential CU pools, considering the available transport network, and the hourly traffic demand per cell of the mobile network.
- Second, we compare the traffic aggregation gain of C–RAN, which is obtained as a result of the inter–sector intra–carrier traffic aggregation, with the one of D–RAN, which is obtained by activating the intra–sector inter–carrier traffic aggregation feature.
- Third, in order to quantify the economic benefit of using the available infrastructure while transiting to C–RAN, we compare two C–RAN migration scenarios (i.e., DU–CU mappings): infrastructure–aware and infrastructure–unaware C–RAN migrations.

6.2 Network Model

This section details the substrate and the virtual network models. It also presents the fronthaul network technology used in this work.

6.2.1 Substrate Network Model

Before presenting the substrate network model, we shall first provide the definition of a carrier, a cell and a sector used throughout this work.

Definition 6.1 (Carrier)**.** *The RF bandwidth (in MHz) owned by the MNO is divided into smaller RF bands called carriers. For example, if an MNO owns* $45MHz$ *of RF bandwidth then such bandwidth may be distributed across a* $10MHz$*, a* $15MHz$*, and a* $20MHz$ *carrier.*

Definition 6.2 (Cell)**.** *An eNB can have a different number of cells depending upon its configuration. In this work, we assume that each cell is assigned one and only one carrier, regardless of its RF bandwidth and the LTE band that it belongs to.*

Definition 6.3 (Sector)**.** *The coverage area of the antenna beam. Each eNB may have one or more sectors with each having one or more cells. In this work, each sector has a maximum of 3 cells. For example, if the total coverage of an eNB is divided into 3 sectors each having 3 carriers (10MHz, 15MHz and 20MHz) then it means that such an eNB has a total of 9 cells.*

The considered substrate network is a small cluster ($5km^2$) of an operational LTE–A network composed of 26 eNBs and in a total of 209 cells (see Fig. 6.1). Let $G_{\mathbf{s}} = (N_{\mathbf{s}}, E_{\mathbf{s}})$ be an *undirected* graph modeling the physical network (i.e., the network to be transformed to C–RAN), where $N_{\mathbf{s}} = N_{\mathbf{s}}^{du} = N_{enb}$ is the set of $m = |N_{\mathbf{s}}|$ DUs/eNBs. Each $n \in N_{\mathbf{s}}^{du}$ DU has a variable number of

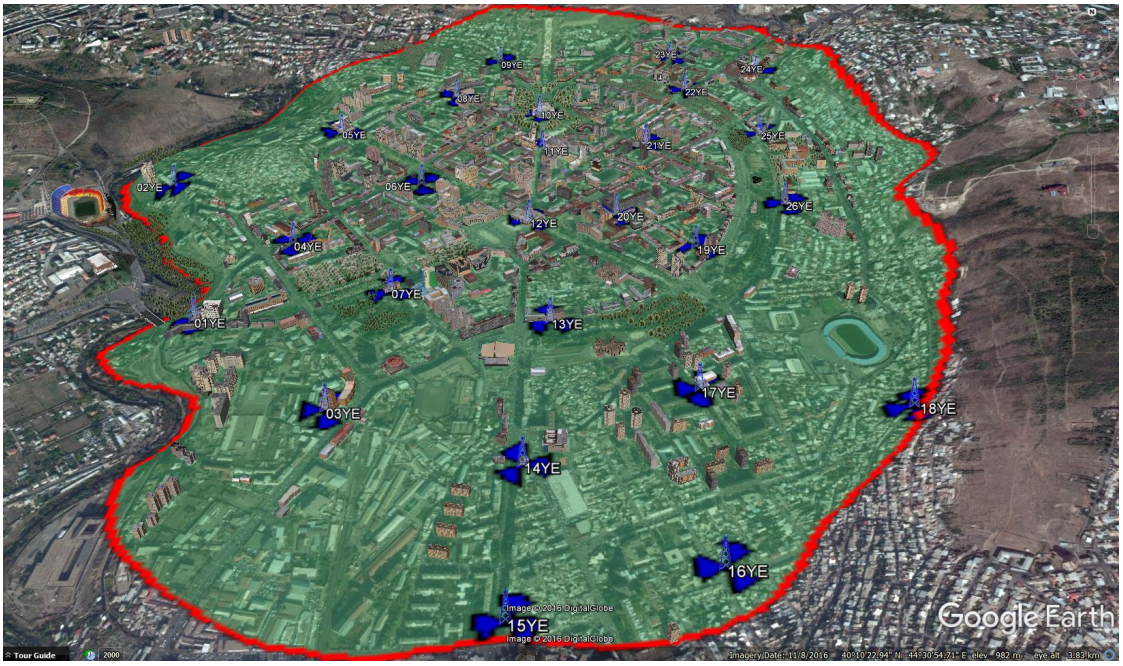


Figure 6.1 – An operational LTE–A network with 26 eNBs (in total 209 cells) in the city center of Yerevan. Each eNBs is composed of variable number of sectors, which in turn is composed of variable number of carriers employing 20MHz, 15MHz and 10MHz LTE channels.

sectors $|N^n_{sct}|$ in the set of $\{2,3,4\}$, each $s \in N^n_{sct}$ sector has a variable number of carriers/cells $|N^{n,s}_{car}|$ in the set of $\{1,2,3\}$, while each carrier $c \in N^{n,s}_{car}$ has its maximum supportable downlink throughput $\omega^{t_{max}}_{\mathbf{s},c}$ (see Table 6.5). A subset of the eNBs $N^{cu}_{\mathbf{s}} \subset N_s$ can be candidates for CU pools. More specifically, the candidate CU pools are selected as follows. The eNBs are sorted in descending order according to the number of inter–eNB transmission links, whereas if some of them have an equal number of inter–eNB links, they are sorted according to their total link capacity, and the first four eNBs are considered as CU pool candidates. The required number of CU pool(s) are then picked starting from the first eNB in the sorted list. Thus, the CU pool candidates (i.e., *anchor* eNBs) are the eNBs that interconnect multiple eNBs and serve as a relay for them to transport their signals to the core network. Selecting an *anchor* eNB as a CU pool enables MNOs to exploit the available transport network (i.e., backhaul links) in an efficient manner without having to invest *too* much in building the fronthaul network while migrating to the C–RAN architecture. It is important to mention that, since in the considered small cluster of the mobile network only optical backhaul links are available, we assume that those backhaul links can be used as fronthaul links in the C–RAN architecture.

Each substrate node (i.e., DU/eNB) is also associated with a geographic location $loc(n)$, as x, y coordinates. In order to mimic the real physical topology (i.e., the accurate inter–eNB distance) of the considered LTE–A network, x, y coordinates of the nodes are obtained by converting the real locations (longitude and latitude) of the eNBs. Lastly, let $E_{\mathbf{s}}$ model the set of inter–eNB links of the real network. An edge $e^{nm} \in E_{\mathbf{s}}$ if and only if a connection exists between DUs/eNBs $n, m \in N_{\mathbf{s}}$. The substrate network parameters can be found in Table6.1.

Table 6.1 – Input Sets and Parameters

Variable	Description
$G_{\mathbf{s}}$	Substrate network graph.
$G_{\mathbf{v}}$	Virtual network graph.
$E_{\mathbf{s}}$	Set of substrate links in $G_{\mathbf{s}}$.
$E_{\mathbf{v}}$	Set of virtual links in $G_{\mathbf{v}}$.
N_{enb}	Set of eNBs in the considered mobile network.
N_{sct}^{n}	Set of sectors of $n \in N_{enb}$ eNB.
$N_{car}^{n,s}$	Set of carriers of $s \in N_{sct}^{n}$ sector of $n \in N_{enb}$ eNB.
$N_{\mathbf{s}}^{cu}$	Set of predefined CU pool candidates in $G_{\mathbf{s}}$.
$N_{\mathbf{v}}^{cu}$	Set of virtual CUs in $G_{\mathbf{v}}$.
$N_{\mathbf{s}}^{du}$	Set of substrate DUs in $G_{\mathbf{s}}$.
$N_{\mathbf{v}}^{du}$	Set of virtual DUs in $G_{\mathbf{v}}$.
N_{hr}	Set of hours in a day.
$N_{\star}^{du}$	Number of DUs not co–located with a CU pool.
$\omega_{\mathbf{v},t}^{c}(h)$	Traffic on $c \in N_{car}^{n,s}$ virtual carrier at $h \in N_{hr}$ hour.
$\omega_{\mathbf{s},C}^{t_{max}}$	Maximum traffic on $C \in N_{car}^{n,s}$ substrate carrier.
$L_{len}(e)$	Length of $e \in E_s$ substrate links [in km].
L_{len}	Overall length of the substrate links [in km].
$L_{len}^{\star}$	Overall length of the built fronthaul links [in km].
$loc(n)$	Geographical location of n virtual/substrate node.
W_C	Bandwidth of $C \in N_{car}^{n,s}$ carrier [in MHz].
μ_b, μ_s	Big and small positive constants, respectively.
λ_b	Lightpath capacity.

6.2.2 Virtual Network Model

The considered mobile network (D–RAN) is modeled as a virtual network, which has to be mapped to the substrate network (C–RAN). Let $G_{\mathbf{v}} = (N_{\mathbf{v}}, E_{\mathbf{v}})$ be an *undirected* graph, where $N_{\mathbf{v}} = N_{\mathbf{v}}^{du} \cup N_{\mathbf{v}}^{cu}$ is the set of $m_1 = |N_{\mathbf{v}}^{du}|$ virtual DUs and $m_2 = |N_{\mathbf{v}}^{cu}|$ virtual CUs. Notice that since in C–RAN an eNB is decomposed into a DU and a CU then each $m \in N_{\mathbf{v}}^{du}$ has to have its CU $m' \in N_{\mathbf{v}}^{cu}$. Therefore, the number of virtual DUs is equal to the number of virtual CUs and is equal to the number of substrate DUs $m_1 = m_2 = m$. In essence, each $m \in N_{\mathbf{v}}^{du}$ virtual DU has its corresponding $n \in N_{\mathbf{s}}^{du}$ substrate DU, and they have the same number of sectors, carriers per sector and the same location[1]. Additionally, at each hour $h \in N_{hr}$, each carrier $c \in N_{car}^{n,s}$ of each sector $s \in N_{sct}^{n}$ of each virtual DU $n \in N_{\mathbf{v}}^{du}$ has its traffic demand $\omega_{\mathbf{v},t}^{c}(h)$, which is taken from the traffic demand statistics of the considered mobile network.

[1] Notice that a single notation is used for those parameters that are the same for the substrate and the virtual network (e.g., the location of the DUs, the number of sectors and the number of carriers).

As opposed to the substrate network model, edges $e^{nm} \in E_{\mathbf{v}}$ in the virtual network request represents the logical mapping between virtual DUs and their corresponding CUs. As an additional constraint, we require each virtual CU to be mapped to one and only one substrate CU pool. Conversely, different virtual CUs from different DUs can be mapped to the same CU pool. This enables advanced interference control algorithms such as CoMP to be employed [9], which is one of the prominent advantages of C–RAN. The virtual network parameters can be found in Table 6.1.

6.2.3 Fronthaul Network (WDM-PON)

We assume that the C–RAN fronthaul network is a WDM Passive Optical Network (PON), which is composed of two main components, ONU and OLT, performing electrical to an optical and reverse signal conversion. The former is located at DUs while the latter is located at CU pools.

