Performance Analysis of Software Defined Networks in LTE-A

Miguel Ângelo Godinho de Sá

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisor: Prof. Luís Manuel de Jesus Sousa Correia

Examination Committee

Chairperson: Prof. José Eduardo Charters Ribeiro da Cunha Sanguino

Supervisor: Prof. Luís Manuel de Jesus Sousa Correia

Member of Committee: Prof. António José Castelo Branco Rodrigues

Member of Committee: Eng. Ricardo Morgado

July 2015

To the ones I love

Acknowledgements

First of all, I would like to thank Prof. Luís M. Correia for the opportunity to write this thesis under his guidance. I will never forget our meetings and his precious advices, which arguably improved my academic performance and shaped my attitude to always aim for the highest degree of excellence. I would also like to thank him for the valuable experience of doing this work in collaboration with Ericsson and being part of GROW, where I felt surrounded by some of the brightest minds working in mobile communications.

To all GROW members, especially my master's thesis colleagues and friends Ana Cláudia Castro, Andrea Marotta, Carlos Martins, João Pires, Ricardo Gameiro, and to Lúcio Ferreira for all the information he kindly shared with me.

To Ericsson Portugal for allowing me to develop a work closely connected to the industry, and in particular to Eng. Ricardo Morgado for the time and effort he has put into following the progress of my work, helping me with his critics and suggestions.

To my colleagues and friends that accompanied me throughout my journey at IST through the good and bad moments: Abel Ribeiro, Caio Rodrigues, Filipe Teixeira, Jorge Palma, Luís Almeida, Miguel Rodrigues and Ricardo Silva.

To my brothers André Mendes and João Silva and my sisters Lisa Silva and Patrícia Chilra, my deep and sincere gratitude for all their support and friendship.

To my amazing family, a big thank you for their support and affection, especially my mother, Maria Luísa Sá, and my father, Ângelo Sá, for always being terrific role models and for their endless love and encouragement.

Last, but not least, to the love of my life, Ana, for her unconditional love and support, for always inspiring me, for believing in me and for being patient and kind when I needed the most.

Abstract

The objective of this thesis was to analyse how Software Defined Networking technology can improve network performance in LTE-A, by taking advantage of the separation between the control and data planes. This work consists of a study concerning the impact of C-RAN and virtualisation techniques in an operator’s network in a near future, namely in terms of the necessary number of storage and processing nodes and the links in between, taking into account increasingly demanding latency and capacity constraints. A model was implemented that takes as an input the positioning of cell sites available countrywide and computes the number and a possible placement of processing nodes for a given maximum latency criterion – an estimate of the number of required blade servers in each node is also computed. Finally, an estimate for the end-to-end delay from Portugal to other regions in the world is made, using typical values for the various delay contributions. Results show that between 56 and 273 BBU (Baseband processing Units) pools are required to cover the whole country, depending on the fronthaul delay restriction. It is also verified that a single core node can support all BBU pools. An inverse proportionality relation between the number of blade servers and their corresponding capacity has been ascertained. A difference in delay of more than 250 ms was found for a server located in Japan in comparison with one in Germany.

Keywords

LTE-Advanced, Virtualisation, SDN, C-RAN, NSC, Delay.

Resumo

O objetivo desta tese foi o de analisar de que forma a tecnologia SDN pode melhorar o desempenho da rede LTE-A, tirando partido da separação entre os planos de controlo e de dados. Este trabalho consiste no estudo do impacto que o uso de técnicas de virtualização e C-RAN podem vir a ter no futuro na rede de um operador, ao nível do número de nós necessários para armazenamento e processamento de dados e das ligações entre estes, tendo em conta restrições de latência e capacidade cada vez mais exigentes. Para a concretização deste trabalho foi implementado um modelo que toma como entrada a localização das estações base de um operador presentes em todo o território de Portugal Continental, e, tendo em conta um critério de latência máxima, calcula o número e possível posicionamento de nós de processamento. É também feita uma estimativa do número de máquinas necessárias em cada nó. Finalmente, é efetuada uma estimativa do atraso extremo-a-extremo de Portugal para outras regiões continentais, utilizando valores típicos para as diversas contribuições. Os resultados obtidos mostram que são necessárias entre 56 e 273 servidor de BBU (unidades de processamento de banda base) para cobrir todo o país, dependendo da restrição de atraso na ligação a nós rádio. Foi também possível verificar que um nó da rede central pode ser suficiente para cobrir o país. Verifica-se uma relação de proporcionalidade inversa entre número de máquinas necessárias e a capacidade das mesmas. Por fim, verificou-se uma diferença de mais de 250 ms entre o atraso para um servidor no Japão e um outro na Alemanha.

Palavras-chave

LTE-Advanced, Virtualização, SDN, C-RAN, NSC, Atraso

Table of Contents

List of Figures

List of Tables

List of Acronyms

3GPP	3rd Generation Partnership Project
4G	Fourth-Generation
AD/DA	Analogue to Digital / Digital to Analogue
AMC	Adaptive Modulation and Coding
API	Application Programming Interface
AS	Autonomous System
BBU	Base Band Unit
BS	Base Station
CA	Carrier Aggregation
CAPEX	Capital Expenditure
CN	Core Network
CO	Central Office
CoMP	Coordinated Multipoint Transmission/Reception
CP	Control Plane
CPRI	Common Public Radio Interface
C-RAN	Cloud Radio Access Network
DL	Downlink
D-RoF	Digital Radio over Fibre
eNodeB	Evolved Node B
EPC	Evolved Packet Core
EPS	Evolved Packet System
ETSI	European Telecommunications Standards Institute
E-UTRA	Evolved Universal Terrestrial Radio Access
E-UTRAN	Evolved Universal Terrestrial Radio Access Network
FDD	Frequency Division Duplex
GBR	Guaranteed Bit Rate
HARQ	Hybrid Automatic Repeat Request
HSS	Home Subscription Service
IMS	IP Multimedia Sub-System
IP	Internet Protocol
ISG	Industry Specifications Group
LTE	Long Term Evolution
LTE-A	LTE-Advanced
M2M	Machine to Machine

MBR	Maximum Bit Rate
MBMS	Multimedia Broadcast and Multicast Services
MCN	Mobile Cloud Networking
MIMO	Multiple Input Multiple Output
MM	Mobility Management
MME	Mobility Management Entity
MPLS	Multi-Protocol Label Switching
MVNO	Mobile Virtual Network Operator
NFV	Network Functions Virtualisation
NFVI	Network Functions Virtualisation Infrastructure
NIST	Nation Institute of Standards and Technology
NOS	Network Operating System
NSC	Network Service Chaining
NV	Network Virtualisation
OBSAI	Open Base Station Architecture Initiative
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
ONF	Open Networking Foundation
OPEX	Operational Expenditure
OWD	One-Way Delay
PBCH	Physical Broadcast Channel
PCC	Policy and Charging Control
PCFICH	Physical Control Format Indicator Channel
PCRF	Policy and Charging Resource Function
PDCCH	Physical Downlink Control Channel
PDSCH	Physical Downlink Shared Channel
PHICH	Physical HARQ Indicator Channel
PMCH	Physical Multicast Channel
PRACH	Physical Random Access Channel
PUCCH	Physical Uplink Control Channel
PUSCH	Physical Uplink Shared Channel
P-GW	Packet Data Network Gateway
QCI	QoS Class Identifier
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RB	Resource Block
RF	Radio Frequency
RLC	Radio Link Control
RRH	Remote Radio Head

RRM	Radio Resource Management
RTT	Round Trip Time
SDWN	Software Defined Wireless Network
SDN	Software Defined Networking
SNR	Signal to Noise Ratio
SON	Self-Optimising Networks
S-GW	Serving Gateway
SAE	System Architecture Evolution
TCP	Transmission Control Protocol
TDD	Time Division Duplex
TE	Terminal Equipment
UE	User Equipment
UL	Uplink
UMTS	Universal Mobile Telecommunications System
UP	User Plane
USIM	Universal Subscriber Identity Module
VBS	Virtual Base Station
VoIP	Voice over IP
VM	Virtual Machine
VNF	Virtual Network Function
vSwitch	Virtual Switch
WAN	Wide Area Network
WDM	Wavelength Division Multiplexing

List of Symbols

δ_{bb}	RTT in the Backbone
δ_{bh}	RTT in the Backhaul
$\delta_{BBUpool}$	Delay due to processing in the BBU
$\delta_{end-to-end}$	End-to-end RTT
$\delta_{end-to-end}^{NSC}$	End-to-end RTT with Network Service Chaining
δ_{eNodeB}	Delay in the eNodeB
δ_{EPC}	Delay in the EPC
δ_{fh}	RTT in the Fronthaul
$\delta_{fh,max}$	Maximum OWD in the Fronthaul
δ_{NSC}	Delay due to Network Service Chaining
δ_{OWD}	One-way delay
δ_{RAN}	RTT in the RAN
δ_{RRH}	Delay due to processing in the RRH
δ_{RTT}	Round trip time
δ_{server}	Delay in the server
$\sqrt{\overline{\varepsilon^2}}$	Root Mean Squared Error
μ_x	Average value of variable x
σ_x	Standard deviation of variable x
$\Sigma_{regression}$	Sum of squares of the regression
Σ_{μ}	Sum of squares about the mean
$C_{backhaul}$	Backhaul link capacity
$C_{BBUpool}$	BBU Pool capacity
C_{Server}	Capacity of a single Blade Server
$C_{CoreNode}$	Core Node capacity
$C_{fronthaul}$	Fronthaul link capacity
C_{Node}	Capacity of a network node (BBU Pool or Core Node)
d_{fh}	Length of a fronthaul link
$d_{fh,max}$	Maximum distance in the Fronthaul
n	Number of observations
$N_{bhlinks}$	Number of backhaul links
$N_{BBUPools}$	Number of BBU Pools
$N_{BladeServers}$	Number of blade servers

$N_{fhlinks}$	Total number of fronthaul links
$N_{mwlinks}$	Number of microwave links
$N_{Servers/BBUPool}$	Number of blade servers per BBU Pool
$N_{Servers/CoreNode}$	Number of blade servers per Core Node
$N_{Sites/BBUPool}$	Number of cell sites per BBU Pool
R^2	Correlation Coefficient
ν	Transmission speed in a link
y_i	Observation i
$\widehat{y_i}$	Estimated or predicted value of y_i

List of Software

Gliffy	Cloud-based diagramming software
Google Earth	Geographical information program
Matlab	Numerical computing environment
Microsoft Excel 2013	Spreadsheet application
Microsoft Word 2013	Word processor

Chapter 1

Introduction

This chapter presents an overview of the context in which this thesis fits, taking into account the current mobile communications scenario, as well as the motivations underpinning the present work, followed by a presentation of its structure.

1.1 Overview

Consumer demands in the mobile communications domain have changed significantly over the past years, shaping mobile operators priorities. Figure 1.1 evidences the present data traffic dominance over voice one, contrasting the stable trend of data traffic growth with the almost flat voice traffic evolution. Video is the largest and fastest growing segment of mobile data traffic – it is expected to grow around 13 times by 2019, according to [Eric14].

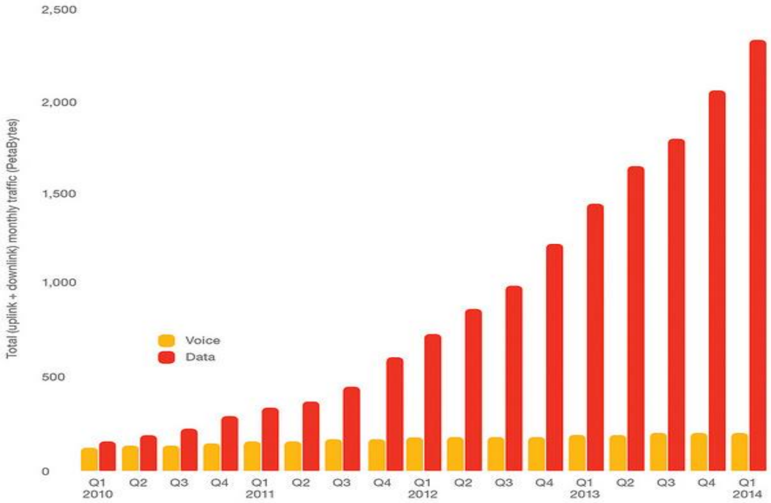

Figure 1.1. Global total traffic in mobile networks, 2010-2014 (extracted from [Eric14]).

This huge increase in network traffic, caused not only by the increase in mobile subscribers but also by the massive adoption of smartphones, forces operators to expand and upgrade their networks. In fact, smartphones support a panoply of applications with different requirements that need to be met, such as web browsing, audio and video streaming, social networks, file sharing, mobile TV or multimedia online gaming, causing frequent traffic peaks that put strain on the network infrastructure.

The definition of the targets for 3rd Generation Partnership Project (3GPP) Long Term Evolution (LTE), often called the fourth-generation (4G), started in 2004. LTE's development was driven by a need for more wireless capacity, a need for lower cost wireless data delivery (higher efficiency), and competition from other wireless technologies. The first LTE release was Release 8, providing a high-data rate, low-latency and packet-optimised system, supporting theoretical peak data rates up to 300 Mbps in Downlink (DL) and 75 Mbps in Uplink (UL). LTE-A (LTE-Advanced) is standardised in Release 10, which specifies data rates up to 3 Gbps in DL and 1 Gbps in UL. Figure 1.2 shows the radio access milestones defined by 3GPP over the years, until the present time.

The major focus for all 3GPP Releases is to make the systems backward and forward compatible, in a way that the operation of user equipment is un-interrupted, meaning that for example an LTE-A terminal can work in an LTE cell and an LTE terminal works in an LTE-A cell.

A significant proportion of 3GPP's recent work has been focused on channel aggregation, to meet the growing demand for data transmission. Other priorities in the radio area include topics related to higher data rates and increased capacity. 3GPP is planning to address energy saving, cost efficiency (including the use of Self-Optimising Networks (SON)), support for diverse application and traffic types, and backhaul enhancements [3GPP14b].

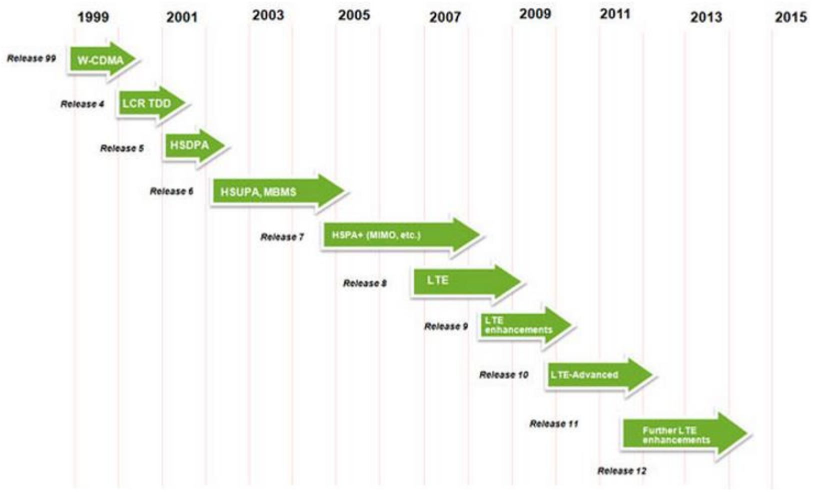

Figure 1.2. Radio Access Milestones (extracted from [3GPP14b]).