It is assumed that each CU pool has one OLT rack and one OLT shelf, which has eight OLT access modules. It is also assumed that each OLT access module, through WDM Mux/DeMux, is connected to a passive splitter by a single fiber link that supports four wavelengths each with λ_b = 10Gbps capacity [139]. There is one passive splitter with 16 ports at each eNB site, while the number of ONUs at each eNB depends on the fronthaul bandwidth requirement of each DU at each eNB. For example, if an eNB has three sectors each having one 20MHz cell and one 15MHz cell (overall 6 cells) then the fronthaul bandwidth requirement, regardless of the cell utilization level[2], would be 7.37 Gbps and 5.53 Gbps in total for, respectively, 20 MHz cells and 15 MHz cells, assuming that CPRI protocol is used and that each cell has 2 × 2 MIMO antenna configuration. This would require two wavelengths ($\lambda_b = 2$), in order to meet the fronthaul bandwidth demand, which translates to two ONUs since there is one to one mapping between ONU and λ_b while four to one mapping between ONU and OLT access module.

6.3 Intra–sector Inter–carrier Traffic Aggregation in D–RAN

In this section, we formally state the intra–sector inter–carrier traffic aggregation problem in the traditional D–RAN, referred as *DRAN–TA* and present its ILP formulation.

6.3.1 Problem Statement: Intra–sector Inter–carrier Traffic Aggregation in D–RAN

In order to show how much the resource utilization efficiency of the eNBs can be increased thanks to the multiplexing gain obtained by the C–RAN architecture, we first need to quantify the resource utilization level of the eNBs of the current D–RAN architecture over the considered period of time.

[2] For a given cell/carrier, fronthaul bandwidth requirement is fixed for the traditional PHY–RF split in the C–RAN architecture and does not depend on traffic requirement of the considered cell as long as the cell is active [99].

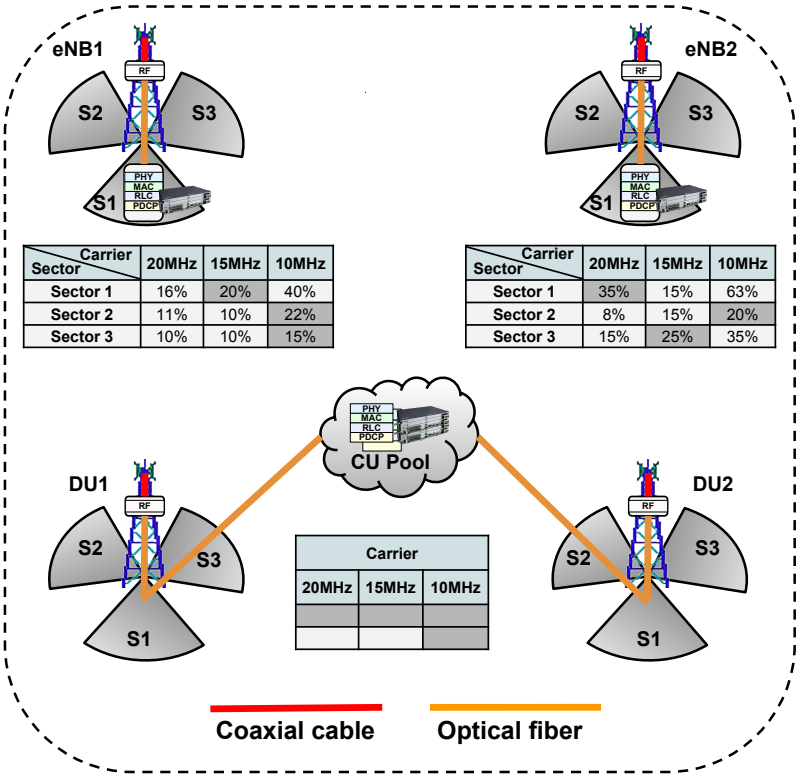


Figure 6.2 – Examples of the intra–sector inter–carrier traffic aggregation (the upper part) and the inter–sector intra–carrier traffic aggregation (the lower part) techniques.

In mobile networks, data traffic undergoes significant fluctuations depending upon the location of eNBs and the time of a day. In traditional LTE networks, when the traffic demand is low on carriers/cells, intra–sector inter–carrier traffic aggregation, as a feature, can be activated with the goal of reducing power consumption by switching off unnecessary carriers in sectors, and, at the same time, ensuring that users traffic demand is met. The upper part of Fig. 6.2 illustrates an example of the intra–sector inter–carrier traffic aggregation technique. Two three–sector eNBs are considered each having three carriers/cell under each sector. The tables below the eNBs show their corresponding average carrier/cells utilization per RF bandwidth per sector over the given period of time. After performing intra–sector inter–carrier traffic aggregation per eNB, considering the maximum capacity of each carrier/cell (see Table 6.5), we can observe that the algorithm aggregates the traffic of all carriers onto overall six carriers (three carriers per eNB, which are marked in gray in the tables). Note that this is the minimum number of carriers since, as the name of the employed traffic aggregation technique suggests, the traffic aggregation takes place under each sector separately; therefore, the number of active carriers/cells should be at least equal to the number of sectors available at the eNBs. As a side effect, this increases radio resource utilization of the *active* carriers (i.e., the carriers on which the traffic of some other carriers under the same sector have been aggregated). Radio resource utilization of eNBs is computed after activating the intra–sector inter–carrier traffic aggregation feature, which is formulated as an ILP problem that can be formally stated as follows:

Given: a small cluster of an operational LTE–A network with the location of eNBs, the sectors per eNB and the carriers/cells per sector with one–month statistics of their hourly traffic demand.

Find: the number of carriers/cells that need to be active at each eNB such that users hourly traffic demand is satisfied.

Objective: through intra–sector inter–carrier traffic aggregation, curtail the number of active carriers per sector per eNB, thereby reducing the power consumption in the network.

6.3.2 ILP Formulation: Intra–sector Inter–carrier Traffic Aggregation in D–RAN

The objective function of the intra–sector inter–carrier traffic aggregation is as follows:

$$minimize \quad \sum_{n \in N_{enb}} \sum_{s \in N_{sct}^{n}} \sum_{C \in N_{car}^{n,s}} \Phi_C V_C \tag{6.1}$$

where V_C is the cost for using the carrier $C \in N_{car}^{n,s}$ for traffic aggregation. V_C is chosen to be proportional to the size of the channel bandwidth of the carrier. Given that the carrier has enough capacity to support the aggregated traffic, the wider is the channel bandwidth of the carrier, the more expensive is its cost to be used for traffic aggregation since it is assumed that the wider is the channel bandwidth of a carrier, the more is the consumed power. By changing V_C, MNOs can give more/less priority to the carriers that they want to be used in the traffic aggregation. Table 6.2 summarizes the binary variables used in the ILP problem formulations.

In order to effectively achieve the aforementioned objective, all the following constraints have to be respected. Constraint (6.2) ensures that data traffic on the carriers at which the traffic of other carriers have been aggregated is at most equal to the maximum traffic capacity of the host carriers. Constraint (6.3) guarantees that if the traffic of any carrier of any sector of any eNB at any time is mapped to any carrier of the same sector at the same time then the host carrier is selected in mappings. Notice that this constraint allows data traffic of a carrier at different times to be mapped to different carriers belonging to the same sector of the same eNB. Lastly, Constraint (6.4) enforces the traffic of all carriers of all sectors of all eNBs to be mapped/aggregated on the host carriers. In other words, it makes sure that users' traffic demand at any time is satisfied.

$$\sum_{c \in N_{car}^{n,s}} \omega_{\mathbf{v},t}^{c}(h) \Phi_C^{c,t,h} \leq \omega_{\mathbf{s},C}^{t_{max}} \quad \forall n \in N_{enb}, \quad \forall s \in N_{sct}^{n}, \quad \forall h \in N_{hr}, \quad \forall C \in N_{car}^{n,s} \tag{6.2}$$

$$\sum_{c \in N_{car}^{n,s}} \sum_{h \in N_{hr}} \Phi_C^{c,t,h} - \mu_b \Phi_C \leq 0 \quad \forall n \in N_{enb}, \quad \forall s \in N_{sct}^{n}, \quad \forall C \in N_{car}^{n,s} \tag{6.3}$$

$$\sum_{c \in N_{car}^{n,s}} \Phi_C^{c,t,h} = 1 \quad \forall n \in N_{enb}, \quad \forall s \in N_{sct}^{n}, \quad \forall h \in N_{hr}, \quad \forall C \in N_{car}^{n,s} \tag{6.4}$$

Table 6.2 – Binary decision variables $\{0,1\}$

Variable	Description
Φ_C	Shows if $C \in N_{car}^{n,s}$ carrier of $s \in N_{sct}$ sector of $n \in N_{enb}$ eNB has been selected for traffic aggregation.
$\Phi_C^{c,t,h}$	Shows if $\omega_{\mathbf{v},t}^{c}(h)$ traffic of $c \in N_{car}^{n,s}$ virtual carrier at $h \in N_{hr}$ time has been aggregated on $C \in N_{car}^{n,s}$ substrate carrier.
Φ_m	Shows if $m \in N_{\mathbf{s}}^{cu}$ candidate CU pool has been selected as a CU pool.
$\Phi_m^{m'}$	Shows if $m' \in N_{\mathbf{v}}^{cu}$ virtual CU has been mapped to $m \in N_{\mathbf{s}}^{cu}$ CU pool.
$\Phi_n^{n'}$	Shows if $n' \in N_{\mathbf{v}}^{du}$ virtual DU has been mapped to $n \in N_{\mathbf{s}}^{du}$ substrate DU.
$\Phi_e^{e'}$	Shows if $e' \in E_{\mathbf{v}}$ virtual link has been mapped to $e \in E_{\mathbf{s}}$ substrate link.

6.4 DU-CU Mapping

In this section, we formally state the DU–CU mapping problem and present its ILP formulation.

6.4.1 Problem Statement: DU–CU Mapping

In the DU–CU mapping problem in C–RAN, an inter–sector intra–carrier traffic aggregation is performed, referred as *CRAN–TA*; that is, the aggregation of the traffic on the carriers that have the same RF bandwidth and belong to the same LTE band. Thus, the traffic on a carrier of a DU/eNB can be aggregated with the traffic on a carrier of another DU/eNB only if the CUs of those DUs are mapped to the same CU pool (i.e., their baseband signal processing is taking place at the same CU pool), those carriers have the same RF bandwidth (i.e., either 20MHz or 15MHz or 10MHz) and they are from the same LTE band. The rationale behind this approach is to guarantee a seamless transition from the D–RAN architecture to the C–RAN architecture, making sure that in the meantime users experience no channel quality degradation. The lower part of Fig. 6.2 illustrates an example of the inter–sector intra–carrier traffic aggregation technique. The entire baseband processing of the eNBs is performed at the same CU pool, harvesting the multiplexing gain in terms of the baseband processing resource as well as the radio resource. As a result, network–wide traffic aggregation is taking place. The traffic utilization on each cell/carrier per eNB before the inter–sector intra–carrier traffic aggregation is reported in the tables (see the upper part of Fig. 6.2). Notice that, as opposed to the intra–sector inter–carrier traffic aggregation that resulted in six carriers/cells being active at the eNBs (one carrier per eNB per sector), after employing inter–sector intra–carrier traffic aggregation, the traffic of all the carriers is aggregated on four carries[3]. The overall number of active carriers/cells is thus reduced compared to the previous traffic aggregation technique. More details on the inter–sector intra–carrier traffic aggregation is provided in Section 6.6.3.2. The DU–CU mapping problem can be stated as follows:

[3]Note that regardless of the number carriers in use after inter–sector intra–carrier traffic aggregation, all sectors of the eNBs still have to provide coverage to the users without necessarily using an entire carrier. In other words, a single carrier in the CU pool can be used/shared by more than one sector thanks to radio resource multiplexing.

The DU–CU mapping problem can be stated as follows:

Given: a small cluster of an operational LTE–A network with the location of eNBs, the transport network topology with the capacity of each link, the sectors per eNB, the carriers per eNB with one–month statistics of its hourly traffic demand and the candidate locations for CU pools.

Find: the overall number of carriers/cells required to meet users traffic demand, the number, and location of CU pools, and DU–CU mappings.

Objective: through the inter–sector intra–carrier traffic aggregation, minimize the overall number of carriers and the number of CU pools required to support users hourly traffic demand and minimize the fronthaul latency for each virtual link by mapping it onto the shortest substrate path.

6.4.2 ILP Formulation: DU–CU Mapping

The available network topology with traffic demand on the carriers of each sector of each eNB is modeled as a virtual network request. Upon arrival of the virtual network request, the substrate network must find the optimal mapping, aiming to minimize the objective function. This VNE process consists of two steps: the node embedding and the link embedding. In the node embedding step, each virtual node (i.e., a virtual DU or a virtual CU) in the request is mapped to a substrate node (i.e., substrate DU or substrate CU pool). In the link embedding step, each virtual link is mapped to a single substrate path. In both cases, nodes and link constraints must be satisfied.

The DU–CU mapping problem is formulated as an ILP problem whose objective function (6.5) aims at minimizing the TCO for those MNOs who own an LTE/LTE–A network and want to adopt the C–RAN architecture. Specifically, the objective function is composed of three parts:

- The first part aims at minimizing the deployment cost of the CU pools by minimizing the required number of CU pools. The candidate CU pool locations are pre–selected based on the inter–eNB connectivity ranking of each eNB. The more optical transmission links an eNB has with its neighbors, the higher is the likelihood of the eNB to become a CU pool. The rationale behind this approach is to reuse the available transmission links as much as possible.

- The second part aims at minimizing the fronthaul deployment cost by exploiting the available transmission links, and therefore, curtailing the investments required to build the fronthaul network for the C–RAN architecture. It also minimizes the fronthaul delay for each virtual link by mapping it onto the shortest substrate path from the substrate DU, on which the virtual DU has been mapped, to the substrate CU pool that has hosted the baseband processing (virtual CU) of the virtual DU.

- The last part of the objective function minimizes the required number of carriers in order to meet the traffic demand on each carrier of each virtual DU at any time. This radio resource multiplexing gain is achieved by considering the hourly traffic demand on each carrier/cell and aggregating the traffic of low utilized carriers onto fewer carriers. Notice that contrary to intra–sector inter–carrier traffic aggregation (see Sec. 6.3), in this case, an inter–sector intra–carrier traffic aggregation is taking place.

$$minimize \quad \sum_{m \in N_{\mathbf{s}}^{cu}} V_{cu}\Phi_m + \sum_{e \in E_{\mathbf{s}}} \sum_{e' \in E_{\mathbf{v}}} \mu_s L_{len}(e)\Phi_e^{e'} + \sum_{n \in N_{\mathbf{s}}^{du}} \sum_{s \in N_{sct}^{n}} \sum_{C \in N_{car}^{n,s}} W_C V_{mhz}\Phi_C \tag{6.5}$$

where V_{cu} is a CU pool built–out cost while V_{mhz} is the annual spectrum fee per MHz.

It is worthwhile to note that the second argument of the objective function is very small compared to the rest of the arguments. This is because in the CAPEX savings computation (see Sec. 6.5.1.1) we consider no fiber rollout cost since in our case the backhaul links of the legacy mobile network are employed as fronthaul links in the C–RAN deployment. Nevertheless, the second argument, although negligible, still exists in order to find the shortest path among the available substrate paths for mapping the virtual links.

The following constraints are defined for this ILP formulation. Constraint (6.6) makes sure that a CU pool candidate is selected as a CU pool as long as it has assigned at least one virtual CU. Notice that the case in which $\sum_{m' \in N_{\mathbf{v}}^{cu}} \Phi_m^{m'} = 0$ and $\Phi_m = 1$ is ruled out since the objective function (6.5) also aims at minimizing the number of CU pools. Constraint (6.7) guarantees that the traffic capacity limit of the host carriers is not violated after traffic aggregation. While Constraints (6.8) and (6.9) make sure that each virtual DU and CU are, respectively, mapped to their corresponding substrate DU and CU pool, Constraint (6.10) enforces for each virtual link $e^{nm} \in E_{\mathbf{v}}$ to be a continuous path established between the pair of the substrate DU and the CU pool on top of which the virtual DU $n \in N_{\mathbf{v}}^{du}$ and the virtual CU $m \in N_{\mathbf{v}}^{cu}$ have been mapped. In Constraint (6.10), $E_{\mathbf{s}}^{\star i}$ is the set of the fronthual links that originate from any node and directly arrive at $i \in N_{\mathbf{s}}$ node, while $E_{\mathbf{s}}^{i\star}$ is set of the fronthaul links that originates from $i \in N_{\mathbf{s}}$ node and arrive at any node directly connected to i. Lastly, Constraint (6.11) ensures that the traffic on all carriers for all sectors of all virtual DUs are mapped; in other words, users' traffic demand at each moment is satisfied.

$$\sum_{m' \in N_{\mathbf{v}}^{cu}} \Phi_m^{m'} - \mu_b \Phi_m \leq 0 \quad \forall m \in N_{\mathbf{s}}^{cu} \tag{6.6}$$

$$\sum_{n \in N_{enb}} \sum_{s \in N_{sct}^{n}} \omega_{\mathbf{v},t}^{c}(h)\Phi_C^{c,t,h} \leq \omega_{\mathbf{s},C}^{t_{max}} \quad \forall h \in N_{hr}, \quad \forall c = C \in N_{car}^{n,s} \tag{6.7}$$

$$\sum_{n \in N_{\mathbf{s}}^{du}} \Phi_n^{n'} = 1 \quad \forall n' \in N_{\mathbf{v}}^{du} \tag{6.8}$$

$$\sum_{m \in N_{\mathbf{s}}^{cu}} \Phi_m^{m'} = 1 \quad \forall m' \in N_{\mathbf{v}}^{cu} \tag{6.9}$$

$$\sum_{e \in E_{\mathbf{s}}^{\star i}} \Phi_e^{e^{nm}} - \sum_{e \in E_{\mathbf{s}}^{i\star}} \Phi_e^{e^{nm}} = \begin{cases} -1 & \text{if } i = n \\ 1 & \text{if } i = m \\ 0 & \text{otherwise} \end{cases} \tag{6.10}$$

$$\forall i \in N_{\mathbf{s}}, \quad \forall e^{nm} \in E_{\mathbf{v}}$$

$$\sum_{c \in N_{car}^{n,s}} \Phi_C^{c,t,h} = 1 \quad \forall n \in N_{\mathbf{v}}^{cu}, \quad \forall s \in N_{sct}^{n}, \quad \forall h \in N_{hr}, \quad \forall C \in N_{car}^{n,s} \tag{6.11}$$

Notice that, although C–RAN has a stringent fronthaul latency requirement, which translates to a maximum admissible length of fronthaul links, which is typically between 20 and 40 km [7], we do not use the fronthaul latency constraint in this ILP formulation since in the considered operational LTE–A network the maximum length of any possible fronthaul route for CPRI flows is far smaller than the mentioned maximum admissible length. Also, notice that we do not have a fronthaul link bandwidth capacity constraint. Initially, given the fronthaul network topology, it is assumed that all the fronthaul links have infinite capacity. This is because our goal is (i) to reuse the available links and (ii) to compute the additional bandwidth, therefore, the additional number OLT access modules and ONUs required in order to meet the network–wide fronthaul bandwidth demand.

6.5 Migration Cost Computation

In this section, we first analyze the cost to migrate from the legacy D–RAN network to C–RAN. We then compare the migration costs of two C-RAN migration scenarios: the migration cost of C–RAN when the available transport network is exploited (i.e., infrastructure–aware migration) and the migration cost of the C–RAN in which the fronthaul infrastructure must be deployed from scratch (i.e., infrastructure–unaware migration).

6.5.1 From D–RAN to C–RAN Migration Savings Computation

As detailed in Section 2.2.3.1, there are many advantages of adopting the C–RAN architecture. These advantages fall into two main categories: feature–related and cost–related advantages. The possibility to exploit advanced interference avoidance/cancellation algorithms such as FeICIC [140] or a coordinated scheduling algorithm [141] are examples of feature–related advantages; whereas, the cost–related advantages are the reduction of CAPEX and OPEX.

In this work, our focus is on the CAPEX and OPEX savings that can be obtained by using the available transport network (i.e., the links that are used to transport the backhaul traffic of the eNBs of legacy mobile networks) and network knowledge while transiting to C–RAN from legacy D–RAN.

6.5.1.1 CAPEX Savings

After mapping the virtual network request onto the substrate network, we start the computation of CAPEX savings (C_{cpx}^{save}), which is computed as the summation of the cost of the available OLT access modules, ONUs and the cost of the available fronthaul transport network (i.e., fiber rollout cost). This is because the mentioned components are reused in the C–RAN deployment. CAPEX savings can thus be computed as follows:

$$C_{cpx}^{save} = \sum_{m \in N_{\mathbf{s}}^{cu}} \Phi_m V_{olt} + |N_{enb}| V_{onu} + L_{len} V_{rol} \tag{6.12}$$

It is important to mention that while all the eNBs possess a single ONU, only a few of them possess an OLT access module.

6.5.1.2 OPEX Savings

The OPEX savings (C_{opx}^{save}), obtained as a result of migrating from D–RAN to C–RAN, is the summation of the following two costs: the annual fee for the spectrum usage per MHz per link and the annual cost of renting cell sites. The former savings, thanks to the multiplexing gain of the C–RAN architecture, is obtained by curtailing the required number of carriers with different channel bandwidths per eNB, while making sure that mobile data traffic demand is satisfied at any hour. Whereas, the latter is the result of reducing the cell site rent. Indeed, in the case of C–RAN, DUs being compact devices can be easily deployed in street furniture (e.g., on a lamp post near to the original location of the eNB in order to provide seamless coverage to users), and therefore, the cell site rent can be curtailed on average by a factor[4] of $\alpha = 0.2$, which is our assumption; while the cell site rent remains the same for the DUs that have a CU pool co–located. The annual OPEX savings can thus be computed as follows:

$$C_{opx}^{save} = \sum_{n \in N_{enb}} \sum_{s \in N_{sct}^{n}} \sum_{C \in N_{car}^{n,s}} (1 - \Phi_C) W_C V_{mhz} + \alpha N_{\star}^{du} V_{rent} \tag{6.13}$$

Notice that, although the OPEX reduction of C–RAN is also contributed by less power consumption compared to D–RAN [7], it is not considered in this study. There are several studies modeling and comparing the power consumption of C–RAN with the one of the traditional D–RAN [84, 93]. Table 6.3 summarizes all the cost parameters defined in the equations.

[4]The exact value of α depends on the country, the exact location where the DUs are deployed, and on several other factors. However, it is a fact that in the traditional C–RAN architecture (i.e., the PHY–RF functional split, option 8 in [10], between DUs and CUs) curtails the site rent for the DUs due to their small space requirement for deployment with respect to the that of eNBs [7].

Table 6.3 – Cost assumptions taken from [139]

Component	Parameter	Cost [k€]
CU pool build–out	V_{cu}	75
OLT access module, 4*10GPON ports	V_{olt}	6.5
ONU	V_{onu}	0.25
Fiber rollout per km	V_{rol}	3.8
Annual spectrum fee per MHz and link	V_{mhz}	0.06
Annual cell site rent	V_{rent}	8

6.5.2 Migration Cost Computation of C–RAN Migration Scenarios

In order to demonstrate the superiority of employing the legacy mobile network infrastructure in the C–RAN deployment in terms of CAPEX[5], we compare two C–RAN migration scenarios: the infrastructure–unaware C–RAN migration and the infrastructure–aware C–RAN migration. In the former C–RAN migration scenario, we assume that the fronthaul network of the C–RAN deployment is designed without taking into account the available optical transport network of the legacy mobile network; thus, no transport network exists, it should therefore be designed from scratch. Conversely, in the latter C–RAN migration scenario, the optical transport network is available, which actually is the backhaul network of the legacy D–RAN that is used in C–RAN as a fronthaul network. Most of the links in the backhaul network of D–RAN, however, do not have sufficient capacity in order to support the fronthaul bandwidth demand of C–RAN since there were originally designed to carry the backhaul traffic of the legacy network, which is much smaller compared to the fronthaul bandwidth demand of C–RAN. This calls for the need for increasing the capacity of the fronthaul links by means of adding the required number of OLT access modules and ONUs in order to satisfy the fronthaul bandwidth demand of C–RAN.