Given the critical challenges imposed by the mobile data traffic growth trend and the ever increasing demand for higher data rates, network operators must find solutions to overcome them. With the success of cloud technology in the enterprise realm, the telecom industry is now looking to the cloud to reap the same benefits – economies of scale, cost effectiveness, scalability, lower Capital Expenditure (CAPEX) and Operational Expenditure (OPEX). Operators want to embrace cloud technologies in their central offices and network functions to achieve these benefits.

In this context, projects like Mobile Cloud Networking (MCN, [Kara14]) are being developed, supported in technologies such as Software Defined Networking (SDN), which appears as a promising technology to transform the way networks are managed. Figure 1.3 discloses the philosophy behind MCN, which the present thesis also shares.

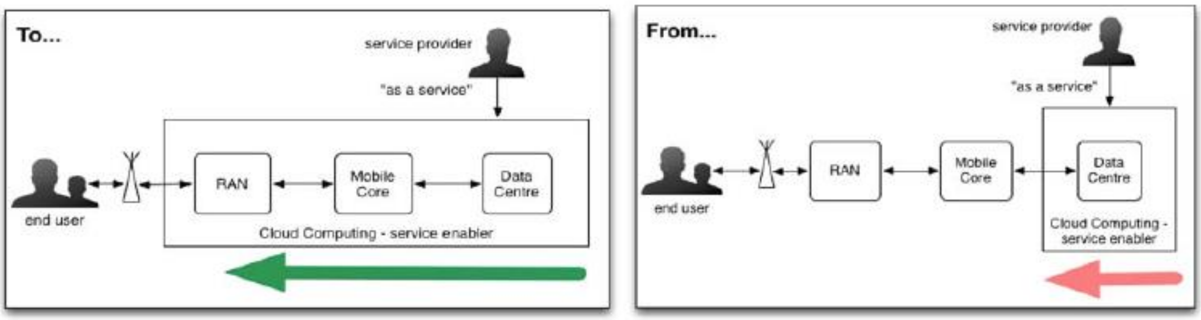

Figure 1.3. Vision of Mobile Cloud Networking (extracted from [Kara14]).

MCN is focusing on the integration of the cloud computing and Network Functions Virtualisation (NFV) concepts into cellular networks; its basis relies on two principles:

- Cloud Computing, as defined by the America Nation Institute of Standards and Technology (NIST), which states that a cloud computing service must display the characteristics of resource pooling, broad network access, rapid elasticity and measured service (Pay-As-You-Go).
- A service-oriented architecture.

Regarding the MCN approach, it is also important to note that infrastructure sharing, enabled by virtualisation, is seen as one of the most fundamental enablers of cloud computing, existing in the mobile Telco industry ever since the emergence of the Mobile Virtual Network Operator (MVNO) [CoCr11]. MVNOs are operators that can adopt different models: in one extreme (light MVNO), the MVNO outsources all the operation, including the marketing and selling force; in the other extreme (full MVNO), the MVNO might operate almost all parts of the cellular network, except the radio spectrum license.

According to a recent survey conducted among operators [InRe14], cost reductions and new revenue models are the main drivers for adopting SDN and NFV. However, there are still several challenges to face before these technologies become widespread. In effect, Figure 1.4 shows that when asked if or when they will introduce SDN into the mobile backhaul network, 29% of respondents say that they are deploying or plan to deploy SDN at some point, while the majority (63%) are evaluating it as a possibility with no set timeframe.

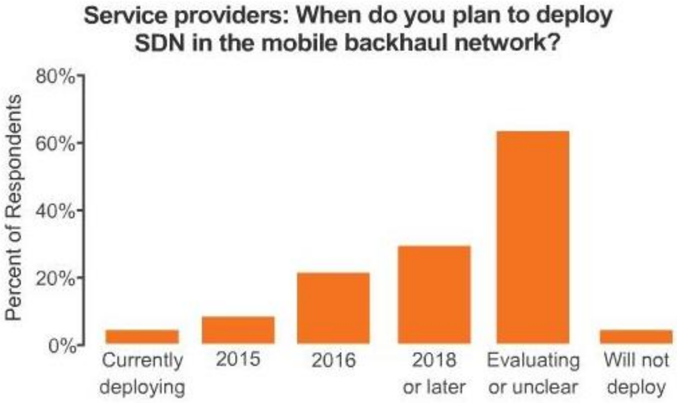

Figure 1.4. Overview of SDN deployment in the mobile backhaul (extracted from [InRe14]).

This reveals that the industry is widely aware of the potential of SDN and NFV, but there is a need for further studies before proceeding to a greater deployment. This thesis is a step further in that direction.

1.2 Motivation and Contents

The nowadays continuously increasing number of mobile users, devices and new mobile applications has resulted in the mobile data traffic growing at an unprecedented rate, which alternatively has a significant impact on the complexity of processes required to provide reliable cellular networks. Over the last few years, mobile operators are struggling to cope with these challenges. Still they rely on highly centralised and custom hardware components that are not designed with elasticity in mind, leading to a weak optimisation of resources during non-peak hours and frequent overloads during peak hours.

From its genesis, LTE has taken an all-IP perspective, i.e., the connection in between all nodes is supported on IP (Internet Protocol), encouraging the use of more efficient techniques to manage the network. The advent of SDN technologies will enable this more efficient management, providing new programmability and flexibility properties, by taking advantage of the separation between control and data planes. Essential features of SDN are implemented in LTE-A by using the OpenFlow open standard.

Under this circumstances, it is crucial to compare the performance of SDN-enabled networks with existing solutions. The current thesis addresses this aspect, establishing a model for network performance comparison of SDN in LTE-A, considering different user data service profiles. The main output of this thesis is the simulator used to implement the referred model, allowing an evaluation of the enhancements introduced by an SDN control strategy and of the fulfilment of delay and throughput requirements for different services. The work was done in collaboration with Ericsson, which had the important role of providing assistance on several technical details and insights into the technologies. Furthermore, the countrywide network (cell site locations) taken as application example comes from Vodafone Portugal.

The present thesis is composed of 5 chapters and 2 annexes, as depicted in Figure 1.5.

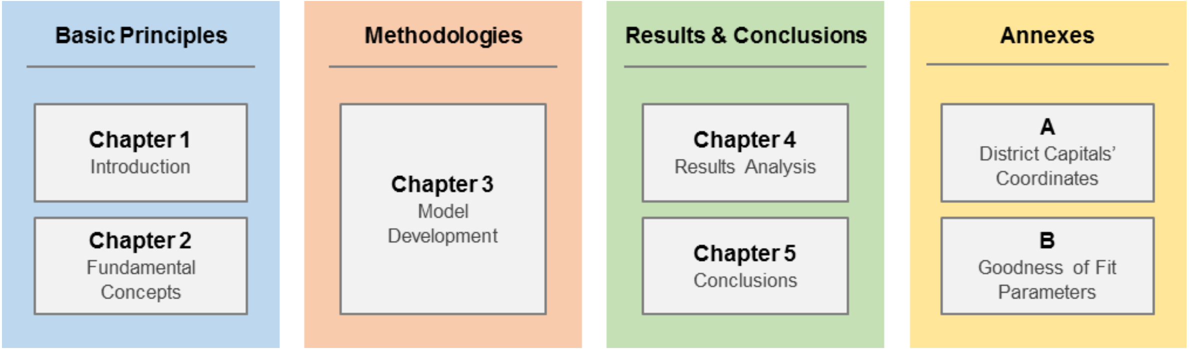

Figure 1.5. Thesis Structure.

Chapter 2 provides background knowledge on the fundamental concepts of LTE, SDN and Virtualisation. It starts with a brief analysis of LTE's network architecture and radio interface, followed by a discussion of SDN and its OpenFlow realisation, and an introduction to NFV and Cloud-RAN. Then, Quality of Service in LTE is examined. Finally, the state of the art is presented, containing the latest work developed related to this thesis.

Chapter 3 concerns the model developed for the purpose of this thesis, explaining its features and limitations with the support of algorithm flowcharts. First, the delay and capacity metrics are presented and detailed, for both network nodes and links, along with a number of assumptions that simplify the analysis. Next, a high level perspective of the model is given, and finally a detailed examination of the nuts and bolts of the developed script is done.

Chapter 4 contains all the results obtained in the context of this work, along with the respective discussion and analysis. First, the scenario is presented. Then, a section dedicated to RAN analysis and another to Core analysis report the relevant findings for these network parts, examining in first place the impact of the variation of the maximum delay and in second place the impact of the variation of blade server capacity. The last section of this chapter is mostly dedicated to evaluating the various components of the total end-to-end delay, particularly the ones due to backbone latency and Network Service Chaining (NSC).

Chapter 5 summarises the main results and conclusions of this thesis, in a way that a general understanding of the most important findings is given. Lastly, this chapter points out future research directions that can be taken in order to improve the analysis done with this work.

At the end, a group of annexes is presented containing additional information. Annex A provides the geographical coordinates of Portuguese district capitals used in the model implementation. Annex B presents the goodness of fit parameters of the different mathematical models used for data fitting.

Chapter 2

Fundamental Concepts

This chapter provides firstly a background on the fundamental concepts of LTE, SDN and Virtualisation, including LTE's network architecture and radio interface, a synopsis of SDN and its OpenFlow realisation, and an introduction to NFV and Cloud-RAN. Then, follows a brief discussion of Quality of Service in LTE. The last section of this chapter is dedicated to an analysis of the state of the art.

2.1 LTE Aspects

2.1.1 Network Architecture

In this section, an overview of LTE's network architecture is given, based on [HoTo11] and [SeTB11].

Parallel to the work on LTE's radio-access technology in 3GPP, the overall system architecture of both the Core Network (CN) and the Radio-Access Network (RAN) was revisited, including the split of functionalities in between them. This work, known as the System Architecture Evolution (SAE), resulted in a flat RAN architecture and a new CN one, referred to as the Evolved Packet Core (EPC). LTE's architecture is divided into four main domains: Services, EPC, Evolved UTRAN (E-UTRAN), and User Equipment (UE), as presented in Figure 2.1. The UE, the E-UTRAN and the EPC constitute the Evolved Packet System (EPS), representing the IP Connectivity Layer.

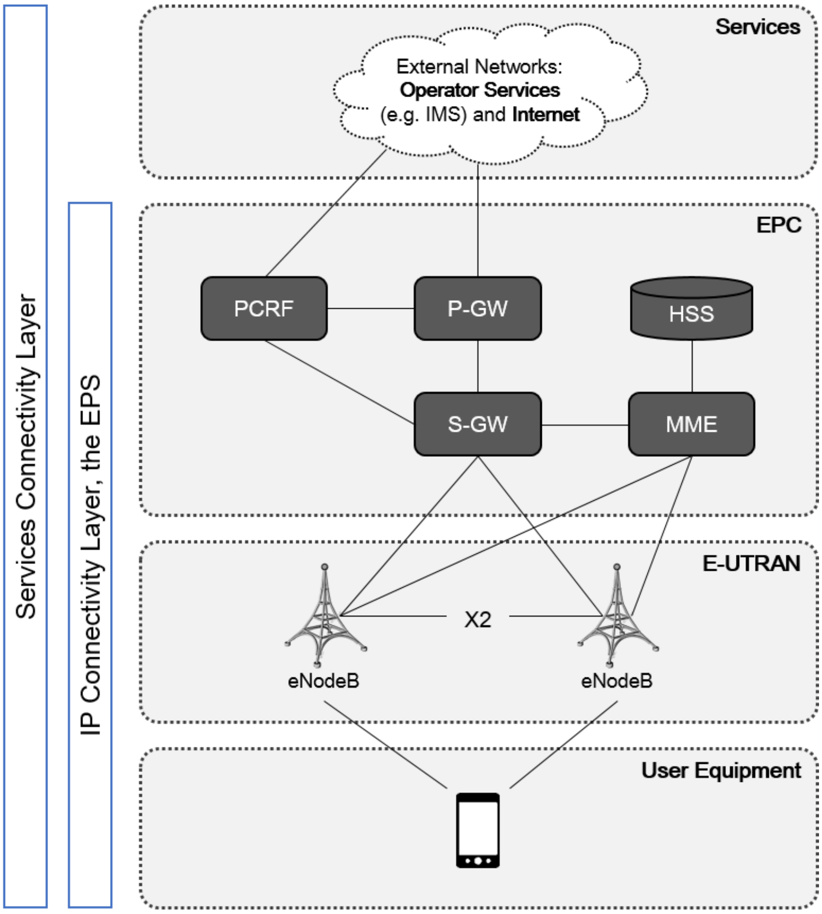

Figure 2.1. System architecture for an E-UTRAN only network (adapted from [HoTo11]).

Concerning the Services domain, it should be noted that it includes not only services provided by the mobile network operator, such as IP Multimedia Sub-System (IMS), but also many others available through the Internet.

When designing the evolution of the 3G system, the 3GPP community decided to use IP as the protocol to transport all services. Therefore, it was agreed that the EPC, unlike core networks of previous 3GPP architectures, would not have a circuit-switched domain. The EPC is responsible for the overall control of the UE and bearers' establishment, and is constituted by the following fundamental nodes:

- The Mobility Management Entity (MME), the main Control Plane (CP) element in the EPC, responsible for processing the signalling between the UE and the EPC. It supports functions related to connection management. Furthermore, the MME handles the inter-working with other networks (e.g. legacy networks).
- The Serving Gateway (S-GW), responsible for User Plane (UP) tunnel management and switching. This node acts as a local mobility anchor during mobility between evolved Nodes B (eNodeBs). Moreover, the S-GW collects information and statistics necessary for charging.
- The Packet Data Network Gateway (PDN Gateway, P-GW) connects the EPC to external packet data networks. It deals with the allocation of the IP address for each terminal, as well as quality-of-service (QoS) enforcement and flow-based charging.
- The Policy and Charging Rules Function (PCRF), responsible for Policy and Charging Control (PCC), deciding on the QoS associated with each service.
- The Home Subscription Service (HSS), which is a database server that records the location and all permanent data of the user.

The E-UTRAN is constituted by a mesh of eNodeBs connected among themselves through the X2 interface, and provides connectivity to the EPC. The eNodeBs handle all functions related with radio access, namely:

- Radio Resource Management (RRM), which controls the usage of the radio interface by allocating resources according to requests, performing UL/DL scheduling in accordance with the required QoS. It continuously monitors the resources availability.
- IP header compression, allowing an efficient use of the radio interface.
- Ciphering of user data streams, for security purposes.
- Mobility Management (MM), which performs handover decisions based on the analysis of radio signal level measurements executed both at the UE and at the eNodeB, dealing with the exchange of handover signalling between eNodeBs and the MME.

Lastly, the UE represents the end user's equipment, including both the Universal Subscriber Identity Module (USIM), used to authenticate and identity the user, and the Terminal Equipment (TE). It communicates with the network in order to establish, maintain and remove its connection.

2.1.2 Radio Interface

This section addresses the main features of LTE's Radio Interface based mostly on [HoTo11], including frequency bands, multiple access techniques, resource allocation, modulation schemes, data and control channels and Multiple-Input Multiple-Output (MIMO).

According to 3GPP Release 12 [3GPP14d], Evolved-Universal Terrestrial Radio Access (E-UTRA) is designed to operate in 44 frequency bands, distributed by the two duplex schemes used in LTE: Frequency Division Duplex (FDD) and Time Division Duplex (TDD). Thereby, 32 frequency bands are assigned to FDD (paired bands) and 12 to TDD (unpaired bands). In Europe, FDD is the widely adopted duplex mode, and the most relevant bands correspond to 800 MHz, 900 MHz, 1 800 MHz and 2.6 GHz. Portugal's communications sector regulator (ANACOM), following a trend among other European countries, issued unified titles of right of use of frequencies for the provision of electronic communications services, by means of an auction. The 450 MHz, 800 MHz, 900 MHz, 1 800 MHz, 2.1 GHz and 2.6 GHz bands were auctioned, from which the 800 MHz, 1 800MHz and 2.6 GHz bands were adopted for LTE [ANAC12], by Portuguese operators.

In what concerns multiple access techniques, LTE uses Orthogonal Frequency Division Multiple Access (OFDMA) in DL and Single Carrier Frequency Division Multiple Access (SC-FDMA) in UL, both with Cyclic Prefix. While OFDMA aims to minimise receiver's complexity and enables frequency domain scheduling with resource allocation flexibility, SC-FDMA aims to optimise the range and power consumption of the UE.

LTE-A benefits from frequency domain scheduling based on Resource Blocks (RBs). An RB consists of a group of 12 sub-carriers, which occupy 180 kHz in the frequency domain, and 6 or 7 OFDM symbols in the time domain, depending on if the extended Cyclic Prefix or the normal one is being used, respectively. The minimum resource unit is the Resource Element, which consists of one sub-carrier during one OFDM symbol, hence an RB has 72 or 84 resource elements per slot in the time domain. The channel bandwidth in LTE-A ranges from 1.4 MHz to 20 MHz, with the corresponding number of RBs ranging from 6 to 100. Figure 2.2 illustrates the resource allocation process in DL.

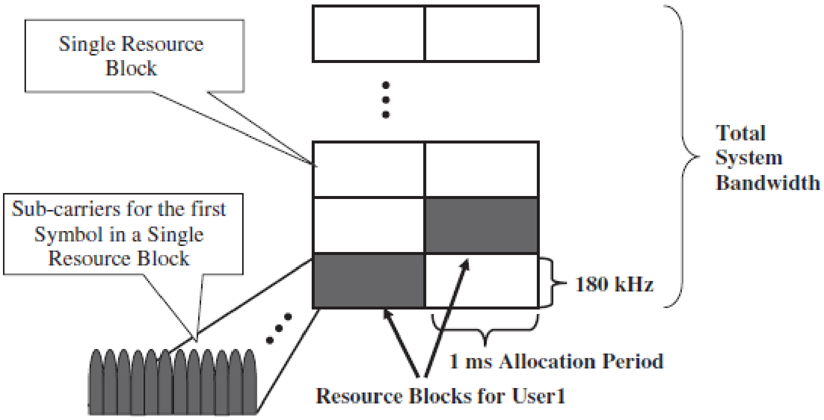
Single Resource Block
Sub-carriers for the first Symbol in a Single Resource Block
Total System Bandwidth
180 kHz
1 ms Allocation Period
Resource Blocks for User1

Figure 2.2. Resource allocation in OFDMA (extracted from [HoTo11]).

Furthermore, Carrier Aggregation (CA) can extend the channel bandwidth up to 100 MHz by aggregating up to 5 carriers with 20 MHz bandwidth. This aggregation may occur with carriers from the same band (Intra-band CA) or among different bands (Inter-band CA).

Resource allocation in UL is done similarly to DL, but RBs are allocated to each user consecutively in the frequency domain, Figure 2.3. The maximum bandwidth that can be allocated is 20 MHz, but there has to be some margin for the guard bands, the useful channel bandwidth being smaller.

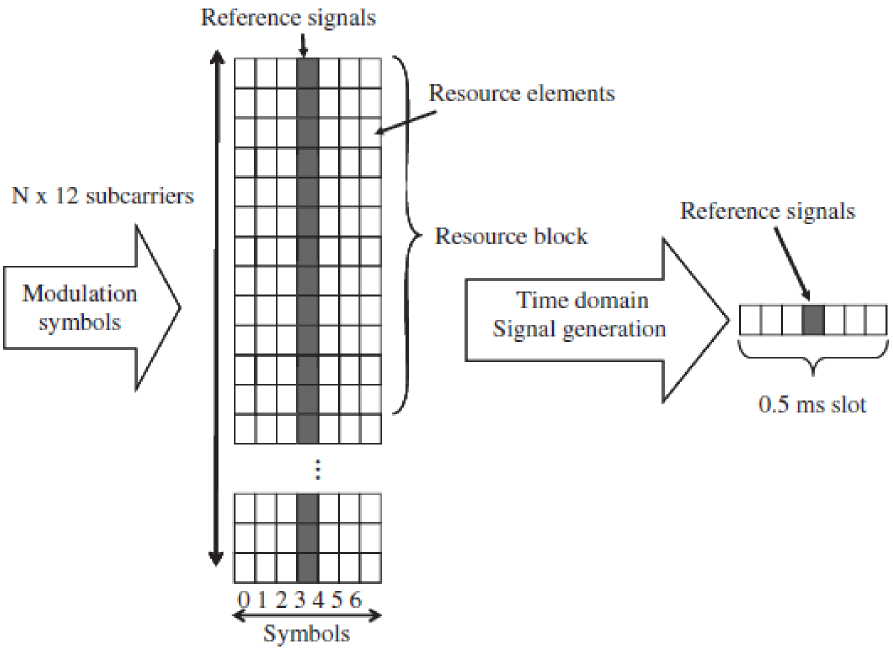

Figure 2.3. Resource allocation in SC-FDMA (extracted from [HoTo11]).

Regarding the modulation schemes, LTE uses QPSK, 16QAM and 64QAM, corresponding to 2, 4 and 6 bits per modulation symbol, respectively, as well as Adaptive Modulation and Coding (AMC).

User and system information are carried by physical channels. In LTE, there are 6 channels for DL and 3 for UL. In DL, the Physical Broadcast Channel (PBCH) carries information needed to access the system, such as cell's bandwidth. Control information is carried by the Physical Control Format Indicator Channel (PCFICH), the Physical Hybrid Automatic Repeat Request (ARQ) Indicator Channel (PHICH) and the Physical Downlink Control Channel (PDCCH). Broadcast system information, paging messages and all user data are carried on the Physical Downlink Shared Channel (PDSCH), while the Physical Multicast Channel (PMCH) carries data associated with Multimedia Broadcast and Multicast Services (MBMS) [SeTB11]. In UL, there is the Physical Random Access Channel (PRACH) to carry random access information, the Physical Uplink Shared Channel (PUSCH) for data carriage, and the Physical Uplink Control Channel (PUCCH) carrying control information, as the PDCCH does for DL.

Also, one of the most important features introduced with LTE is MIMO, which increases the peak data rate by a factor of 2, with a 2×2 antenna configuration, or 4 if a 4×4 one is used.

2.2 Software Defined Networking

SDN has emerged as a network architecture paradigm capable of supporting the dynamic nature of future network functions and applications, while lowering operation costs through the usage of simplified hardware, software, and management [Seze13]. It should be noted that SDN is not equal to network virtualisation – SDN is a mechanism that can be applied in Network Virtualisation (NV) or not [LiYu14].

This section seeks to provide a background on SDN, focusing on its principles and architecture, its differences and advantages compared to traditional networking, and the challenges related to its implementation, with special attention to the mobile networking domain of application.

The Open Networking Foundation (ONF) defines SDN as a network architecture where there is a split between control and data planes, network intelligence and state are logically centralised, and the underlying network infrastructure is abstracted from the applications [ONFo12]. SDN focuses on four key principles [Jars14]:

- Separation of control and data planes: externalisation of the CP from network devices to a separate entity - the controller – which possesses the ability to change the forwarding behaviour of network devices. This way, the control and data planes can evolve separately from each other. It is important to note that this separation imposes a business model adaptation in vendors, traditionally used to sell hardware with bundled control and data plane functionalities.
- Logically centralised control: the controller is a logically centralised entity, meaning it can consist of multiple physical or virtual instances, but behaves like a single component. The global network view kept by the controller enables it to adapt network policy in terms of routing and forwarding in a faster way than a system of traditional routers could. However, the referred centralisation raises scalability issues depending on the scenario and network size. So, an implementation of the SDN controller as a distributed system might be needed in some cases, to assure scalability.
- Open interfaces: a closed or proprietary interface limits component exchangeability and innovation, so it is crucial that interfaces remain open for SDN to reach its full flexibility potential.
- Programmability: It represents the ability to treat the network as a single programmable entity instead of a set of devices that have to be configured individually. By this principle, SDN can be treated as a suitable complement to network virtualisation, providing a simple control framework for managing virtual networks and specific flows within these. Of course that it is imperative to balance the trade-off between abstraction level (ease-of-use for programmers) and the respective overhead that must not introduce significant performance degradation.

In the literature one can find multiple block diagrams describing SDN's architecture. Figure 2.4 presents a generic SDN architecture based on [ONFo12] and [Jars14], capturing the most significant interfaces and components. This architecture consists of three layers:

- The Application Layer, consisting of applications that consume the SDN communications services. The set of possible applications is huge, ranging from messaging and M2M to real-time gaming or video streaming.
- The Control Layer, providing the logically centralised control functionality that commands the behaviour of the underlying infrastructure.
- The Infrastructure Layer, in this case represented by three Autonomous Systems (AS): a legacy access network at the user end, an SDN-based transit Wide Area Network (WAN), and an SDN-enabled data centre network (Cloud). So, this layer comprehends mainly devices that provide packet switching and forwarding.

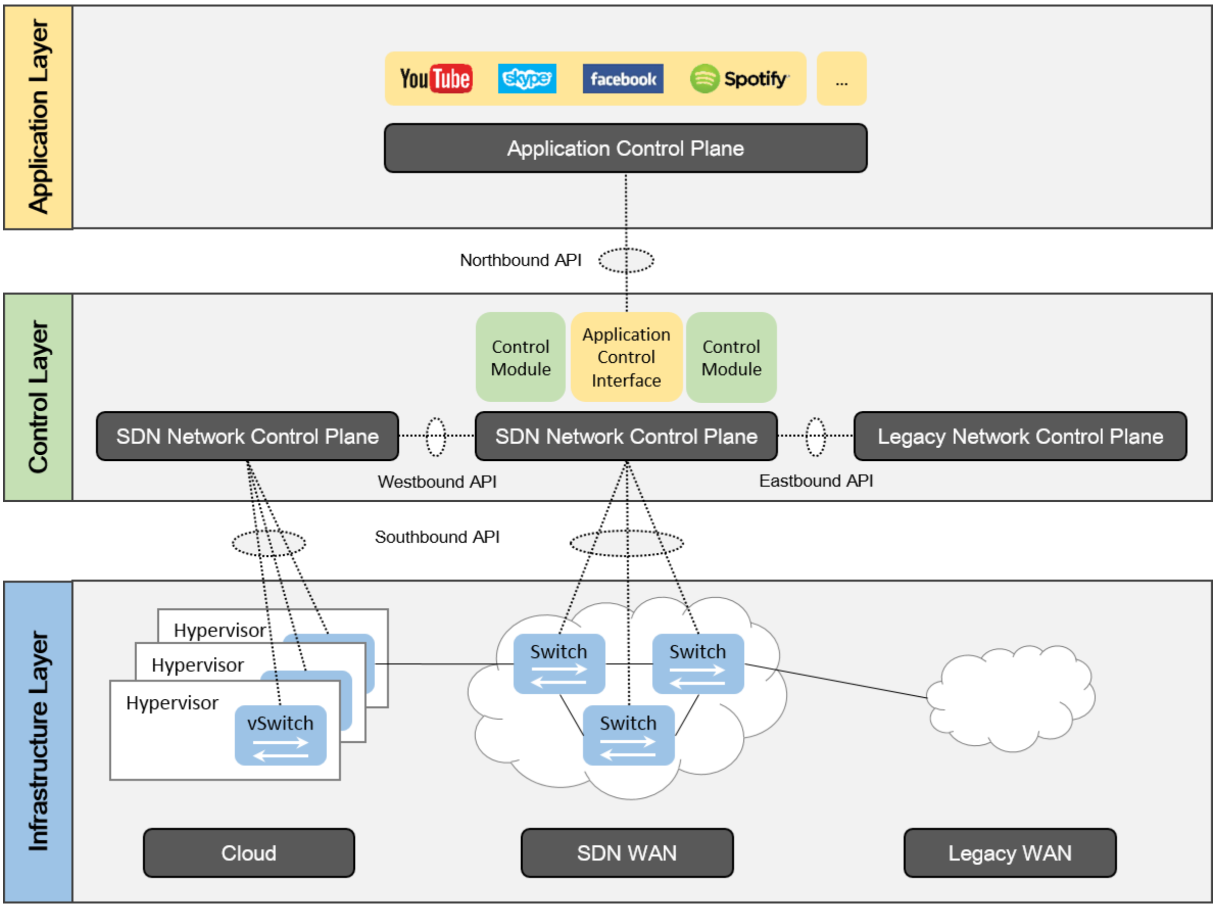

Figure 2.4. Generic SDN architecture (adapted from [Jars14]).

Regarding the constituent modules, one should note that the control layer combines distinct interconnected control instances that globally represent a controller platform responsible for adapting the operation of the physical infrastructure to application needs. Also, the cloud environment deserves special attention. It is constituted by hypervisors supporting different types of virtual machines (VMs), namely virtual switches (vSwitches). Essentially, a hypervisor manages and orchestrates the physical and logical resources of the virtualised environment. It is aware of the VMs that are using the underlying hardware and manages resource scheduling and decisions, such as migration, resource scaling, and fault and failure recovery, more efficiently to meet the specified QoS requirements of VMs [HSMA14].

With respect to the interfaces represented in Figure 2.4, one has:

- The Southbound Application Programming Interface (API), separating control and data planes, and so assuring the first fundamental principle of SDN. Its realisation is a standardised instruction set for governing the networking hardware's behaviour. The currently most notable example of implementation of this API is the OpenFlow protocol.
- The Northbound API, enabling the exchange of information between applications and the SDN controller. There is no universal, standardised Northbound API. Furthermore, as the kind of information exchanged, its form and frequency depends on the targeted application and network, such universal API is not useful.

- The Westbound API, operating as an information conduit among SDN CPs of different network domains. It allows the exchange of network state information to influence the routing decisions of each controller, at the same time enabling the seamless setup of network flows across multiple domains.
- The Eastbound API, responsible for the communication with non-SDN domains, like Multi-Protocol Label Switching (MPLS) domains. The implementation of this interface depends on the technology used in the non-SDN domain.