For the infrastructure–aware C–RAN migration scenario, we use the ILP formulation presented in Section 6.4.2. In this case, in order to compute the CAPEX (C_{cpx}^{with}), we first need to compute the fronthaul bandwidth $B_{fh}(n)$ requirement at each $n \in N_{enb}/N_{\mathbf{v}}^{du}$ eNB. According to [105], $B_{fh}(n)$ can be computed as follows:

$$B_{fh}(n) = \sum_{s \in N_{sct}^{n}} \sum_{c \in N_{car}^{n,s}} 2 f_s(c) N_o N_Q N_R \quad \forall n \in N_{\mathbf{v}}^{du} \tag{6.14}$$

where 2 accounts for the complex nature of IQ samples, while the other parameters are reported in Table 6.4. We can now compute the CAPEX C_{cpx}^{with} as follows:

$$C_{cpx}^{with} = \sum_{n \in N_{\mathbf{v}}^{du}} \left(\left\lceil \frac{B_{fh}(n)}{\lambda_b} \right\rceil - 1 \right) V_{onu} + \sum_{m \in N_{\mathbf{s}}^{cu}} \left(\left\lceil \frac{\sum_{m' \in N_{\mathbf{v}}^{cu}} B_{fh}(m'_{\star}) \Phi_m^{m'}}{4\lambda_b} \right\rceil - 1 \right) \Phi_m V_{olt} \tag{6.15}$$

[5] OPEX is the same in both C–RAN migration scenarios.

Table 6.4 – Parameters for calculating fronthaul data rates

Parameter	Symbol	Value
Bandwidth	B	[20MHz, 15MHz, 10MHz]
Sampling frequency	f_s	[30.72MHz, 23.04MHz, 15.36MHz]
Quantization bits per I/Q	N_Q	10
RX antennas	N_R	2
Oversampling factor	N_o	2

where $m'_\star$ is the corresponding virtual DU/eNB of m' virtual CU. The first and the second fractions calculate the additional number of, respectively, ONUs in DUs and OLT access modules in CU pools required in order to support the network–wide fronthaul bandwidth demand. It is important to mention that, since in this case, the optical transport network is available, there is no fiber rollout cost in the CAPEX computation. Notice that we do not consider the cost of building CU pools since, as we will see in Section 6.6, the number of CU pools (not the candidate CU pools) after embedding is the same in both scenarios.

For the infrastructure–unaware C–RAN migration scenario, we use the ILP formulation presented in Section 6.4.2 with a slight modification in the objective function (6.5), while keeping the constraints the same. The objective function for this scenario is the following:

$$minimize \quad \sum_{m\in N_{\mathbf{s}}^{cu}} V_{cu}\Phi_m + \sum_{e\in E_{\mathbf{s}}}\sum_{e'\in E_{\mathbf{v}}} V_{rol}L_{len}(e)\Phi_e^{e'} + \sum_{n\in N_{\mathbf{s}}^{du}}\sum_{s\in N_{sct}^{n}}\sum_{C\in N_{car}^{n,s}} W_C V_{mhz}\Phi_C \tag{6.16}$$

Notice that as opposed to (6.5), here the second argument is not negligible and accounts for the fiber rollout cost of the fronthaul network, since in this case it is assumed that the fronthaul network is not given. Notice also that, as opposed to the infrastructure–aware C–RAN migration scenario in which the candidates for CU pools are predefined, in this scenario, any eNB can be a candidate for a CU pool. Whereas like in the first scenario, also here it is initially assumed that the mapped links have infinite capacity, meaning that there are enough number of OLT access modules and ONUs to host the required fronthaul bandwidth on any fronthaul link. However, the exact fronthaul capacity of the mapped fiber links, which is, the exact number of OLT access modules and ONUs is computed by considering the fronthaul bandwidth requirements of the mapped virtual links.

Given that the fronthaul bandwidth requirement $B_{fh}(n)$ on each eNB $n \in N_{enb}$ is the same in both scenarios, the CAPEX C_{cpx}^{wout} in the infrastructure–unaware C–RAN migration scenario can be computed as follows:

$$C_{cpx}^{wout} = \sum_{n\in N_{\mathbf{v}}^{du}} \left\lceil \frac{B_{fh}(n)}{\lambda_b} \right\rceil V_{onu} + \sum_{m\in N_{\mathbf{s}}} \left\lceil \frac{\sum_{m'\in N_{\mathbf{v}}^{cu}} B_{fh}(m'_\star)\Phi_m^{m'}}{4\lambda_b} \right\rceil V_{olt} + L_{len}^{\star} V_{rol} \tag{6.17}$$

Table 6.5 – LTE–A Network parameters

Bandwidth	Average modul.	Max. DL traffic	RF config.	Overhead
20 MHz	16 QAM	100 Mbps	2×2 MIMO	0.25 %
15 MHz	16 QAM	75 Mbps	2×2 MIMO	0.25 %
10 MHz	16 QAM	50 Mbps	2×2 MIMO	0.25 %

The first argument computes the cost of ONUs, the second argument computes the cost of OLT access modules, while the last argument computes the fiber rollout cost.

6.6 Evaluation

The goal of this section is to compare the presented traffic aggregation algorithms (*DRAN–TA* and *CRAN–TA*) and the C–RAN migration scenarios. We shall first describe the simulation environment used in our study. We will then report on the outcomes of the numerical simulations carried out using a discrete event simulator implemented in Matlab.

6.6.1 Simulation Environment

A small cluster (26 eNBs deployed on rooftops with 209 cells in total) of an operational LTE–A mobile network in the city center of Yerevan is considered in our simulations (see Fig. 6.1). The cluster provides mobile coverage in an area of $5km^2$. The average number of RRC connected users per hour in the cluster varies between 450 and 2000 depending on the time of the day. Whereas, the number of sectors per eNB, as well as the number of carriers/cells per sector, vary in the set of, respectively, $\{2,3,4\}$ and $\{1,2,3\}$, depending upon the coverage and/or the capacity need in the area covered by the eNB. Three LTE channels, 20MHz, 15MHz and 10MHz, are used in the network, and only optical fiber links are used to connect the eNBs to the core network. This is a representative of a dense urban mobile network deployment scenario.

The maximum downlink traffic $\omega_{\mathbf{s},C}^{t_{max}}$ per carrier/cell $C \in N_{car}^{n,s}$, which has either 20MHz or 15MHz or 10MHz channel bandwidth, is derived considering 2×2 MIMO antenna configuration in every sector, the average modulation order, which is assumed to be 16 QAM since eNBs are deployed densely, and 25% overhead that takes into account Physical Broadcast Channel (PBCH), Physical Downlink Control Channel (PDCCH), reference signal, synchronization signals and channel coding. Whereas, hourly downlink traffic demands per carrier $\omega_{\mathbf{v},t}^{c}(h)$ is derived from the traffic demand statistics of the RRC connected users in the considered LTE–A network. Table 6.5 shows the parameters used to derive the maximum downlink throughput per carrier. The simulations are conducted for each day separately, and the reported results are the average of 30 simulations (one month) with 95% confidence intervals.

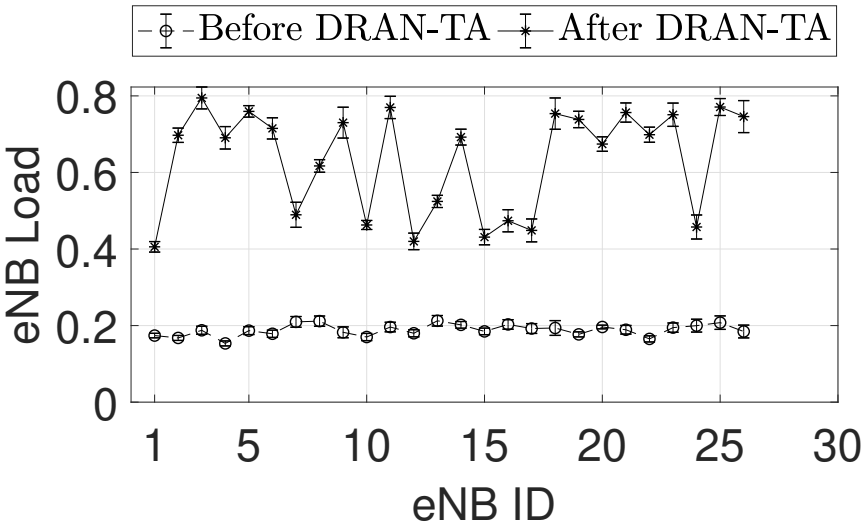
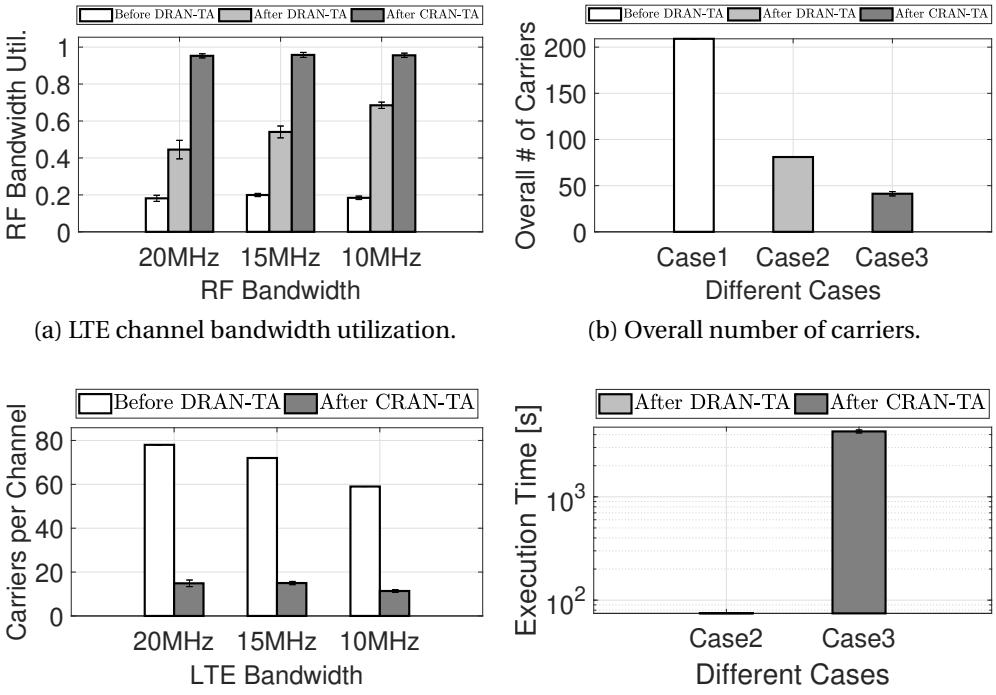


Figure 6.3 – Traffic load per eNB before and after intra–sector inter–carrier traffic aggregation.

6.6.2 Simulation Results

As it has been mentioned, the objective function for the intra–sector inter–carrier traffic aggregation problem (see formula (6.1)) aims at curtailing the number of active carriers in D–RAN by aggregating the traffic of low–utilized carriers on fewer carriers and then switching off the unnecessary carriers. For MNOs, the effect of minimizing this objective function (i.e., the activation of the intra–sector inter–carrier traffic aggregation feature) is the OPEX savings obtained in the power consumption bills. We look at this objective however from the perspective of the load at eNBs, since curtailing the number of active carriers increases the utilization of the remaining active carriers after switching off the low–utilized carries.