What makes SDN a powerful tool for network control and operation is the combination of these interfaces with some core features deeply embedded in any SDN control platform, namely programmability, protocol independence, and the ability of increasing and decreasing resource consumption based on the required capacity – elasticity. From these characteristics derives the versatile nature of SDN in terms of use cases, spanning from Cloud Orchestration and Network Management, to Routing, Load Balancing and Monitoring.

In [LiMa12], an SDN architecture is presented particularly designed for cellular networks, where the SDN controller consists of a Network Operating System (NOS) running a collection of application modules, such as RRM, mobility management, and routing, as illustrated in Figure 2.5.

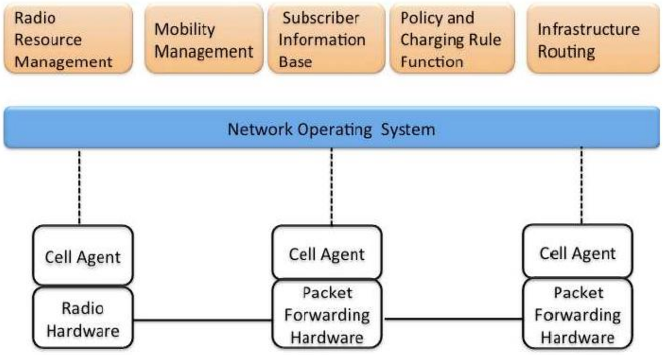
Radio Resource Management
Mobility Management
Subscriber Information Base
Policy and Charging Rule Function
Infrastructure Routing
Network Operating System
Cell Agent
Radio Hardware
Cell Agent
Packet Forwarding Hardware
Cell Agent
Packet Forwarding Hardware

Figure 2.5. Cellular SDN Architecture (extracted from [LiMa12]).

Moreover, to improve control-plane scalability, each switch runs a local control agent that performs simple actions at the behest of the controller.

In what concerns challenges arising from SDN implementation (some already referred), [Seze13] identifies four key domains, regardless of the area of application (Telco or not):

- Performance vs. Flexibility: balancing this trade-off.
- Scalability: design of scalable controllers.
- Security: protection of SDNs from malicious attacks.
- Interoperability: integration of SDN solutions into existing networks.

Several efforts are being developed to tackle these issues. This thesis addresses mainly the first two. Furthermore, while first debugging and troubleshooting techniques for SDN have been proposed, integrating them into operational network management systems is challenging, since most tools are limited to particular well-defined problems [Csas13].

So, although SDN holds great promises in terms of simplifying network deployment and operation along with lowering the associated costs, SDN's application in cellular networks is still inconclusive and embryonic [Zhou14]. In [LiMa12], the authors identify on the one hand scalability challenges that future SDN architectures should address, such as supporting many subscribers, frequent mobility, fine-grained measurement and control, and real-time adaptation, while on the other hand, various use cases are highlighted, like Monitoring for Network Control and Billing, Seamless Subscriber Mobility, and QoS Control Policies.

This thesis seeks to explore how mobile network operators can leverage SDN, NFV and Cloud paradigms to promote agility, innovation and efficiency in their networks, taking the importance of a scalable high performance architecture into account.

The success and fast progress of SDN are widely due to the success of OpenFlow [Jarr14]. The OpenFlow standard splits apart the control and data planes in network switching and routing equipment, moving the CP into a separate, centralised controller [Kemp12]. The controller communicates with the switching and routing equipment through a secure channel, using the OpenFlow protocol to program the switches with flow specifications. These specifications define the routes of packets through the network. The switches implementation is considerably simplified, since they only need to run an OpenFlow CP. Figure 2.6 illustrates the main components of an OpenFlow switch.

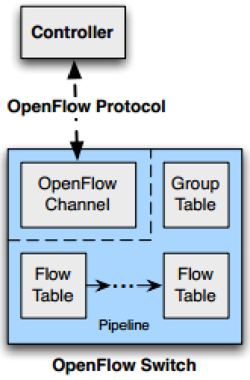

Figure 2.6. OpenFlow Architecture (extracted from [ONFo13]).

An OpenFlow switch is modelled as a group table and a collection of flow tables containing three columns: rules, actions, and counters. The rules column defines the header fields associated to each packet flow. Rules are matched against the headers of incoming packets. In case of match, the actions from the actions column are applied to the packet and the counters in the counter column are updated. Rules are applied according to its level of priority, in the event of a packet matching multiple rules. Each rule specifies an exact match from header fields or a wild card ANY. The possible actions are: modify the packet in some way, forward the packet to an output port, or send the packet to the next table or to the group table.

The group table contains group entries, and is used to implement packet duplication and multiple output. Each group entry contains the group identifier, the group type, counters and action buckets holding the actions related to the group. There are four different types of groups [ONFo13]:

- All: All action buckets are executed in this group. This type of group is used for multicast or broadcast forwarding.
- Select: One bucket is executed in this group, determined by a selection algorithm, taking for example the hash of the header. This type of group is used to implement load balancing.
- Indirect: The one defined bucket is executed in this group. This type of group allows multiple flow entries or groups to point to a common group identifier, which enables quicker action bucket execution.
- Fast Failover: The first live bucket is executed in this group. Each action bucket is associated with a specific port that controls its liveliness. This group type enables the switch to change forwarding without requiring a round trip to the controller.

Although earlier versions of OpenFlow supported just a single table, OpenFlow 1.2 and later versions support multiple tables [Kemp12]. Figure 2.7 illustrates OpenFlow's packet processing pipeline.

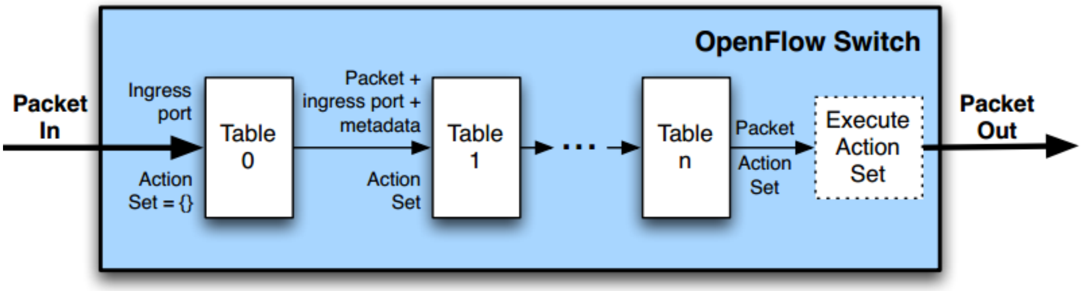

Figure 2.7. OpenFlow's Packet Processing Pipeline (extracted from [ONFo13]).

This pipeline processes each incoming packet, performing actions from the matched table entries in each table, until a packet out action is encountered. A 64 bit metadata register can be used for communication between flow tables, namely related to the matched parts of the packet.

2.3 Background on Virtualisation and Cloud

2.3.1 Virtualisation Aspects

The networking area is not only experiencing the emergence of SDN but also NV and NFV. Together, they build an automated, scalable and agile networking and cloud environment [Jamm14]. Since these three technologies are closely related, it is important to distinguish them, describe how they relate with each other, and identify the benefits and challenges of their implementation.

On the one hand, NV partitions the network logically and focuses on building tunnels (i.e., overlays) that connect different domains, so that there can be multiple virtual networks sharing the same physical infrastructure [Jamm14]. On the other hand, NFV deploys network functions into those overlays, such as DNS and firewalls [SaSt14].

So NFV, inspired by the technological shift in the IT industry introduced by server virtualisation, is primarily about using virtualisation to implement network functions on commodity servers in data centres, rather than on proprietary hardware running expensive proprietary software [Kemp14]. Hence, with NFV, network functions can be instantiated in various locations in the network as required, without the need for installation of new equipment [ETSI12]. The European Telecommunications Standards Institute (ETSI) has formed an Industry Specifications Group (ISG) to work on NFV standardisation. Figure 2.8 shows the basic blocks that constitute an NFV framework.

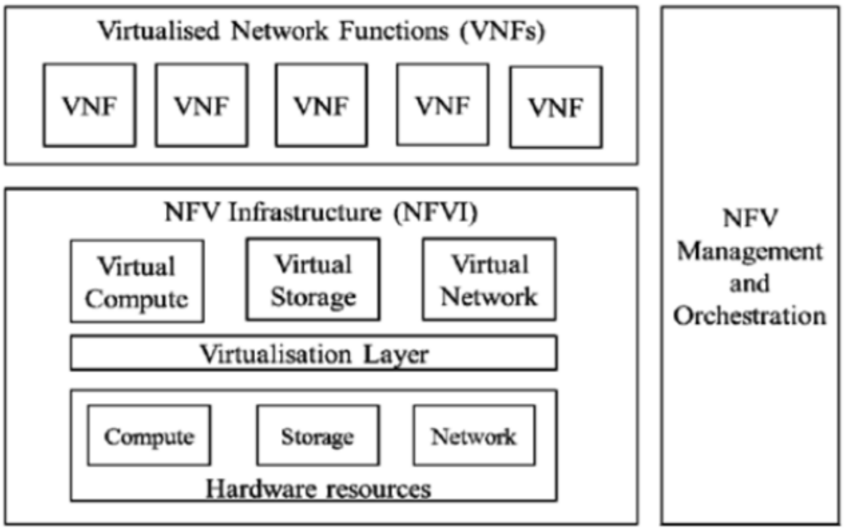

Figure 2.8. Basic NFV Framework (extracted from [Damo13]).

To achieve the objectives promised by NFV, such as flexibility in assigning virtual network functions (VNFs) to hardware, a Management and Orchestration Platform is responsible for the dynamic initiation and orchestration of VNF instances, as well as for the management of the NFV Infrastructure (NFVI) hosting environment that comprehends virtualisation technologies to meet all VNF requirements [HSMA14].

This thesis addresses the integration of SDN in the current LTE architecture, but given its close relation to NFV, it is also important to discuss the benefits and challenges brought by NFV. SDN can be interpreted as a framework that manages and automates NV and NFV. [Jamm14] states that NV can be seen as an SDN application; [Jarr14] states that NFV and SDN are considered complementary technologies and share the goals of openness and innovation, but are two independent paradigms. However, SDN support adds value to NFV by enhancing compatibility, easing maintenance problems and providing support for standardisation. Figure 2.9 shows the differences in SDN operation without and with NFV.

On the left of Figure 2.9, a number of dedicated appliances can be seen, each performing a specific network function, and all being managed by an SDN controller; on the right, functions have been virtualised and are running on a standard server platform. These functions are still managed by the SDN controller, but the overall cost of deploying and maintaining them is significantly reduced, due to the NFV approach. Recently, the OpenDaylight initiative [ODLP14] was formed to bring the NFV vision of open source software to the SDN controller area [Kemp14].

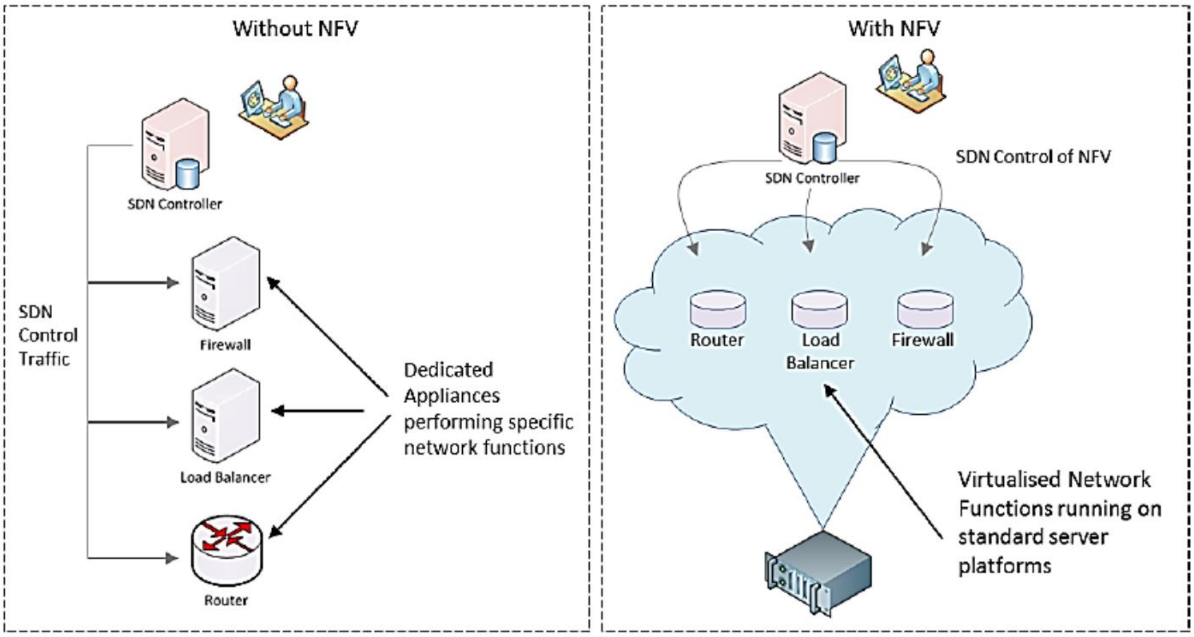

Figure 2.9. SDN Control, without NFV (left) and with NFV (right) – extracted from [Khan14].

So, NFV's implementation can bring several benefits to industry, wherein the most notable is the reduction in development time and costs for deploying new services in order to meet new business requirements [Jarr14]. A detailed list of benefits to network operators brought by NFV is given in [ETSI12]. [Bast13], for example, highlights the attractiveness of migrating the core nodes in a mobile network to an NFV environment, instead of deploying them by separate dedicated hardware boxes.

Regarding the challenges that arise for NFV implementation, various examples can be referred to [ETSI12], such as performance, automation, integration and security issues. [Bast14a], for example, states that limited attention has been drawn to the network load and infringed data-plane delay imposed by introducing NFV.

2.3.2 Cloud Radio Access Network

Cloud Radio Access Network (C-RAN) is a new base station (BS) architecture that intends to explore the benefits of applying a cloud computing approach to the radio access, as depicted in Figure 2.10.

It consists of centralising the Base Band Unit (BBU) of several classic BSs, placing it in a remote Central Office (CO). Thus, the centralised BBU is shared among multiple Remote Radio Heads (RRHs), through the implementation of resource pooling [Pizz13]. According to the function splitting between BBU and RRH, there are two possible C-RAN solutions [CMRI10], as shown in Figure 2.11:

- Fully centralisation (Solution 1): BBUs are responsible for baseband processing, as well as Layer 2 and Layer 3 functions.
- Partial centralisation (Solution 2): BBUs are only responsible for Layer 2 and Layer 3 functions.

Although fully centralisation has higher bandwidth requirements, associated with the transport of the baseband signal between BBU and RRH, it facilitates network upgrades, has better capability for supporting multi-standards operation, and is more convenient to support joint signal processing, therefore, being the solution considered in this thesis.

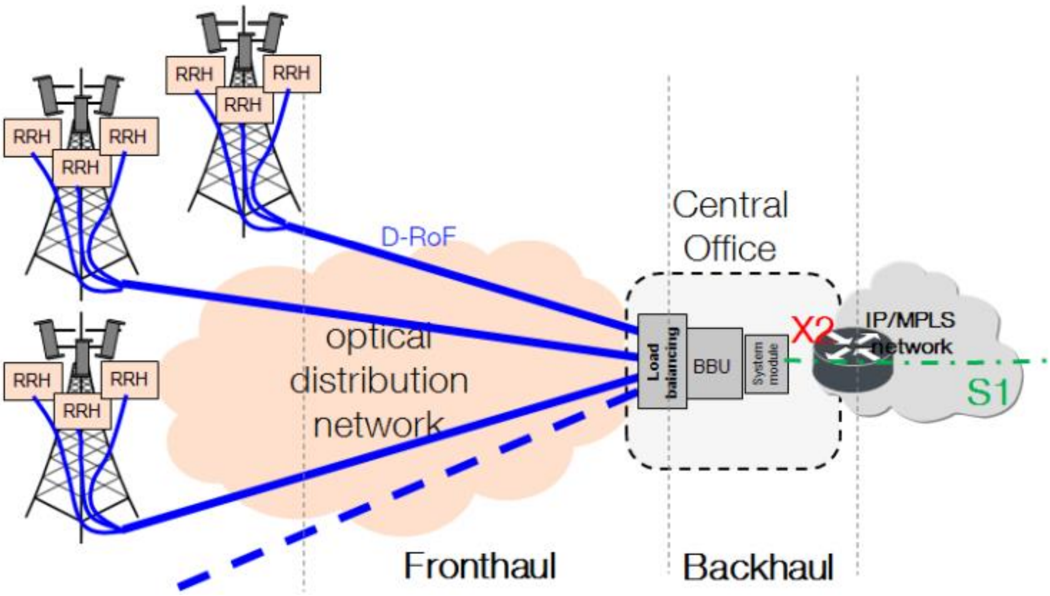

Figure 2.10. C-RAN architecture with resource pooling (extracted from [Pizz13]).

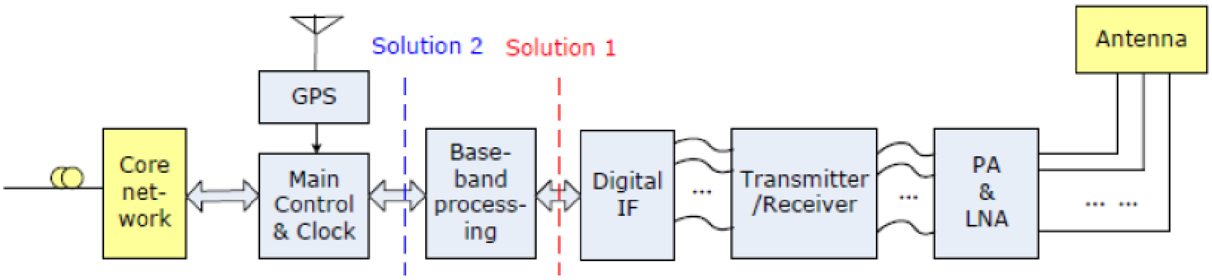

Figure 2.11. Split of functions between RRH and BBU for both architectures (extracted from [CMRI10]).

While the RRHs contain Radio Frequency (RF) transmit and receive components, and handle the Analogue to Digital/Digital to Analogue (AD/DA) conversion, the BBU is composed of high performance programmable processors and real-time virtualisation technology. The resource virtualisation allows to create a BBU pool that gathers in the same physical infrastructure several traditional BBUs, hence, enabling a logically centralised control of their resources. Improvements in joint processing and scheduling and spectral efficiency are some of the potential benefits of this central control structure.

The connection between RRHs and the BBU pool is done through a low latency optical network, in accordance with one of two standards: Common Public Radio Interface (CPRI) or Open Base Station Architecture Initiative (OBSAI) – in both, the radio signal is digitised (D-RoF, Digital Radio over Fibre). The present work takes into account CPRI and the latency and bit rate constraints associated with the transmission in the optical distribution network.

Lastly, the Load Balancer acts as a router between the BBU pool and the RRHs, assigning the radio bearers to each BBU, depending on parameters such as latency and resource usage. [Habe12] provides an in-depth explanation of this module.

The main advantages brought by C-RAN are:

- Energy Efficiency.
- Cost-saving on CAPEX and OPEX, mainly due to the reduction in the Operation & Management costs.
- Capacity Improvement.
- Adaptability to Non-uniform Traffic.
- Smart Internet Traffic Offload.

However, C-RAN deployment brings a set of constraints, namely the high bit rate capacities needed in fronthaul, the latency constraints in the fronthaul, jitter and synchronisation issues, the Baseband Pool interconnection method, and the need for a BS virtualisation technology. Research is being done to tackle these aspects.

It should be noted that SDN and NFV principles can act as drivers for C-RAN implementation, since NFV fosters C-RAN's virtualisation needs and SDN can offer flexible and programmable control features (e.g., load balancing capabilities) required for an efficient C-RAN deployment. OpenRAN ([Yang13]) represents an example of application of SDN in C-RAN.

2.4 Services and Applications

For the development of simulation scenarios, it is important to take traffic information into account in order to achieve reliable results. 3GPP specified four QoS classes to characterise traffic in UMTS: Conversational, Streaming, Interactive and Background. Table 2.1 summarises the main attributes associated with each one of these classes, showing that the basic distinguishing factor is delay sensitivity. While the Background class is suited for the most delay insensitive traffic, the Conversational one is suited for very delay sensitive traffic [3GPP14c]. These classes were not specifically set for LTE analysis, but they provide important insights on the main traffic types to consider.

Table 2.1. UMTS QoS Classes (adapted from [Corr14]).

		Service Class			
		Conversational	Streaming	Interactive	Background
Main Attributes	Real Time	Yes	Yes	No	No
	Symmetric	Yes	No	No	No
	Guaranteed Rate	Yes	Yes	No	No
	Delay	Minimum Fixed	Minimum Variable	Moderate Variable	High Variable
	Buffer	No	Yes	Yes	Yes
	Bursty	No	No	Yes	Yes
	Example	Voice	Video Streaming	Web Browsing	E-mail, SMS

In LTE, since all provided services are packet based, QoS is an important indicator. Multiple applications may be running simultaneously in a UE, each one with different QoS requirements. Supporting these multiple requirements comprehends the establishment of different bearers within the EPS, which can be classified, according to the kind of QoS they assure, into:

- Minimum Guaranteed Bit Rate (GBR), used for applications that have an associated GBR value for which dedicated transmission resources are permanently allocated at bearer establishment or modification. An example of such an application is Voice over IP (VoIP). Bit rates higher than the GBR may be allowed if resources are accessible, which entails the definition of a Maximum Bit Rate (MBR) parameter that sets an upper limit to the available bit rate.
- Non-GBR, used for applications that require no guarantees in terms of bit rate, such as web browsing or FTP transfer. Therefore, no bandwidth resources are allocated in a permanent way for these bearers.

Moreover, each bearer has an associated QoS Class Identifier (QCI), characterised by priority, packet delay budget and acceptable packet loss ratio, which determines the corresponding QoS that must be ensured in the access network by the eNodeB. A few QCIs have been standardised, for vendors to have a uniform view of the underlying service characteristics, as shown in Table 2.2. This standardisation allows an operator to expect a homogeneous traffic handling behaviour throughout its network, regardless of the manufacturer of the eNodeB equipment.

The priority and packet delay budget (and, to a certain extent, the acceptable packet loss ratio) from the QCI label define how the scheduler in the MAC handles packets sent over the bearer and are responsible for the Radio Link Control (RLC) mode configuration. For instance, one can expect a packet with higher priority to be scheduled before a packet with lower priority.

Table 2.2. Standardised QCIs for LTE (extracted from [3GPP14a]).

QCI	Resource Type	Priority	Packet Delay Budget [ms]	Packet Error Loss Ratio	Example Services
1	GBR	2	100	10^{-2}	Conversational Voice
2		4	150	10^{-3}	Conversational Video (Live Streaming)
3		3	50	10^{-3}	Real Time Gaming
4		5	300	10^{-6}	Non-Conversational Video (Buffered Streaming)
5	Non-GBR	1	100	10^{-6}	IMS Signalling
6		6	300	10^{-6}	Video (Buffered Streaming) TCP-based (e.g., www, e-mail, chat, ftp, p2p file sharing, progressive video, etc.)
7		7	100	10^{-3}	Voice, Video (Live Streaming), Interactive Gaming
8		8	300	10^{-6}	Video (Buffered Streaming) TCP-based (e.g., www, e-mail, chat, ftp, p2p file sharing, progressive video, etc.)
9		9			

2.5 State of the Art

Currently, the study of Software Defined Wireless Networks (SDWNs) is still in its infancy, and most related works focus on the SDWN architecture. OpenRoads [YapK10] is one of the first works about SDWNs, where an OpenFlow-based and backward compatible wireless network infrastructure is proposed. Network devices are controlled by NOX [Gude08]. Virtualisation is introduced by Flowvisor [Sher09]. However, OpenRoads is oriented towards WiFi and provides no special support for cellular networks. The novel concept of Software Defined Cellular Networks is considered in [LiMa12]. This work provides challenges and open issues for the integration of SDN within cellular networks. In particular, the OpenFlow controller is responsible for the forwarding procedures by allowing the network operators to direct the traffic, change the traffic paths and schedule the traffic according to QoS requirements.

[MaSe14] builds upon [LiMa12] to propose a backward-compatible modular redesign of the backhaul of mobile networks, based on the principles of SDN. Authors argue that the new design based on the centralised SDN controller not only allows for programmability and evolution of the mobile networks, but also in the long run reduces various maintenance and upgrade costs of the network. Moreover, mobility management is considered as a case study and advantages of SDN design are discussed in terms of reducing energy consumption at the UE, the communication overhead of the signalling over the backhaul and the handover delay for UEs. Still, this work lacks a quantitative analysis of the gains through a realistic implementation.

Software-defined radio access is an attractive subject of research, since the RAN imposes challenges not only related to its wireless nature, but also to the mobility features that are required. SoftRAN [Gudi13] is a software-defined centralised control plane of the RAN that proposes a virtual big-base station abstraction. The virtual big-base station consists of a central controller and radio elements (individual physical base stations). To achieve the trade-off between the optimal centralised control and the inherent delay, SoftRAN redesigns the control plane functionalities cooperatively between the controller and the radio elements. Specifically, the centralised controller handles the cross radio elements decision, while the individual radio elements deal with the frequently varying parameters. SoftRAN focuses on a scenario consisting of microcells, which is not suitable for heterogeneous deployments. In V-Cell [Rigg14], an approach similar to SoftRAN is taken, since it also relies on the principle of grouping existing cells into virtual big-base stations. However, V-Cell is more focused on smoother mobility between cells.

OpenRAN [Yang13] proposes a software-defined RAN by introducing cloud computing inspired in Cloud RAN. So, it consists of a wireless spectrum resource pool, a cloud computing resource pool and an SDN controller. The wireless spectrum pool covers multiple heterogeneous wireless networks. The cloud computing resource pool handles the baseband processing of these heterogeneous networks. According to the dynamic network requirement, the SDN controller establishes the virtual base station in the wireless spectrum resource pool, and corresponding virtual baseband processing unit in the cloud computing pool.

In [Sund13], a logically reconfigurable fronthaul for C-RAN deployments is proposed. The authors argue that by enabling this reconfiguration, BBU resources can be used more efficiently, as physical resources can be mapped onto the minimum number of necessary baseband units. The presented abstraction layer lies between the EPC and the BBUs, with virtual cells representing groupings of BBUs. A small-scale C-RAN test-bed was used to demonstrate the benefits of the presented framework, namely in terms of improving traffic demand satisfaction and reducing the compute resource usage.

CellSDN [LiMR12] and SoftCell [JLVR13] follow a different path from the previously described studies, considering the implementation of SDN in LTE’s core network. CellSDN covers both access and core networks, and deploys a NOS to abstract the control functions from both access and forwarding devices. To meet the demand for fast and frequent updates, it introduces local agents that perform real-time decision. SoftCell succeeds CellSDN and focuses on the challenges of core networks. It analyses a significant simplification of the P-GW in two dimensions: introducing software-defined access switches that implement packet classification; configuration, by an SDN controller, of optimal forwarding paths across various specific middle boxes. MobileFlow [Pent13], in its turn, analyses carrier networks and proposes a blueprint for flow-based forwarding, in order to enable a rich environment for innovation in the network.

An OpenFlow-based control plane for LTE/EPC architecture is proposed in [Said13], splitting the control and data forwarding planes related to the SGWs. The control plane is centralised and uses the OpenFlow protocol to remotely manage the SGW data plane. It is shown that the proposed architecture easily ensures an on-demand connectivity service even in critic situations, such as network equipment failure and overload situations. Nonetheless, resiliency and load balancing aspects are not covered in this study. In [Sama14], the same authors emphasise the programmability that SDN can bring to the EPC, showing how the data plane can be easily configured thanks to OpenFlow. In addition, the signalling load of the proposed architecture is evaluated and compared to that of 3GPP LTE/EPC architecture. Results show that the signalling load is significantly reduced with OpenFlow.

In [Bast13], EPC nodes are analysed and their functions are classified according to their impact on data-plane and control-plane processing. A mapping for these functions onto four alternative deployment frameworks based on SDN and OpenFlow is proposed. In addition, the current OpenFlow implementation’s capability to realise basic core operations such as QoS, data classification, tunnelling and charging is analysed. The analysis shows that functions which involve high data packet processing such as tunnelling, have more potential to be kept on the data-plane network element, i.e., realised by an OpenFlow Switch.

In [Bast14a], the authors discussed alternative deployments for mobile core network gateways, namely fully virtualised gateways hosted in a data centre and decomposed gateways based on SDN. Possible placements of virtualised gateways or decomposed gateway functions with respect to delay and imposed network load are analysed. A packet processing delay measurement for the virtualised and decomposed gateway implementations is provided. For fully virtualised gateways, the processing delay results in a maximum of 132µs compared to 15µs in case of decomposed gateways. However, only uniform traffic demands are considered, so it still lacks quantitative evaluation of the impact of virtualisation and SDN on mobile networks and their flexibility gain considering the time-varying property of the traffic.

In [[Bast14b]], an architecture that supports the virtualisation of the mobile core network gateways is introduced, where the gateways are realised by software instances hosted in datacentres and SDN based network elements at the transport network. Authors formulated a model for the time-varying traffic patterns that can be observed within the mobile network core according to the user population and traffic intensity changing with time. The trade-off between transport network load and power consumption is quantified, showing the advantage of considering the time-varying property of the traffic for network dimensioning.

Given the close connection between NV and SDN, it is crucial to explore how these approaches can cooperate to provide efficient network operation. FlowVisor [Sher09] combines NV and SDN. Different from common network slicing techniques such as MPLS and VLAN, FlowVisor is a hardware abstraction layer that uses OpenFlow, partitioning the flow space in each switch in different slices. It hosts multiple guest controllers, one per slice, and guarantees data and control plane isolation between slices, so that each controller can just manage its own slice. [Phil12] provides an architecture for eNodeB virtualisation based on OpenFlow, through the usage of FlowVisor. Furthermore, it should be noted that NFV deployments are ultimately a combination of virtualisation techniques and SDN.