Figure 6.3 displays the average traffic load per eNB over 24 hours averaged for one month. The traffic load at each hour at each eNB is the summation of the carrier loads of that eNB. It can be observed that by activating the intra–sector inter–carrier traffic aggregation feature, the load of the active carriers per sector can be significantly increased. This is a consequence of aggregating the traffic of the low–utilized and already inactive carriers on the active carriers. It can also be observed, however, that there is still room for increasing the load at eNBs, and therefore, resulting in a more efficient carrier utilization. Towards this end, we adopt the C–RAN architecture and study different scenarios for migration from legacy D–RAN to C–RAN. Figure 6.4 illustrates the LTE channel bandwidth utilization, the overall number of carriers, the number of carriers per channel and the execution time of the considered traffic aggregation before the intra–sector inter–carrier traffic aggregation ($case$ 1), after the intra–sector inter–carrier traffic aggregation ($case$ 2) and after the inter–sector intra–carrier traffic aggregation in C–RAN ($case$ 3). We can observe that in the considered period of time the LTE channel bandwidth utilization barely reaches 20% at all the considered channels before the intra–sector inter–carrier traffic aggregation (see Fig. 6.4a). Although the intra–sector inter–carrier traffic aggregation feature increases the utilization of the active carriers, as we have also seen in Fig. 6.3, it can still be significantly increased since 20MHz, 15MHz and 10MHz carriers are



(a) LTE channel bandwidth utilization.

(b) Overall number of carriers.

(c) Number of carriers per channel.

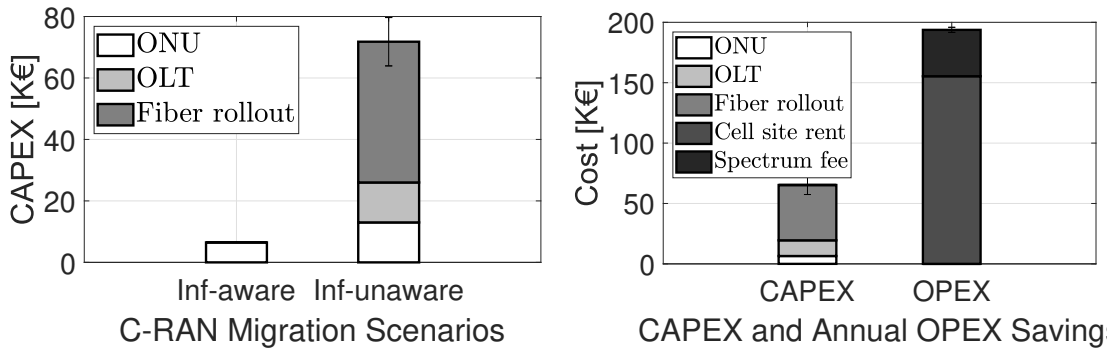
(d) Execution time.

Figure 6.4 – LTE channel bandwidth utilization, the overall number of carriers, the number of carriers per channel and the execution time for the considered cases.

underutilized by, respectively, 56%, 46% and 31% with the maximum 5% of difference from the mean values in their confidence intervals. We can observe that after adopting the C–RAN architecture the utilization of all the carriers with different channel bandwidths is increased up to approximately 95%.

The overall number of carriers for all the cases is depicted in Fig. 6.4b. Notice that, although the number of active carriers after intra–sector inter–carrier traffic aggregation (*case* 2) is reduced by 61%, this just curtails the power consumption bills without exempting the MNO from paying the fee for using the spectrum of temporarily unused carriers. Whereas, after adopting the C–RAN architecture (*case* 3), the number of active carriers is curtailed by 80%, which not only reduces the power consumption but also significantly lowers the overall fee for using the spectrum. It is worthwhile to observe that, regardless of different traffic requirements on the carriers in different days, there is no change in the overall number of carriers after the intra–sector inter–carrier traffic aggregation (i.e., the confidence interval is zero in *case* 2). This is because unlike the inter–sector intra–carrier traffic aggregation in C–RAN (*case* 3), the intra–sector inter–carrier traffic aggregation technique aggregated the traffic of carriers per sector separately.

Figure 6.4c shows the number of carriers per LTE channel for *case* 1 and *case* 3. We remind the reader that each sector has a variable number of carriers (i.e., 20MHz, 15MHz or 10MHz LTE channels, maximum one carrier from each channel). As expected, C–RAN curtails the



(a) CAPEX of C–RAN migration scenarios.

(b) TCO savings.

Figure 6.5 – CAPEX in infrastructure–aware and infrastructure–unaware C–RAN migration scenarios, and TCO savings in the former scenario.

number of carriers in all LTE channels and distributes the overall traffic more evenly across the different channels. Lastly, Fig. 6.4d compares the time required to execute the intra–sector inter–carrier traffic aggregation (*case* 2) and the time to execute the inter–sector intra–carrier traffic aggregation in C–RAN (*case* 3). It can be seen that in *case* 2 the execution time is extremely shorter compared to the execution time in *case* 3. This is justified by the fact that, as opposed to *case* 3 in which a global inter–sector intra–carrier traffic aggregation is taking place, in *case* 2 the intra–sector inter–carrier traffic aggregation is confined within each sector of each eNB. Even though one could suggest resorting to heuristic algorithms to address larger instances of the problem, in our opinion, MNOs may agree to wait even a week in order to find the optimal mapping solution for their CU–DU mapping problem rather than find a suboptimal solution very fast. This is because there is no need to frequently perform this kind of DU–CU mappings (inter–sector intra–carrier traffic aggregation).

In order to get an insight into how much the MNO can gain in terms of CAPEX and OPEX savings while migrating from their D–RAN to C–RAN using the available infrastructure of their legacy network, let us analyze Fig. 6.5. Figure 6.5a compares CAPEX of two C–RAN migration scenarios: infrastructure–aware and infrastructure–unaware migration from D–RAN to C–RAN. We remind the reader that the former implies that the available infrastructure of D–RAN (e.g., optical backhaul links, OLT access modules, ONUs, etc.) is used in the C–RAN. Whereas in the latter case, it is assumed that only the eNB site locations are available and the fronthaul infrastructure should be designed from scratch. Notice that no CU pool build–out cost is considered in CAPEX savings since in both C–RAN cases the ILP–based DU–CU mapping algorithms select two CU pools to be built in order to support centralized signal processing of all eNBs. We can observe that around seven times more capital investment is required if the available infrastructure is not considered in the C–RAN deployment. The greatest share of this CAPEX constitutes the cost of building the optical fronthaul network. Additionally, we can observe that, as opposed to the infrastructure–unaware migration, the infrastructure–aware migration requires no OLT deployment cost in this particular network setup since after the DU–CU mapping the capacity of the OLTs available in the legacy LTE–A

network is enough to meet the fronthaul traffic demand in C–RAN. It is worthwhile to note that while there is no change in the OLT and the ONU shares in CAPEX for both migration scenarios, which essentially means that the required number of ONUs, and OLTs is the same regardless of traffic variation in different days at different carriers, the fiber rollout cost share in CAPEX changes due to different fronthaul link mappings in different days. Therefore, the infrastructure–unaware fronthaul network design requires careful analysis of the traffic pattern change at different days at the eNBs.

Let us now analyze how much the MNO can gain from the considered small part of the LTE–A network in terms of CAPEX and annual OPEX savings (see Fig. 6.5b) as a result of employing the available network infrastructure. CAPEX savings come from three components: fiber rollout cost, ONUs and OLT access modules. As it is expected, the greatest part of CAPEX savings is obtained from the fronthaul deployment cost. The cost of OLT access modules and ONUs is significantly lower compared to the fiber rollout cost. Whereas, annual OPEX savings comes from reduced spectrum fee and cell site rent. It can be observed that in this particular scenario the annual cell site rent of the curtailed cell sites is around four times higher than the annual spectrum fee of the curtailed carriers. It can also be observed that apart from the fiber rollout cost variation, there is a negligible OPEX variation also in the spectrum fee. This is due to the fact that in the C–RAN architecture, after the inter–sector intra–carrier traffic aggregation, the overall number of carriers and their RF bandwidth depend on the traffic demand per carrier before the aggregation, which varies in different days.

6.6.3 Discussion

This section discusses intra–sector inter–carrier and inter–sector intra–carrier traffic aggregation techniques, respectively for D–RAN and C–RAN. Additionally, a discussion on the infrastructure–aware C–RAN migration approach is presented.

6.6.3.1 Intra–sector Inter–carrier Traffic Aggregation in D–RAN

The intra–sector inter–carrier traffic aggregation feature has several pros. For instance, if the traffic demand on the carriers is low then by activating this feature in current LTE/LTE–A networks the overall power consumed by carriers can be reduced by deactivating the unused carriers (see the upper part of Fig. 6.2). Whereas if the required traffic on the carriers is high, the inactive carriers can then be reactivated, meeting the traffic demand and providing the possibility of carrier aggregation, which is one of the prominent features of LTE–A technology. Nonetheless, the cons of the intra–sector inter–carrier traffic aggregation lay in the fact that the overall OPEX can be reduced only in terms of power consumption cost for the period in which some of the carriers are inactive. Moreover, by aggregating the intra–sector inter–carrier traffic, the host carriers might still be underutilized (see Fig. 6.3).

6.6.3.2 Inter–sector Intra–carrier Traffic Aggregation in C–RAN

The inter–sector intra–carrier traffic aggregation in C–RAN ($case$ 3) has an important advantage over the intra–sector inter–carrier traffic aggregation in D–RAN ($case$ 2). The carriers at different frequency bands (e.g., Band 3, Band 5, Band 7) with different RF bandwidths (e.g., 10MHz, 15MHz and 20MHz) have their peculiarities. For example, high–frequency bands (e.g., Band 7, $2620-2690$MHz) are more beneficial for LTE networks' construction in the regions with a large population where high speed of data transfer is required. Whereas, LTE network deployments in the low–frequency bands (e.g., Band 5, $869-894$MHz) is very appealing from the cost viewpoint and is ideal for the regions with low population (e.g., suburban areas, villages). The pros of such a deployment are costs, better penetration inside buildings and coverage of large territories. Thus, some users may experience performance degradation while forcing them to handover from one carrier to another under the same sector. Whereas, this problem does not exist in $case$ 3, since in this case an inter–sector intra–carrier traffic aggregation is taking place.

6.6.3.3 Infrastructure–aware C–RAN migration

In Section 6.6, we have seen that significant TCO savings can be obtained by exploiting the legacy mobile network infrastructure while migrating to C–RAN. In this work, however, just a small part of an operational LTE–A network is considered. The TCO savings will increase with the increase of the network size. More accurate TCO savings estimation depends not only on the size of the network but also on the geographical areas (e.g., urban, suburban, rural), the link types of the transport network, population, traffic demand and on several other factors. Thus, although the considered scenario the obtained results cannot be extrapolated to all kind of deployments, it is, however, a good example of urban mobile network deployment.

7 Conclusions and Future Work

This chapter summarizes the contributions of the research presented in this dissertation and provides concluding remarks. We also suggest some promising research directions for future work.

7.1 Conclusions

5G mobile network is on the horizon with the promise to revolutionize the communication landscape. Fueled by "Internet of Everything" broad concept, 5G is expected to support massive device connectivity, delivering Internet access to cars, robots, sensors, buildings, and many other objects, resulting in a formidable amount of mobile data traffic increase. Besides, 5G enables a wide variety of services, including massive broadband, machine to machine communications, tactile Internet, virtual/augmented reality, high definition media delivery, autonomous vehicles, real–time monitoring and control, and so on. Many of these services have stringent QoS requirements in terms of data transmission rate, latency, jitter, reliability, and mobility. All the aforementioned requirements will mandate MNOs to perform costly upgrades in their network in order to remain competitive in the market while satisfying the demands of their subscribers.