OpeNB [Cost14] presents a novel framework for virtualising LTE base stations to improve user performance and lower DL interference. Moreover, it attempts to use SDN and OpenFlow to introduce NV in LTE networks with an adaptable, extensible and less invasive manner. A mechanism that allows the virtualisation of LTE base stations depending on the network state is presented. Simulation results show that the proposed framework improves the average interference, throughput and packet loss experienced by the end users. Furthermore, the proposed framework enables an operator to lease on-demand the physical infrastructure and resources of additional eNodeBs (usually owned by different operators).

[Kemp14] presents a new architectural approach supporting rapid and flexible cross-domain service orchestration and management: Service Provider SDN. The approach is inspired by the SDN work for transport networks and the NFV work for operator networks, but adds to these the capability to rapidly define and manage innovative cross-domain services using Web APIs. Two examples of cross-domain APIs and associated services are presented: Cloud Atlas, allowing rapid, elastic virtual network connectivity to be orchestrated end-to-end from a data centre into the wide area network; Network Slices, providing on-demand policy based QoS for mobile cloud services. However, this study lacks quantitative measurements to access its performance.

[John13] presents NSC as a service deployment concept that promises increased flexibility and cost efficiency for future carrier networks, leveraging SDN and NFV. It provides a detailed array of research directions in the context of NSC, including service instance deployment, network service definition, programming, and operations, as well as the concept of continuous network service delivery.

These studies provide a background and a guideline to the work performed in this thesis.

Chapter 3

Model Development

A description of the model used in this thesis is provided in this chapter, starting with its metrics and overview, and proceeding with a more detailed analysis of its implementation, concerning the access and core parts of the network. The chapter ends with a brief assessment of the model.

3.1 Model Parameters

In this section, the metrics that underlie the model used in the current work are described.

3.1.1 Delay

Packet data latency or delay is a key performance metric in today's communication systems that is regularly measured not only by vendors and operators but also by end-users, e.g., via speed test applications. There are several applications that do not require a very high data rate, but that do require very low delay. The present work mainly focuses on user plane delay.

Delay can be measured by the time it takes for a small IP packet to travel from the terminal through the network to the server, and back. This measure is called round trip time (RTT), which has a more meaningful impact on quality of experience (QoE) than one-way delay (OWD). Therefore, RTT is the delay measure considered in this work, even though most of the information available regards one-way delay (e.g., Packet Delay Budget column in Table 2.2). In order to simplify the analysis the present work assumes that RTT is given by:

$$\delta_{RTT\ [\mathrm{ms}]} = 2 \cdot \delta_{OWD\ [\mathrm{ms}]} \tag{3.1}$$

where:

- δ_{OWD} – One-way delay.

The end-to-end RTT is given by (3.2) and estimates for its maximum and minimum values are shown in Table 3.1 – these estimates are based on values present in [3GPP14e] (RAN), [Eric15] (Backhaul) and [Nika11] (EPC, IP Backbone, Server).

$$\delta_{end-to-end\ [\mathrm{ms}]} = \delta_{RAN\ [\mathrm{ms}]} + \delta_{bh\ [\mathrm{ms}]} + \delta_{EPC\ [\mathrm{ms}]} + \delta_{bb\ [\mathrm{ms}]} + \delta_{server\ [\mathrm{ms}]} \tag{3.2}$$

where:

- δ_{RAN} – RTT in the RAN;
- δ_{bh} – RTT in the Backhaul;
- δ_{EPC} – EPC delay;
- δ_{bb} – RTT in the Backbone;
- δ_{server} – Server delay.

Table 3.1. End-to-end RTT estimate.

End-to-End RTT Components [ms]									
RAN		Backhaul		EPC		IP Backbone		Server	
Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
7	10	2	12	1	4	30	300	1	4

Overall End-to-End RTT [ms]	
Min.	Max.
41	330

Thus, it is clear that the end-to-end RTT results from a sum of various delay components, some due to propagation in links and others due to processing and queueing in network nodes, e.g., the EPC delay is just related to processing and queueing, while the Backhaul one consists of a sum of processing and queueing delays in routers and propagation delays in the links between them – nonetheless, for the purpose of this thesis, the Backhaul delay is considered to be exclusively due to link propagation.

Furthermore, with respect to the values in Table 3.1, one should notice that the IP Backbone delay is very hard to estimate and predict, since it will certainly depend on a set of different network policies present along the packet path. Also, the delays in the EPC and Server might change depending on the number and type of sessions and requests that need to be handled.

When using NSC, some packets might experience an extra delay, particularly those that have not been included in the flow tables of OpenFlow switches yet. This fact results in an extra component that must be used for computing the end-to-end RTT (typical values for the NSC delay component are presented in Section 4.4):

$$\delta_{end-to-end\,[\mathrm{ms}]}^{NSC} = \delta_{end-to-end\,[\mathrm{ms}]} + \delta_{NSC\,[\mathrm{ms}]} \tag{3.3}$$

where:

- $\delta_{end-to-end}$ – End-to-end RTT, disregarding NSC;
- δ_{NSC} – Delay due to NSC operations.

Also, the NSC delay comes from three sources: RTT in controller-switch communication, processing in the controller, and flow table update procedure in switches.

The values presented in Table 3.1 for the RAN are explained in detail in Table 3.2, in a one-way perspective. One shows how the delay in the RAN critically depends on the percentage of HARQ Retransmissions.

Table 3.2. RAN delay components (adapted from [3GPP14e]).

Process	OWD [ms]	
	0% HARQ Retransmissions	30% HARQ Retransmissions
UE wakeup time	Implementation dependent – Not included.	Implementation dependent – Not included.
UE Processing Delay	1	
Frame Alignment	0.5	
TTI for UL Data Packet (Piggy back scheduling information)	1	
HARQ Retransmission	0	0.3×5
eNB Processing Delay	1	
Total	3.5	5

One must note that the eNodeB processing delay in Table 3.2 is for the traditional eNodeB, where BBU and RRH are co-located. However, since the present thesis is intended to have a Cloud RAN approach, where every RRH is connected to a BBU Pool through a fronthaul link, it is important to further detail how the eNodeB delay is distributed among the different cloud configuration components. So, in C-RAN (see Figure 3.1) the eNodeB delay can be computed through:

$$\delta_{eNodeB\,[\mathrm{ms}]} = \delta_{RRH\,[\mathrm{ms}]} + \delta_{fh\,[\mathrm{ms}]} + \delta_{BBUpool\,[\mathrm{ms}]} \quad (3.4)$$

where:

- δ_{RRH} – Delay due to processing in the RRH;
- δ_{fh} – RTT in the Fronthaul;
- $\delta_{BBUpool}$ – Delay due to processing in the BBU.

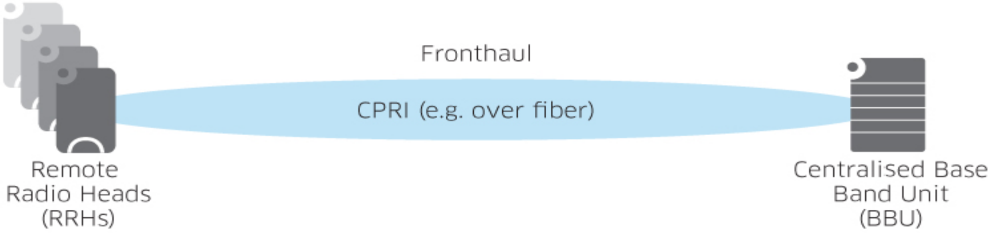

Figure 3.1. C-RAN's eNodeB perspective.

The introduction of this fronthaul delay means that the BBU and RRH processing must be expedited, for the eNodeB processing delay shown in Table 3.2 to remain unchanged. Typical values for the maximum fronthaul delay are presented in the next chapter.

The length of the fronthaul link is dictated by its characteristic speed (200 km/ms for fibre) and by the fronthaul OWD, hence:

$$d_{fh\,[\mathrm{km}]} = \mathrm{v}_{[\mathrm{km/ms}]} \cdot \frac{\delta_{fh\,[\mathrm{ms}]}}{2} \quad (3.5)$$

where:

- v – Transmission speed in the link;
- δ_{fh} – RTT in the Fronthaul.

Equation (3.5) can of course be also used to obtain the RTT in the fronthaul as a function of the length of the fronthaul link.

3.1.2 Capacity

Three types of capacity are considered in the present work:

- Network link capacity – for the front- and backhauls.
- Network node capacity – for BBU Pools and for Core Nodes.
- Blade Server Capacity.

Fronthaul link capacity depends on the maximum traffic demand that can be supported by the corresponding cell site, which in turn depends on the number and type of RRHs that the cell site possesses. As referred to in Chapter 2, the most widely used protocol for fronthaul communication is CPRI, hence, the data rates concerning a single RRH presented in Table 3.3 are the ones considered for fronthaul capacity, for the purpose of this thesis.

Table 3.3. CPRI fronthaul required capacity in function of radio technologies (adapted from [Pizz13]).

RAN	GSM 1T2R	WCDMA 1T2R	LTE 10MHz 2x2	LTE 20MHz 2x2	LTE 20MHz 4x4	LTE 100MHz 8x8
CPRI data rate [Mbps]	24.608	614.4	1 228.8	2 457.6	4 915.2	49 152

The value denoted for an LTE 100 MHz 8x8 RRH was inferred from the three preceding values in the table, which evidence that either an increase in bandwidth or in the MIMO order by a certain factor cause a CPRI data rate increase by that same factor. Also, it should be noted that this value of almost 50 Gbps pinpoints the possibility of taking full advantage of LTE-A features, such as Carrier Aggregation.

The backhaul link capacity is regarded as depending on the number and capacity of the fronthaul links that must be supported, and on a 10% margin accommodating the BBU Pool introduced overhead:

$$C_{backhaul\,[\mathrm{Mbps}]} = 1.1 \cdot \sum_{i=1}^{N_{fhlinks}} C_{fronthaul,i\,[\mathrm{Mbps}]} \tag{3.6}$$

where

- $N_{fhlinks}$ – Number of fronthaul links;
- $C_{fronthaul,i}$ – Capacity of fronthaul link i.

Network node capacity in its turn depends just on the number and capacity of links that the node must support. Thereby, the capacities required for a BBU Pool and for a Core Node are given by (3.7) and (3.8), respectively:

$$C_{BBUpool\,[\mathrm{Mbps}]} = \sum_{i=1}^{N_{fhlinks}} C_{fronthaul,i\,[\mathrm{Mbps}]} \tag{3.7}$$

$$C_{CoreNode\,[\mathrm{Mbps}]} = \sum_{i=1}^{N_{bhlinks}} C_{backhaul,i\,[\mathrm{Mbps}]} \tag{3.8}$$

where:

- $N_{bhlinks}$ – Number of backhaul links;
- $C_{backhaul,i}$ – Capacity of backhaul link i.

Blade Server Capacity is critical for determining the physical dimension of every network node, since, for a given network node capacity, blade servers with a higher capacity mean that a small number of them will be required:

$$N_{BladeServers} = \frac{C_{Node\,[\mathrm{Mbps}]}}{C_{Server\,[\mathrm{Mbps}]}} \tag{3.9}$$

where:

- C_{Node} – Capacity of a network node (BBU Pool or Core Node);
- C_{Server} – Capacity of a single Blade Server.

No load balancing considerations are done in this thesis, so every cell site is statically associated with a BBU Pool – the same happens with the association of BBU Pools with Core Nodes.

3.2 Model Overview

The model described in this section in a high level perspective aims at drawing insights about the positioning, quantity and delay characteristics of network resources needed to support the set of cell sites available countrywide. The network resources addressed are fronthaul links, BBU Pools, backhaul links and Core Nodes, hence, this model has two kinds of outputs: the ones related to the RAN and the ones related to the Core part of the network, as shown in Figure 3.2. In this figure, besides the inputs and outputs of the model, one can also see, in light blue, the four main blocks that constitute the model: RAN Dimensioning, Core Dimensioning, Data Processing and Data Fitting. It must be highlighted that the EPC, Backbone, Server and NSC delays previously referred to are used for the end-to-end estimate in Chapter 4, but are out of the scope of the model here described.

Regarding the outputs, it is worth saying that they can be further divided into two classes: the ones associated with the maximum delay constraint, and the ones associated with the blade server capacity constraint. In what concerns the inputs, it should be mentioned that the cell sites positioning is static (data provided by a mobile operator) and also that the maximum delay array leads to the RAN and Core delay-related outputs, while the maximum blade server capacity array leads to the RAN and Core capacity-related outputs. Moreover, the BBU Pools Positioning input used in Core Dimensioning correspond to the highest value of maximum fronthaul delay used in RAN Dimensioning.

The first step of RAN Dimensioning positions the majority of BBU Pools. However, this first step leads to disconnected cell sites, so the second step of this module has the goal of finding disconnected cell sites that can be connected to one of the neighbouring BBU Pools. The third step of RAN Dimensioning creates and positions extra BBU Pools for sites or groups of sites that ended up disconnected even after the first two steps. Finally, the last step of RAN Dimensioning accounts for the total capacity that needs to be satisfied in every BBU Pool, accounting for the number and class of supported cell sites. Core Dimensioning follows an identical algorithm to the one used in RAN Dimensioning, differing fundamentally in the delay constraint, which in this case is maximum backhaul delay instead of maximum fronthaul delay, and instead of having cell sites, BBU Pools and fronthaul links, one has BBU Pools, Core Nodes and backhaul links, respectively. Also, all backhaul links are assumed to work over fibre – no microwave links are regarded for this part of the network.