Mobile network densification along with C–RAN has the potential to meet some of the requirements of the 5G mobile networks while easing the network management and enhancing resource utilization efficiency in the network. Indeed, small cells are being pervasively deployed in mobile networks, boosting their system capacity and increasing the achievable data rates especially for the cell–edge users. The main challenge posed by small cells, however, lies in the increased level of ICI, which might result in a significant users' performance deterioration. C–RAN has been proposed as an efficient way to tackle this challenge by enabling the use of advanced interference cancellation/avoidance techniques. Nonetheless, C–RAN raises a series of challenges some of which we addressed in the presented study.

In this doctoral dissertation, we investigated the potential of the C–RAN architecture along with its several promising functional split options to be used in 5G mobile networks. We studied the trade–offs between different functional splits, analyzed in detail their deployment scenarios, benefits and challenges, having an ultimate goal of providing a set of algorithms to MNOs for adopting the C–RAN architecture, employing the appropriate functional splits and exploiting network resources in the most efficient way.

As it has been already mentioned, the PHY–RF split in the C–RAN architecture is the most fruitful functional split option. However, its tight requirements in term of fronthaul bandwidth and latency can undermine its economical convenience since although optical fiber, which is considered to be the best fronthauling option, is capable of carrying high bandwidth with low latency, it incurs huge deployment cost. Therefore, providing optical fiber connectivity to thousands of small cells, which are expected to be deployed in order to meet the future mobile data traffic demand, cannot simply be a viable solution. This calls the need for exploring other fronthauling mediums/technologies that could transport high data rates and other functional splits within the RAN protocol stack that could alleviate the fronthaul network requirements yet enabling MNOs to reap some of the centralized processing benefits in C–RAN without sacrificing all of them.

In Chapter 4, we considered mmWave technology for the fronthaul network and examined its performance in different functional split scenarios. Specifically, using ILP techniques, we formulated and solved a CU placement problem as a VNE problem whose goal was to minimize the fronthaul bandwidth utilization while curtailing the number of active mmWave interfaces, and therefore, also minimizing the power consumption in the fronthaul network. A placement heuristic has also been proposed, in order to address the VNE problem in practical–sized networks. We demonstrated that a substantial difference exists in the fronthaul bandwidth utilization for different functional split cases. This evidences that using the appropriate functional split option plays a pivotal role from the perspective of optimal fronthaul bandwidth utilization. Besides, we conducted a relative fronthaul deployment cost analysis between wireless mmWave fronthaul and optical fiber. It can be concluded that although the fronthaul bandwidth of the mmWave technology does not scale like the one of the optical fiber, it, however, can be a cost–efficient fronthauling option especially for the high–layer functional splits as their bandwidth requirements grow with the resources actually used by the users as opposed to the one in the case of PHY–RF split.

With regards to CU pool deployment in the network, we can conclude that two of the important factors that should be taken into account when deploying CU pools are the distance between CU pools and DUs, and the deployment cost of CU pools such as to minimize the latter, avoid underutilization of CU pools while meeting the traffic demand at DUs connected to CU pools.

As it has been detailed in Section 3.5, each functional split has its pros and cons. In general, the lower–layer is the functional split level within the RAN protocol stack, the more are the resource centralization benefits, but also the more severe are the fronthaul bandwidth and latency requirements. Thus, implementing a specific functional split may be justifiable only

in a certain situation, while in other situations it may not be the best choice, entailing a waste of network resources. It is, therefore, of paramount importance to scrutinize mobile network capabilities (e.g., fronthaul medium, its capacity, distance between DUs and the CU pool, offered services, etc.), the requirements of functional splits and the need for employing advanced inter–cell coordination techniques in order to implement a particular functional split at a small cell.

Given continuously changing mobile network requirements (e.g., traffic demand, the need for supporting new services) and interference scenarios, in our opinion, it is not a viable solution to implement a certain fixed functional split at a small cell, especially in the long run. In Chapter 5, we presented an approach that allowed MNOs to flexibly select the most optimal functional split option at their small cells. Specifically, dynamic and static CU placement problems have been formulated as VNE problems, aiming at minimizing the network–wide ICI and the fronthaul bandwidth utilization by selecting the appropriate functional split at each small cell at the given time. ILP–based algorithms and heuristics have been developed in order to solve both dynamic and static VNE problems. We considered PHY–RF, PHY II, and MAC splits, demonstrating that the functional split option at a small cell can be dynamically changed based on the level of ICI. While doing so, the selected functional split would enable the use of an appropriate technique/algorithm aiming to reduce/cancel the ICI while optimizing the fronthaul bandwidth utilization. We have seen that the proposed algorithms provide a range of options for MNOs to opt out the one that would best cater to the needs in their specific scenario. Simulation results have proven that our algorithms are designed to effectively select functional splits for small cells in, respectively, dense and ultra–dense small–cell deployment scenarios, depending on the specific scenario, each coming with different execution time requirement and mapping solution optimality. Lastly, by conducting simulations over a star, a ring and a tree fronthaul network topologies, we corroborated that, although flexible functional split poses some additional challenges (e.g., the need for designing functional–split–agnostic frame format) it enables MNOs to garner its benefits, such as multiplexing gain in the fronthaul network, that cannot be obtained when a certain fixed functional split is used.

It has been previously discussed that the PHY–RF split in C–RAN has no restriction in terms of resource centralization and virtualization gain, although it requires huge fronthaul bandwidth, which, in turn, increases the fronthaul deployment cost. In Chapter 6, we presented an approach for MNOs to migrate from their legacy D–RAN architecture to the C–RAN architecture with minimal investments. A case study has been conducted on a small cluster of a commercial LTE–A mobile network, aiming to estimate the efficiency of the proposed migration algorithm. From D–RAN to C–RAN migration problem has been modeled as a VNE problem and solved using an ILP–based algorithm.

We demonstrated that in comparison with the infrastructure–unaware C–RAN migration algorithm, our infrastructure–aware C–RAN migration algorithm significantly curtails both CAPEX and OPEX by exploiting the backhaul links of the available mobile network, the data traffic demand statics and, in general, the architectural knowledge on the legacy network in C–RAN

migration. Moreover, we demonstrated that C–RAN, thanks to intra–carrier inter–sector traffic aggregation, yields significantly higher radio resource multiplexing gain compared to the one that can be obtained in the legacy D–RAN by employing the intra–sector inter–carrier traffic aggregation technique. As for the TCO savings, we can conclude that although our approach leads to considerable TCO savings in C–RAN, its accurate estimation is scenario–specific and depends on various parameters such as the number of eNBs, the desired functional split options to be implemented at the eNBs, the inter–eNB, the transport network capacity, the traffic demand and etc.

7.2 Future Work

Inspired by the functional selection problem complexity and significantly different fronthaul requirements and benefits, we have outlined the following research directions in order to delve deeper into the problem better justifying the impact of each functional split option and the need for its use.

- As we have seen, numerous factors have to be taken into consideration in order to select the optimal functional split option for a small cell. In Chapter 6, given the fronthaul capacity and the radio frequency bandwidth, we proposed an approach to dynamically select the optimal functional split option based only on the estimated ICI. One of the research directions that could be interesting to explore is to look for other parameters that can play key roles in selecting a functional split option. Once the number of decision parameters increases, it may call the need for designing a Self–Organizing Network (SON) algorithm that would allow for an exchange of these parameters among the small cells that belong not only to the same CU pool but also to other CU pools. These would foster cooperation between different small cells, making them more conversant about their neighbors, and therefore, leading to an automated and optimal functional split selection for all of them while ensuring that the decision parameters such as ICI remain within the acceptable range for the functional split employed by each small cell.

- It has been previously emphasized that one of the merits of the classical PHY–RF split in C–RAN is the compact size of DUs, which make them easy to deploy, exempting MNOs from the need to purchase or rent real estate for deploying their BSs. This stems from the fact that DUs in the PHY–RF split case perform no baseband signal processing. Whereas, when flexible functional split comes into play, depending upon which functionalities may be executed at DUs, it might offset the benefits yielded by the small form factor of DUs. Thus, another intriguing research direction could be to conduct an in–depth analyses on the level of functionalities the DUs should be endowed with in order for the flexible functional split to be an economically viable approach. This has to take into consideration also the design complexity of DUs.

- Recently, packet–based transport technologies such as Ethernet started to gain ground as a cost–efficient way to transport the fronthaul/backhaul traffic since they constitute the major part of the transport network in the present–day mobile networks. They offer the benefits of the packet–based transmission such as statistical multiplexing, flexible routing, and better scalability; nevertheless, as highlighted in Section 3.5.2.3, they pose additional challenges on the fronthaul network in terms of delay, jitter, and synchronization. Yet another interesting research direction could be to try applying the flexible functional split approach to the packet–based fronthaul network. This would introduce another level of complexity where functional split choice at the small cells, due to their fronthaul delay requirements, would affect the routing decisions.

- While we have already studied the DU–CU mapping problem in Chapter 6, the coverage area ($5km^2$) and the quantity of the small cells (26 eNBs) was rather small. We, therefore, considered only a fixed DU–CU mapping. In other words, once a DU has been associated with a CU pool, it could not, later on, change its association. The next interesting problem that is worthwhile studying is to find the optimal DU–CU mapping in larger–scale mobile networks composed of several hundreds of small cells and tens of CU pools based on the dynamically selected functional split option. On occasion, this would result in some DUs to change their CU pool association and, therefore, also the routing due to their functional split option in order to meet the fronthaul delay requirement imposed by the employed split option. This DU–CU mapping problem could be studied with the goal of, for example, load balancing the CU pools.

- Lastly, with regards to the optimization techniques used to formulate the optimization algorithms, it would of great interest for us to investigate the capabilities of OMT [22] to be used to formalize and solve the studied problems, drawing a comparison with the ILP techniques in terms of their execution time and the optimality of mapping solution.

Bibliography

[1] *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021*. Tech. rep. Cisco, 2017.

[2] *Ericsson Mobility Report, 2016-2021*. Tech. rep. Ericsson, June, 2017.

[3] *NSN Liquid Core*. URL: http://nsn.com/portfolio/liquidnet/liquidcore.

[4] Imtiaz Parvez et al. "A survey on low latency towards 5G: RAN, core network and caching solutions". In: *IEEE Communications Surveys & Tutorials* (2018).

[5] *3GPP Technical Specification Group, Release 15*. Tech. rep. 3GPP, 2018.

[6] *The Road to 5G: Drivers, Applications, Requirements and Technical Development*. URL: https://www.huawei.com/minisite/5g/img/GSA_the_Road_to_5G.pdf.

[7] *C-RAN, The Road Towards Green RAN*. Tech. rep. China Mobile, 2011.

[8] *Small cell virtualization functional splits and use cases*. Tech. rep. Small Cell Forum, 2015.

[9] Stefania Sesia, Issam Toufik, and Matthew Baker. *LTE - the UMTS long term evolution: from theory to practice*. Wiley, 2011. ISBN: 978-0-470-66025-6.

[10] *Technical Specification Group Radio Access Network; Study on New Radio Access Technology; Radio Access Architecture and Interfaces*. Tech. rep. 3GPP TR 38.801 V2.0.0, 2017.

[11] *Further study on critical C-RAN technologies*. Tech. rep. NGMN, 2015.

[12] Huang Jinri and Yuan Yannan. "White Paper of next Generation Fronthaul Interface". In: (2015).

[13] *Mint Mobile*. [Online]. Available: https://www.mintmobile.com/.

[14] *Affinity Cellular*. [Online]. Available: https://www.affinitycellular.com/.

[15] Anath Fischer et al. "Virtual network embedding: A survey". In: *Communications Surveys Tutorials, IEEE* 15.4 (2013), pp. 1888–1906.

[16] Edoardo Amaldi et al. "On the computational complexity of the virtual network embedding problem". In: *Electronic Notes in Discrete Mathematics* 52 (2016), pp. 213–220.

[17] Zoran Despotovic et al. "VNetMapper: A fast and scalable approach to virtual networks embedding". In: *Proc. of IEEE ICCCN*. Ed. by. Shanghai, China, 2014.