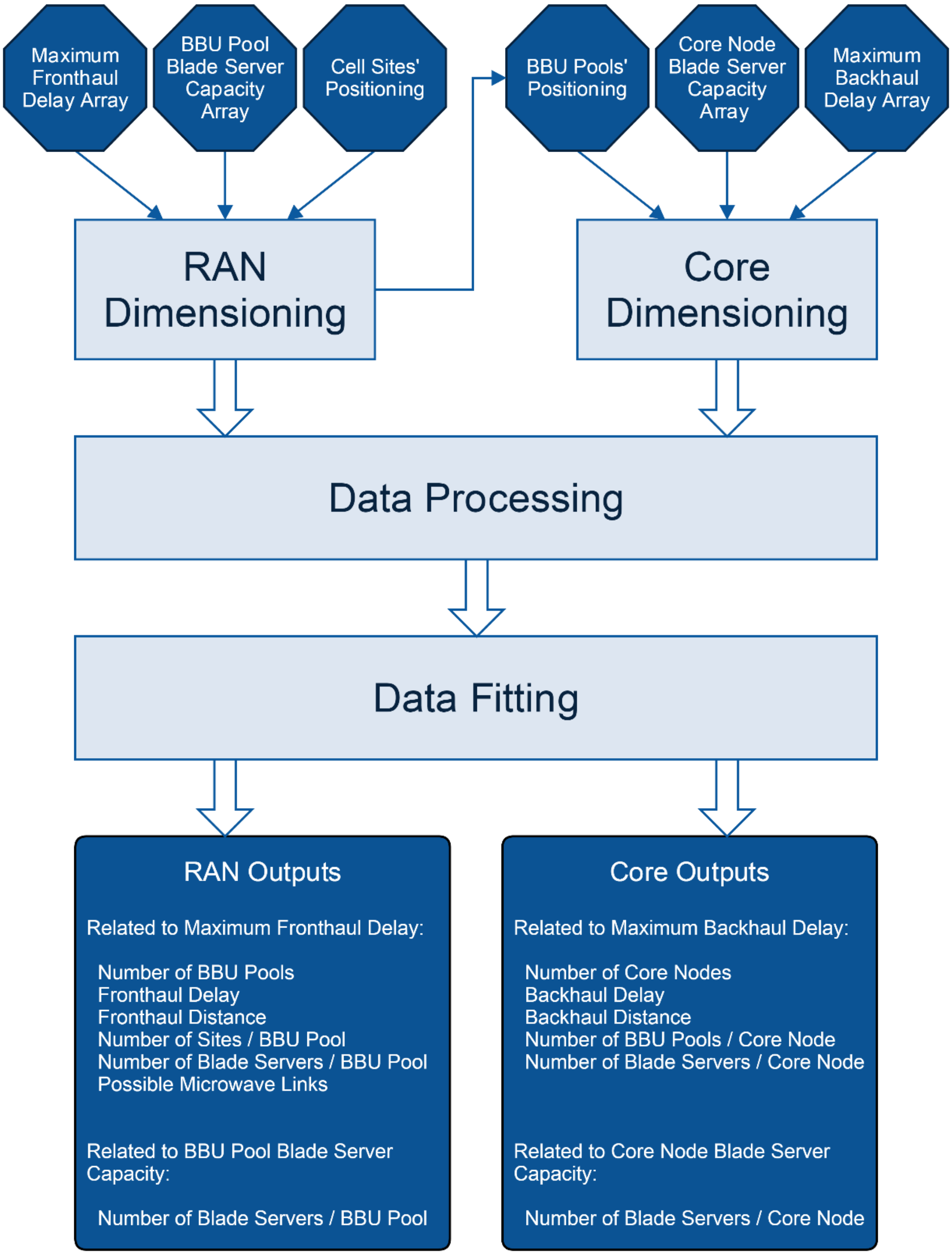
Maximum Fronthaul Delay Array
BBU Pool Blade Server Capacity Array
Cell Sites' Positioning
BBU Pools' Positioning
Core Node Blade Server Capacity Array
Maximum Backhaul Delay Array
RAN Dimensioning
Core Dimensioning
Data Processing
Data Fitting
RAN Outputs
Related to Maximum Fronthaul Delay:
Number of BBU Pools
Fronthaul Delay
Fronthaul Distance
Number of Sites / BBU Pool
Number of Blade Servers / BBU Pool
Possible Microwave Links
Related to BBU Pool Blade Server Capacity:
Number of Blade Servers / BBU Pool
Core Outputs
Related to Maximum Backhaul Delay:
Number of Core Nodes
Backhaul Delay
Backhaul Distance
Number of BBU Pools / Core Node
Number of Blade Servers / Core Node
Related to Core Node Blade Server Capacity:
Number of Blade Servers / Core Node

Figure 3.2. Model Overview.

The Data Processing and Data Fitting blocks of the model aim at drawing meaningful insights and gathering trends for the data obtained from RAN and Core Dimensioning. So, Data Processing is responsible for analysing delay, distance and capacity demand tables and computing the relevant outputs – first, the ones related to maximum delay constraints in the front- and backhauls, and then the ones concerning the number of required blade servers in BBU Pools and Core Nodes for different capacity values per unit. The Data Fitting block is responsible for fitting the data sets coming from the Data Processing stage with suitable mathematical models, in order to provide a clearer knowledge of the kind of behaviour presented by the various outputs. This fitting stage is carried out through the usage of Matlab’s Curve Fitting Tool. The selection of the most suitable mathematical model for fitting the data is done in a first approach by analysing which one has the highest adjusted correlation, R^2, and lowest Root Mean Square Error (RMSE). When more than one of the mathematical models have very similar R^2 and RMSE, one has chosen for the fitting the one that is the most suitable from a theoretical viewpoint.

3.3 Model Implementation

3.3.1 RAN Dimensioning

The developed algorithm takes cell site positioning as an input to find a possible geographical distribution for BBU Pools and Core Nodes, as well as the capacity requirements of these nodes – although the geographical distribution is not fully optimised, still it offers a good estimate of how the nodes and links could effectively be placed. Figure 3.3 to Figure 3.5 present the first three steps in the RAN Dimensioning stage, which define the positioning of BBU Pools, while Figure 3.6 shows the last step in this stage of the model, which consists of a capacity study of the previously placed BBU Pools.

After loading the input data, consisting of the positioning of cell sites and the set of values considered for the maximum fronthaul delay, the placement of BBU Pools is done for each one of these values. Starting at a point with Portugal’s westernmost longitude and northernmost latitude, a set of square form BBU Pool coverage areas is placed until all country is covered – this grid is not more than a first estimate of the coverage areas that will effectively exist. Representing the coverage areas with circles would be more accurate, with the sites in the overlapping zones being connected to possibly more than one BBU Pool, but since load balancing is not under the scope of this work and to simplify the problem of attributing sites to BBU Pools, the square approach was adopted. The size of the squares is determined by the maximum fronthaul value in use, so for example if 100 μs is used, and considering a speed of 200 km/ms in the fibre, a circle diameter of 20 km would be indicated for the coverage areas, but since a square form approach is used then a diagonal of 20 km is used for every square area.

Before placing a new square form coverage area, it is checked if at least part of it is within the geographical limits under analysis. Each time the referred condition is verified, a new square is placed in the map and the number of sites within the square is counted – if there are no sites within the square, the algorithm continues to the next square placement, but if there is at least one site then two cases are distinguished.

In the first one, there is just one site within the square that is added to the list of disconnected sites in order to be handled in a later phase; because it is not beneficial to have a BBU Pool serving a single site, such situation is avoided as much as possible; later it might happen that this site can be successfully connected to a BBU Pool located in a neighbouring square, leveraging the multiplexing gains of C-RAN.

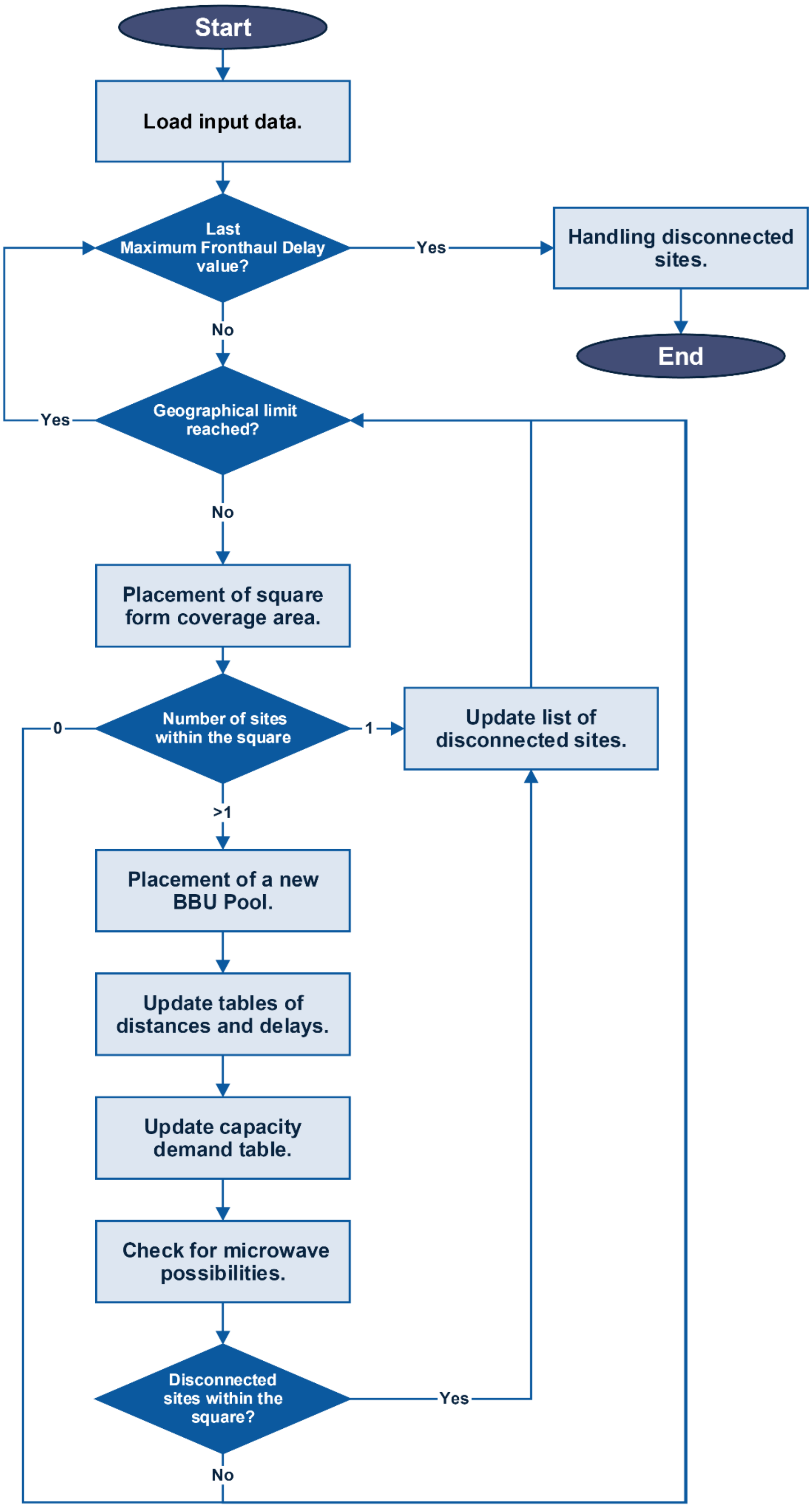
Start
Load input data.
Last Maximum Fronthaul Delay value?
Yes
Handling disconnected sites.
End
No
Yes
Geographical limit reached?
No
Placement of square form coverage area.
0
Number of sites within the square
1
Update list of disconnected sites.
>1
Placement of a new BBU Pool.
Update tables of distances and delays.
Update capacity demand table.
Check for microwave possibilities.
Disconnected sites within the square?
Yes
No

Figure 3.3. RAN Dimensioning – Part I.

The second case is the one where multiple cell sites are found within the square. In this situation, a BBU Pool is placed within the area under analysis, with its coordinates computed as an average of the coordinates of the cell sites within the square. This procedure for computing the coordinates of the BBU Pool leads to a scenario where delays are smaller comparing to a scenario where the BBU Pool is placed at the centre of the square. Also, placing the BBU Pool always at the centre of every square would lead to situations where it would be out of Portugal's area, laying in Spain or in the ocean.

Anyway, one should note that this procedure for placing the BBU Pool has also the important consequence of shifting its effective coverage area, which means that at this point the square area ceases to accurately represent the coverage of the BBU Pool. For example, if within a square nearly every site is very close to the bottom left corner, but there is one site very close to the top right corner, since the BBU Pool will most likely be near the bottom left corner, then the cell site in the top right corner will need to be connected to another BBU Pool, because the BBU Pool near the bottom left corner will not fulfil the latency requirements for that particular site. This means that cell sites within the square area might be too far from the BBU Pool, which thus are added to the list of disconnected sites, so that in a later phase of the algorithm it is decided if any of them needs a dedicated BBU Pool or if some or all of them can be connected to BBU Pools located in neighbouring areas.

After the BBU Pool is put in place, the fronthaul distances table and consequently the fronthaul delays table are updated, taking into account the distances between the cell sites and the newly placed BBU Pool. At this point, it is verified if any of the added distances is equal or less than 2 km. For those that verify this condition, the list of possible microwave links is updated, although it should be highlighted that microwave technology might not be viable in many cases due to capacity issues – this viability study in terms of microwave capacity is outside the scope of the present thesis.

Next, a survey of the sites connected to the BBU Pool is made, in order to gather the data for, in the Data Processing stage, computing the total amount of capacity that the BBU Pool needs to support. As more sites are connected, a higher capacity demand is at stake. This will also allow, by using a blade server capacity value, to compute the number of blade servers that must be present in the BBU Pool in order to support the capacity demand associated with the connected cell sites.

After the first part of RAN Dimensioning is concluded and in order to ensure fronthaul connectivity to the cell sites that remain disconnected, possibly with the placement of extra BBU Pools, the second and third parts take place – Figure 3.4 and Figure 3.5. All disconnected sites are inspected in order to check which can be connected to neighbouring BBU Pools, i.e., if it is at a distance less or equal than the one corresponding to the maximum fronthaul delay, with a link speed of 200 km/ms – the ones that cannot are handled in the third part. When a cell site can be connected to a neighbouring BBU Pool, it is removed from the list of disconnected sites, the list of sites associated with the BBU Pool is updated, as well as the tables containing the fronthaul distances and delays. Then, it is verified if the length of the fronthaul link is less or equal to 2 km – if it is, the list of possible microwave links is updated. This phase of the algorithm continues until the whole list of disconnected sites is analysed and updated. At this point, the third part of RAN Dimensioning begins – Figure 3.5.

For every site in the disconnected list, it is checked if there are neighbouring ones that can form a cluster of sites connected to a single BBU Pool, according to its coordinates. If that is not possible, then a new BBU Pool is placed and co-located with the site and the number of single site BBU Pools is updated, but if it is, then the new BBU Pool is placed in the same way as followed in Part I – its coordinates are computed as the average of the coordinates of the cell sites.

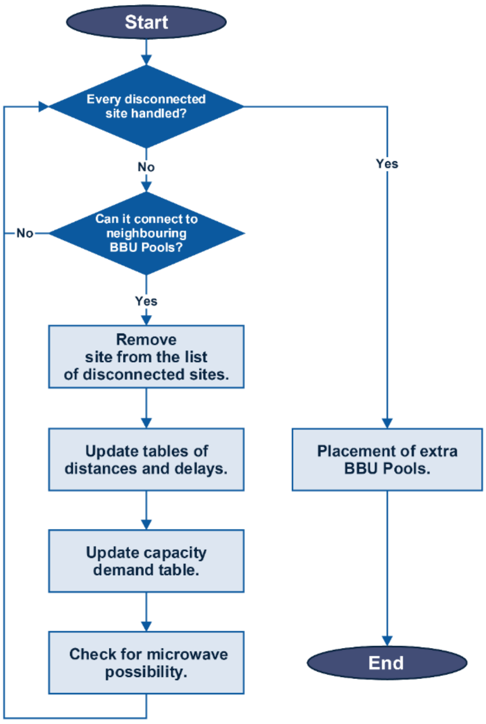
Start
Every disconnected site handled?
No
Yes
Can it connect to neighbouring BBU Pools?
No
Yes
Remove site from the list of disconnected sites.
Update tables of distances and delays.
Update capacity demand table.
Check for microwave possibility.
Placement of extra BBU Pools.
End

Figure 3.4. Handling disconnected sites (RAN Dimensioning – Part II).