[18] Mosharaf Chowdhury, Muntasir Raihan Rahman, and Raouf Boutaba. “Vineyard: Virtual network embedding algorithms with coordinated node and link mapping”. In: *IEEE/ACM Transactions on Networking (TON)* 20.1 (2012), pp. 206–219.

[19] Sheng Zhang et al. “Virtual network embedding with opportunistic resource sharing”. In: *IEEE Transactions on Parallel and Distributed Systems* 25.3 (2014), pp. 816–827.

[20] *Matlab*. [Online]. Available: https://www.mathworks.com/help/optim/ug/intlinprog.html.

[21] *IBM ILOG CPLEX Optimizer*. [Online]. Available: http://www.ibm.com/software/commerce/optimization/cplex-optimizer/.

[22] Roberto Sebastiani and Silvia Tomasi. “Optimization modulo theories with linear rational costs”. In: *ACM Transactions on Computational Logic (TOCL)* 16.2 (2015).

[23] Davit Harutyunyan and Roberto Riggio. “Functional Decomposition in 5G Networks”. In: *IFIP International Conference on Autonomous Infrastructure, Management and Security*. Springer. 2016, pp. 62–67.

[24] Roberto Riggio et al. “SWAN: Base-band units placement over reconfigurable wireless front-hauls”. In: *2016 12th International Conference on Network and Service Management (CNSM)*. IEEE. 2016, pp. 28–36.

[25] Davit Harutyunyan et al. “CU Placement over a Reconfigurable Wireless Fronthaul in 5G Networks with Functional Splits”. In: *Wiley International Journal of Network Management* (2019, Accepted).

[26] Davit Harutyunyan and Roberto Riggio. “Flexible functional split in 5G networks”. In: *2017 13th International Conference on Network and Service Management (CNSM)*. IEEE. 2017, pp. 1–9.

[27] Davit Harutyunyan and Roberto Riggio. “Flex5G: Flexible functional split in 5G networks”. In: *IEEE Transactions on Network and Service Management* 15.3 (2018), pp. 961–975.

[28] Davit Harutyunyan and Roberto Riggio. “How to Migrate From Operational LTE/LTE–A Networks to C–RAN With Minimal Investment?” In: *IEEE Transactions on Network and Service Management* 15.4 (2018), pp. 1503–1515.

[29] Faqir Zarrar Yousaf et al. “NFV and SDN-Key Technology Enablers for 5G Networks”. In: *IEEE Journal on Selected Areas in Communications* 35.11 (2017), pp. 2468–2478.

[30] Bruno Astuto A Nunes et al. “A survey of software-defined networking: Past, present, and future of programmable networks”. In: *IEEE Communications Surveys & Tutorials* 16.3 (2014), pp. 1617–1634.

[31] *ETSI Network Function Virtualization (NFV): Architectural Framework. ETSI GS NFV 002 v1.1.1 (2013-10)*. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/nfv/001_099/002/01.01.01_60/gs_nfv002v010101p.pdf.

[32] Nathalie Omnes et al. “A programmable and virtualized network & IT infrastructure for the internet of things: How can NFV & SDN help for facing the upcoming challenges”. In: *Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on*. IEEE. 2015, pp. 64–69.

[33] Van-Giang Nguyen, Truong-Xuan Do, and YoungHan Kim. “SDN and virtualization-based LTE mobile network architectures: A comprehensive survey”. In: *Wireless Personal Communications* 86.3 (2016), pp. 1401–1438.

[34] *OpenFlow Switch Specification 1.5.1*. URL: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-switch-v1.5.1.pdf.

[35] *Open Networking Foundation*. 2017. URL: http://www.opennetworking.org.

[36] *Network Function Virtualisation (NFV): Use Cases. ETSI GS NFV 001 v1.1.1 (2013-10)*. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/NFV/001_099/001/01.01.01_60/gs_NFV001v010101p.pdf.

[37] *Common Public Radio Interface (CPRI); Interface Specification*. Tech. rep. CPRI Specification V7.0, 2015.

[38] Aleksandra Checko et al. “Cloud ran for mobile networks–a technology overview”. In: *Communications Surveys & Tutorials, IEEE* 17.1 (2014), pp. 405–426.

[39] *Cloud RAN – the benefits of virtualization, centralization and coordination*. URL: http://www.ericsson.com/res/docs/whitepapers/wp-cloud-ran.pdf.

[40] *The Benefits of Cloud-RAN Architecture in Mobile Network Expansion*. URL: http://www.fujitsu.com/downloads/TEL/fnc/whitepapers/CloudRANwp.pdf.

[41] Checko Aleksandra, Henrik Lehrmann Christiansen, and Michael Stübert Berger. “Evaluation of energy and cost savings in mobile Cloud RAN”. In: *Proc. of OPNETWORK*. 2013.

[42] Line MP Larsen, Aleksandra Checko, and Henrik L Christiansen. “A survey of the functional splits proposed for 5G mobile crosshaul networks”. In: *IEEE Communications Surveys & Tutorials* (2018).

[43] *"EUTRA Medium Access Control (MAC) Specification"*. Tech. rep. 3GPP TS36.321.

[44] *Amazon Web Services (AWS)*. URL: https://aws.amazon.com/.

[45] *Microsoft Azure*. URL: https://azure.microsoft.com/.

[46] D. Breitgand et al. “Network aware virtual machine and image placement in a cloud”. In: *Proc. of IEEE CNSM*. 2013.

[47] M. Barshan, H. Moens, and F. De Turck. “Design and evaluation of a scalable hierarchical application component placement algorithm for cloud resource allocation”. In: *Proc. of IEEE CNSM*. 2014.

[48] M. Barshan et al. “Algorithms for efficient data management of component-based applications in cloud environments”. In: *Proc. of IEEE NOMS*. 2014.

[49] M. Ghaznavi et al. “Elastic Virtual Network Function Placement”. In: *Proc. of IEEE CloudNet*. Niagara Falls, Canada, 2014.

[50] Milad Ghaznavi et al. “Elastic virtual network function placement”. In: *Cloud Networking (CloudNet), 2015 IEEE 4th International Conference on*. IEEE. 2015, pp. 255–260.

[51] Brendan Jennings and Rolf Stadler. “Resource Management in Clouds: Survey and Research Challenges”. English. In: *Journal of Network and Systems Management* (2014), pp. 1–53. ISSN: 1064-7570. DOI: 10.1007/s10922-014-9307-7.

[52] Roberto Riggio et al. “Scheduling Wireless Virtual Networks Functions.” In: *IEEE Trans. Network and Service Management* 13.2 (2016), pp. 240–252.

[53] Qian Hu, Yang Wang, and Xiaojun Cao. “Virtual network embedding: An optimal decomposition approach”. In: *Proc. of IEEE ICCCN*. Shanghai, 2014.

[54] Ashraf A Shahin. “Virtual network embedding algorithms based on best-fit subgraph detection”. In: *arXiv preprint arXiv:1502.02768* (2015).

[55] NM Mosharaf Kabir Chowdhury, Muntasir Raihan Rahman, and Raouf Boutaba. “Virtual network embedding with coordinated node and link mapping”. In: *Proc. of IEEE INFOCOM*. Rio de Janeiro, 2009.

[56] Stefano Coniglio et al. “Optimal offline virtual network embedding with rent-at-bulk aspects”. In: *arXiv preprint arXiv:1501.07887* (2015).

[57] Stefano Coniglio, AMCA Koster, and Martin Tieves. “Virtual network embedding under uncertainty: exact and heuristic approaches”. In: *Proc. of IEEE DRCN*. Kansas City, MO, 2015.

[58] Yong Zhu and Mostafa H Ammar. “Algorithms for Assigning Substrate Network Resources to Virtual Network Components.” In: *Proc. of IEEE INFOCOM*. Barcelona, Spain, 2006.

[59] Riccardo Guerzoni et al. “A novel approach to virtual networks embedding for SDN management and orchestration”. In: *Proc. of IEEE NOMS*. Krakow, Poland, 2014.

[60] Cihangir Beşiktaş et al. “Flexible bandwidth-based virtual network embedding”. In: *Proc. of IEEE NOMS*. Istambul, Turkey, 2016.

[61] Long Gong et al. “Toward profit-seeking virtual network embedding algorithm via global resource capacity”. In: *Proc. of IEEE INFOCOM*. Toronto, ON, 2014.

[62] Sen Su et al. “Energy-aware virtual network embedding through consolidation”. In: *Proc. of IEEE INFOCOM WKSHPS*. Orlando, FL, 2012.

[63] Pengchao Han, Lei Guo, and Yejun Liu. “Virtual Network Embedding in SDN/NFV based Fiber-Wireless Access Network”. In: *Proc. of IEEE ICSN*. Jeju, 2016.

[64] Adil Razzaq and Muhammad Siraj Rathore. “An approach towards resource efficient virtual network embedding”. In: *Proc. of IEEE INTERNET*. Valcencia, 2010.

[65] Yang Wang, Phanvu Chau, and Fuyu Chen. “A framework for security-aware virtual network embedding”. In: *Proc. of IEEE ICCCN*. Las Vegas, NV, 2015.

[66] Giorgos Chochlidakis and Vasilis Friderikos. “Robust virtual network embedding for mobile networks”. In: *Proc. of IEEE PIMRC*. Hong Kong, 2015.

[67] Min Feng et al. “Topology-aware virtual network embedding based on multiple characteristics”. In: *Proc. of IEEE ICC*. Sydney, NSW, 2014.

[68] Fida-E Zaheer, Jin Xiao, and Raouf Boutaba. “Multi-provider service negotiation and contracting in network virtualization”. In: *Proc. of IEEE NOMS*. Osaka, 2010.

[69] Kailing Guo et al. “Particle swarm optimization based multi-domain virtual network embedding”. In: *Proc. of IEEE IM*. Ottawa, ON, 2015.

[70] Ines Houidi et al. “Virtual network provisioning across multiple substrate networks”. In: *Computer Networks* 55.4 (2011), pp. 1011–1023.

[71] Nick Feamster, Lixin Gao, and Jennifer Rexford. “How to lease the Internet in your spare time”. In: *ACM SIGCOMM Computer Communication Review* 37.1 (2007), pp. 61–64.

[72] Meng Shen et al. “Towards efficient virtual network embedding across multiple network domains”. In: *Proc. of IEEE IWQoS*. Hong Kong, 2014.

[73] Mosharaf Chowdhury, Fady Samuel, and Raouf Boutaba. “Polyvine: policy-based virtual network embedding across multiple domains”. In: *Proc. of ACM SIGCOMM*. 2010.

[74] Leonardo Richter Bays et al. “Virtual Network Embedding in Software-Defined Networks”. In: ().

[75] Sen Su et al. “Energy-aware virtual network embedding”. In: *IEEE/ACM Transactions on Networking* 22.5 (2014), pp. 1607–1620.

[76] Minlan Yu et al. “Rethinking virtual network embedding: substrate support for path splitting and migration”. In: *ACM SIGCOMM Computer Communication Review* 38.2 (2008), pp. 17–29.

[77] Ilhem Fajjari et al. “Adaptive-VNE: A flexible resource allocation for virtual network embedding algorithm”. In: *Proc. of IEEE GLOBECOM*. Anaheim, CA, 2012.

[78] Ying Yuan et al. “A Novel Algorithm for Embedding Dynamic Virtual Network Request”. In: *Proc. of IEEE ICISCE*. Shanghai, 2015.

[79] Zhongbao Zhang et al. “Energy aware virtual network embedding with dynamic demands: Online and offline”. In: *Computer Networks* 93 (2015), pp. 448–459.

[80] Lan Li et al. “An efficient Virtual Network embedding algorithm based on subgraph”. In: *Proc. of IEEE WPMC*. Sydney, NSW, 2014.

[81] Nicola Carapellese, Massimo Tornatore, and Achille Pattavina. “Placement of baseband units (BBUs) over fixed/mobile converged multi-stage WDM-PONs”. In: *Proc. of IEEE ONDM*. Brest, France, 2013.

[82] N. Carapellesse, Massimo Tornatore, and Achille Pattavina. “Energy-efficient baseband unit placement in a fixed/mobile converged WDM aggregation network”. In: *IEEE Journal on Selected Areas in Communications* 32.8 (2014), pp. 1542–1551.

[83] Bharat JR Sahu et al. “Energy Efficient BBU Allocation for Green C-RAN”. In: *IEEE Communications Letters* (2017).

[84] Xinbo Wang et al. “Green virtual base station in optical-access-enabled cloud-RAN”. In: *Proc. of ICC*. London, UK, 2015.

[85] M. Y. Lyazidi, N. Aitsaadi, and R. Langar. “Dynamic resource allocation for Cloud-RAN in LTE with real-time BBU/RRH assignment”. In: *Proc. of IEEE ICC*. Kuala Lumpur, Malaysia, 2016.

[86] Bo-Syuan Huang, Yi-Han Chiang, and Wanjiun Liao. “Remote radio head (RRH) deployment in flexible C-RAN under limited fronthaul capacity”. In: *Proc. of IEEE ICC*. Paris, France, 2017.

[87] Dalin Zhu and Ming Lei. “Traffic and interference-aware dynamic BBU-RRU mapping in C-RAN TDD with cross-subframe coordinated scheduling/beamforming”. In: *Proc. of IEEE ICC*. Budapest, Hungary, 2013.

[88] Shinobu Namba, Takayuki Warabino, and Shoji Kaneko. “BBU-RRH switching schemes for centralized RAN”. In: *Proc. of IEEE CHINACOM*. Kunming, China, 2012.

[89] Shinobu Namba et al. “Colony-RAN architecture for future cellular network”. In: *Proc. of IEEE FutureNetw*. Berlin, Germany, 2012.

[90] A Asensio et al. “Study of the centralization level of optical network-supported cloud RAN”. In: *Proc. of ONDM*. Cartagena, Spain, 2016.

[91] Yuh-Shyan Chen, Wen-Lin Chiang, and Min-Chun Shih. “A Dynamic BBU–RRH Mapping Scheme Using Borrow-and-Lend Approach in Cloud Radio Access Networks”. In: *IEEE Systems Journal* (2017).

[92] Henrik Holm et al. “Optimal assignment of cells in C-RAN deployments with multiple BBU pools”. In: *Proc. of EuCNC*. Paris, France, 2015.

[93] Shari Sofia Lisi et al. “Cost-effective migration towards C-RAN with optimal fronthaul design”. In: *Proc. of ICC*. Paris, France, 2017.

[94] Francesco Musumeci et al. “Optimal BBU placement for 5G C-RAN deployment over WDM aggregation networks”. In: *Journal of Lightwave Technology* 34.8 (2016), pp. 1963–1970.

[95] K. Samdanis and X. Costa-Perez and V. Sciancalepore. “From network sharing to multi-tenancy: The 5G network slice broker”. In: *IEEE Communications Magazine* 54.7 (2016), pp. 32–39.

[96] X. Zhou and R. Li and T. Chen and H. Zhang. “Network slicing as a service: enabling enterprises’ own software-defined cellular networks”. In: *IEEE Communications Magazine* 54.7 (2016), pp. 146–153.

[97] M. Richart and J. Baliosian and J. Serrat and J. L. Gorricho. “Resource Slicing in Virtual Wireless Networks: A Survey”. In: *IEEE Transactions on Network and Service Management* 13.3 (2016), pp. 462–476.

[98] I. da Silva and G. Mildh and A. Kaloxylos and P. Spapis and E. Buracchini and A. Trogolo and G. Zimmermann and N. Bayer. “Impact of network slicing on 5G Radio Access Networks”. In: *Proc. of EuCNC*. Athens, Greece, 2016.

[99] Uwe Dötsch et al. “Quantitative analysis of split base station processing and determination of advantageous architectures for LTE”. In: *Bell Labs Technical Journal* 18.1 (2013), pp. 105–128.

[100] *nFAPI and FAPI specifications*. Tech. rep. Small Cell Forum, 2017.

[101] *5G; NG-RAN; Architecture description*. Tech. rep. 3GPP TS 38.401 version 15.3.0 Release 15, 2018.

[102] *xRAN*. [Online]. Available: http://www.xran.org/.

[103] Dario Sabella et al. “RAN as a service: Challenges of designing a flexible RAN architecture in a cloud-based heterogeneous mobile network.” In: *Future Network & Mobile Summit* 2013 (2013).

[104] Peter Rost et al. “Cloud technologies for flexible 5G radio access networks”. In: *IEEE Communications Magazine* 52.5 (2014), pp. 68–76.

[105] Dirk Wubben et al. “Benefits and impact of cloud computing on 5G signal processing: Flexible centralization through cloud-RAN”. In: *Signal Processing Magazine, IEEE* 31.6 (2014), pp. 35–44.

[106] Jens Bartelt et al. “Fronthaul and backhaul requirements of flexibly centralized radio access networks”. In: *Wireless Communications, IEEE* 22.5 (2015), pp. 105–111.

[107] Carlos J Bernardos et al. “Challenges of designing jointly the backhaul and radio access network in a cloud-based mobile network”. In: *Proc. of IEEE FutureNetworkSummit*. Lisbon, Portugal, 2013.

[108] Jingchu Liu et al. “Graph-based framework for flexible baseband function splitting and placement in C-RAN”. In: *Proc. of IEEE ICC*. London, United Kingdom, 2015.

[109] Iordanis Koutsopoulos. “Optimal functional split selection and scheduling policies in 5G Radio Access Networks”. In: *Proc. of IEEE ICC Workshop*. Paris, France, 2017.

[110] Paul Arnold et al. “5G radio access network architecture based on flexible functional control/user plane splits”. In: *Proc. of EuCNC*. Oulu, Finland, 2017.

[111] Aleksandra Checko et al. “Evaluating C-RAN fronthaul functional splits in terms of network level energy and cost savings”. In: *Journal of Communications and Networks* 18.2 (2016), pp. 162–172.

[112] Andres Garcia-Saavedra et al. “WizHaul: On the Centralization Degree of Cloud RAN Next Generation Fronthaul”. In: *IEEE Transactions on Mobile Computing* (2018).

[113] Andres Garcia-Saavedra et al. “Joint optimization of edge computing architectures and radio access networks”. In: *IEEE Journal on Selected Areas in Communications* 36.11 (2018), pp. 2433–2443.

[114] *Free Space Optics Technology*. URL: http://www.cablefree.net/wirelesstechnology/free-space-optics/.

[115] Theodore S Rappaport et al. “Millimeter wave mobile communications for 5G cellular: It will work!” In: *IEEE access* 1.1 (2013), pp. 335–349.

[116] Ping-Heng Kuo and Alain Mourad. “Millimeter wave for 5G mobile fronthaul and backhaul”. In: *Proc. of IEEE EuCNC*. Oulu, Finland, 2017.

[117] Peng Wang et al. “Multi-gigabit millimeter wave wireless communications for 5G: From fixed access to cellular networks”. In: *IEEE Communications Magazine* 53.1 (2015), pp. 168–178.

[118] Dawit Hadush Hailu et al. “Mobile fronthaul transport options in C-RAN and emerging research directions: A comprehensive study”. In: *Optical Switching and Networking* (2018).

[119] Rami Al-obaidi et al. “Optimizing Cloud-RAN deployments in real-life scenarios using Microwave Radio”. In: *Proc. of EuCNC*. Paris, France, 2015.

[120] Andreas Maeder et al. “Towards a flexible functional split for Cloud-RAN networks”. In: *Proc. of EuCNC*. Bologna, Italy, 2014.

[121] M. Jaber, D. Owens, M. A. Imran, R. Tafazolli and A. Tukmanov. “A joint backhaul and RAN perspective on the benefits of centralised RAN functions”. In: *Proc. of IEEE ICC*. Kuala Lumpur, Malaysia, 2016.

[122] Bin Guo et al. “LTE/LTE-A signal compression on the CPRI interface”. In: *Bell Labs Technical Journal* 18.2 (2013), pp. 117–133.

[123] *Radio over Ethernet (EoF)*. Tech. rep. IEEE NGFI P1914.3, 2018.

[124] *eCPRI; Interface Specification*. Tech. rep. eCPRI Specification V1.0, 2017.

[125] *Control, User and Synchronization Plane Specification*. Tech. rep. XRAN-FH.CUS.0-v02.00, 2018.

[126] *Presision Time Protocol (PTP)*. Tech. rep. IEEE 1588v2, 2008.

[127] *Time-Sensitive Networking for Fronthaul (IEEE 802.1CM)*. Tech. rep. IEEE 802.1CM.

[128] C. Y. Chang and R. Schiavi and N. Nikaein and T. Spyropoulos and C. Bonnet. “Impact of packetization and functional split on C-RAN fronthaul performance”. In: *Proc. of IEEE ICC*. Kuala Lumpur, Malaysia, 2016.

[129] Chia-Yu Chang et al. “FlexCRAN: A flexible functional split framework over ethernet fronthaul in Cloud-RAN”. In: *Proc. of IEEE ICC*. Paris, France, 2017.

[130] Nikos Makris et al. “Experimental evaluation of functional splits for 5G cloud-RANs”. In: *Proc. of IEEE ICC*. Paris, France, 2017.

[131] Aleksandra Checko, Henrik Holm, and Henrik Christiansen. “Optimizing small cell deployment by the use of C-RANs”. In: *Proc. of European Wireless*. Barcelona, Spain, 2014.

[132] T. S. Rappaport and S. Sun and R. Mayzus and H. Zhao and Y. Azar and K. Wang and G. N. Wong and J. K. Schulz and M. Samimi and F. Gutierrez. “Millimeter Wave Mobile Communications for 5G Cellular: It Will Work!” In: *IEEE Access* 1 (2013), pp. 335–349.

[133] Z. Pi and F. Khan. “An introduction to millimeter-wave mobile broadband systems”. In: *IEEE Communications Magazine* 49.6 (2011), pp. 101–107.

[134] A. Ghosh et al. “Millimeter-Wave Enhanced Local Area Systems: A High-Data-Rate Approach for Future Wireless Networks”. In: *IEEE Journal on Selected Areas in Communications* 32.6 (2014), pp. 1152–1163.

[135] Sundeep Rangan, Theodore S Rappaport, and Elza Erkip. “Millimeter-wave cellular wireless networks: Potentials and challenges”. In: *Proceedings of the IEEE* 102.3 (2014), pp. 366–385.

[136] Yasir Zaki et al. “LTE wireless virtualization and spectrum management”. In: *Proc. of IEEE WCNC*. Budapest, Hungary, 2010.

[137] Yasir Zaki et al. “LTE mobile network virtualization”. In: *Mobile Networks and Applications* 16.4 (2011), pp. 424–432.

[138] Kokku, Ravi and Mahindra, Rajesh and Zhang, Honghai and Rangarajan, Sampath. “NVS: a substrate for virtualizing wireless resources in cellular networks”. In: *IEEE/ACM Transactions on Networking* 20.5 (2012), pp. 1333–1346.

[139] Ashraf Awadelkarim Widaa Ahmed et al. “Techno-economics of Green Mobile Networks Considering Backhauling”. In: *Proc. of European Wireless*. Barcelona, Spain, 2014.

[140] Haonan Hu, Jialai Weng, and Jie Zhang. “Coverage Performance Analysis of FeICIC Low-Power Subframes”. In: *IEEE Transactions on Wireless Communications* 15.8 (2016), pp. 5603–5614.

[141] Oscar D Ramos-Cantor et al. “Centralized Coordinated Scheduling in LTE-Advanced Networks”. In: *arXiv preprint arXiv:1702.03890* (2017).