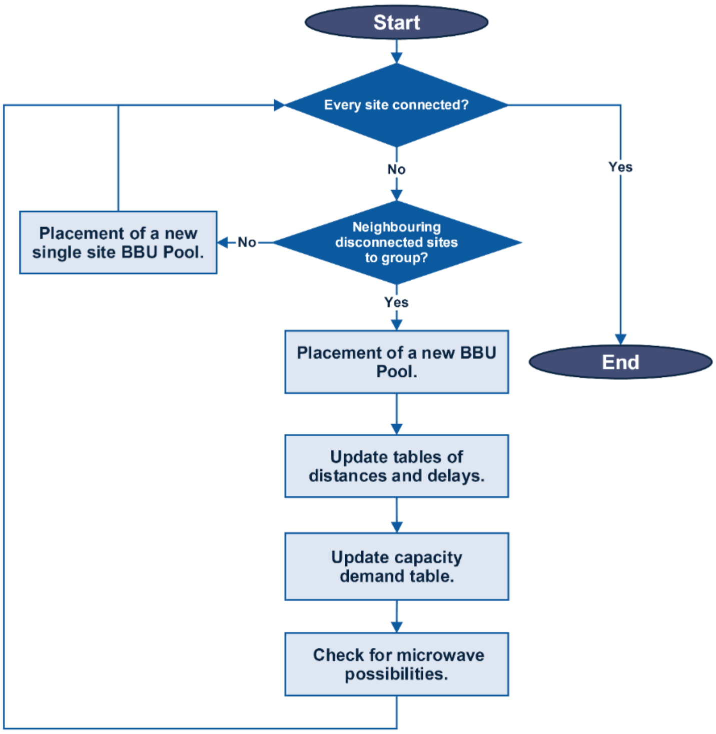
Start
Every site connected?
No
Yes
Neighbouring disconnected sites to group?
No
Placement of a new single site BBU Pool.
Yes
Placement of a new BBU Pool.
End
Update tables of distances and delays.
Update capacity demand table.
Check for microwave possibilities.

Figure 3.5. Placement of extra BBU Pools (RAN Dimensioning – Part III).

In any case, the tables containing the fronthaul distances and delays are updated. Furthermore, for the cases where the length of the fronthaul link is less or equal to 2 km, the list of possible microwave links is updated.

After this part of RAN Dimensioning is completed, every cell site in the country is associated with a BBU Pool and all BBU Pools are positioned. Next, follows a basic capacity assessment of every BBU Pool, which simply consists of determining the blade servers demand imposed by each site in the corresponding BBU Pool – see Figure 3.6. This will allow the computation of the number of blade servers required for each one of the previously placed BBU Pools, in Data Processing.

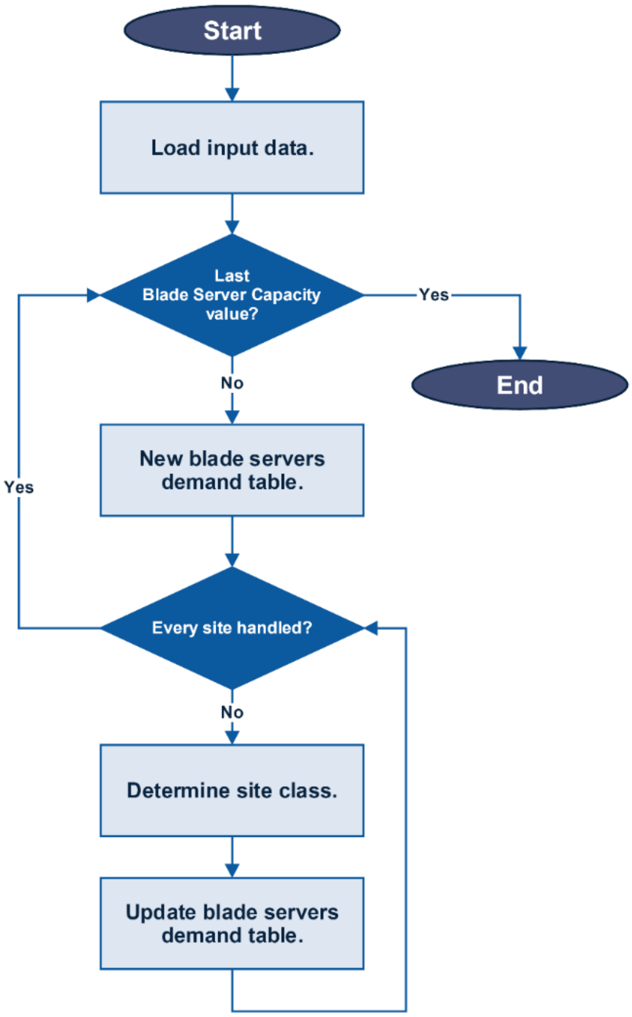
Start
Load input data.
Last Blade Server Capacity value?
Yes
End
No
New blade servers demand table.
Yes
Every site handled?
No
Determine site class.
Update blade servers demand table.

Figure 3.6. Survey of blade servers' demand per site, depending on blade server capacity (RAN Dimensioning – Part IV).

So, after the positioning of the BBU Pools for a given maximum fronthaul delay is loaded, the above procedure takes place. A new blade servers' demand table is created for each value of blade server capacity. First, the class of every site is examined, according to its location – cell sites in the surroundings of the district capital have different capacity requirements than the rest of the cell sites, as further explained Section 4.1. Then, the capacity of the BBU Pool supporting the site under analysis is updated according to the site class. These procedure goes on until the capacity of every BBU Pool is calculated. RAN Dimensioning is thereby concluded.

3.3.2 Core Dimensioning

The algorithm structure for Core Dimensioning, and consequently its flowcharts are identical to the ones for RAN Dimensioning, except that instead of having cell sites, BBU Pools and fronthaul links, one has BBU Pools, Core Nodes and backhaul links, respectively. Figure 3.7 to Figure 3.9 present the first three steps in the second stage of the model, which define the positioning of Core Nodes, while Figure 3.10 shows the last step in the second stage of the model, which consists of a capacity study of the previously placed Core Nodes.

After loading the input data, consisting of the positioning of BBU Pools and the set of values considered for the maximum backhaul delay, the placement of Core Nodes is done for each one of these values. The method used for this placement is identical to the one used in the RAN stage, starting by the placement of square form coverage areas for each Core Node. Here, the size of the squares is determined by the maximum backhaul delay, hence, if for example this delay is of 2 ms, then considering a speed of 200 km/ms in the fibre and neglecting delays in backhaul routers, a diagonal of 400 km is used for every square area.

Before placing a new square form coverage area, it is checked if at least part of it is within the geographical limits under analysis. Each time the referred condition is verified, a new square area is placed in the map and the number of BBU Pools within the square is counted – if there are no BBU Pools within the square, the algorithm continues to the next square placement, but if there is at least one BBU Pool then two cases are distinguished. In the first case, there is just one BBU Pool within the square, which is added to the list of disconnected BBU Pools in order to be handled in a later phase, because it is not beneficial to have a Core Node serving a single BBU Pool, hence, such situation is avoided as much as possible – later it might happen that this BBU Pool can be successfully connected to a Core Node located in a neighbouring square. Again, one should note that this procedure for placing the Core Node has the important consequence of shifting its effective coverage area.

After a Core Node is put in place, the backhaul distances table and consequently the backhaul delays table are updated, taking into account the distances between the BBU Pools and the newly placed Core Node. All backhaul links are considered fibre links, hence, here there is no test for possibilities of using microwave technology by measurement of links' length.

Next, a survey of the BBU Pools connected to the Core Node is made, in order to gather the data for, in the Data Processing stage, computing the total amount of capacity that the Core Node needs to support. As more sites are connected, a higher capacity demand is at stake. This will also allow, by using a blade server capacity value, to compute the number of blade servers that must be present in the Core Node in order to support the capacity demand associated with the connected BBU Pools.

In order to ensure backhaul connectivity to the BBU Pools that remain disconnected, possibly with the placement of extra Core Nodes, the second and third parts take place, Figure 3.8 and Figure 3.9. All disconnected BBU Pools are inspected in order to check which can be connected to neighbouring Core Nodes, i.e., if it is at a distance less or equal than the one corresponding to the maximum backhaul delay, with a link of 200 km/ms.

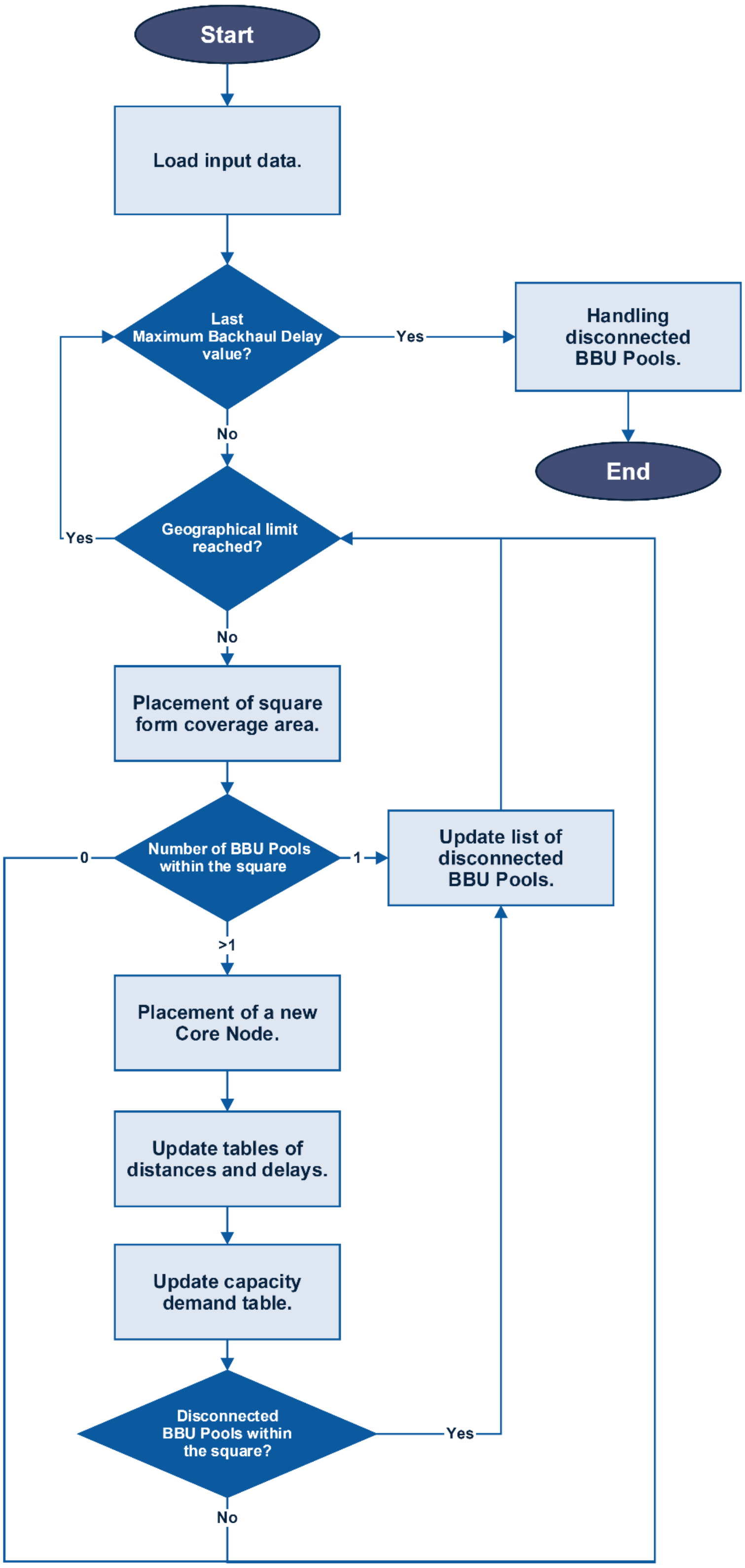
Start
Load input data.
Last Maximum Backhaul Delay value?
Yes
Handling disconnected BBU Pools.
End
No
Geographical limit reached?
Yes
No
Placement of square form coverage area.
Number of BBU Pools within the square
0
1
Update list of disconnected BBU Pools.
>1
Placement of a new Core Node.
Update tables of distances and delays.
Update capacity demand table.
Disconnected BBU Pools within the square?
Yes
No

Figure 3.7. Core Dimensioning – Part I.

When a BBU Pool can be connected to a neighbouring Core Node, it is removed from the list of disconnected BBU Pools, the number of BBU Pools associated with the Core Node is updated, as well as the tables containing the backhaul distances and delays. This phase of the algorithm continues until the whole list of disconnected BBU Pools is analysed and updated.

At this point, the third part of Core Dimensioning begins – Figure 3.9. For every BBU Pool in the disconnected list, it is checked if there are neighbouring ones that can form a cluster of BBU Pools connected to a single Core Node, according to its coordinates. If that is not possible, then a new Core Node is placed and co-located with the BBU Pool and the number of single BBU Pool Core Nodes is updated, but if it is, then the new Core Node is placed in the same way followed in the first part of Core Dimensioning – its coordinates are computed as the average of the coordinates of the BBU Pools. In any case, the tables containing the backhaul distances and delays are updated.

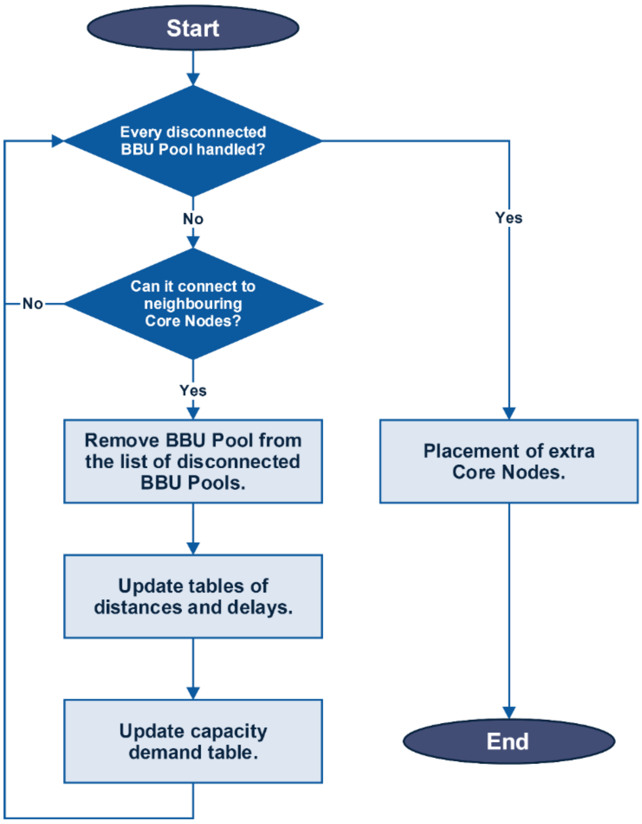
Start
Every disconnected BBU Pool handled?
No
Yes
Can it connect to neighbouring Core Nodes?
No
Yes
Remove BBU Pool from the list of disconnected BBU Pools.
Placement of extra Core Nodes.
Update tables of distances and delays.
Update capacity demand table.
End

Figure 3.8. Handling disconnected BBU Pools (Core Dimensioning – Part II).

After this part of Core Dimensioning is completed, every BBU Pool in the country is associated with a Core Node and all Core Nodes are positioned. Next follows a basic capacity assessment of every Core Node, which simply consists of determining how many blade servers are required for each one of the previously placed Core Nodes – see Figure 3.10.

So, after the positioning of the Core Nodes for a given maximum backhaul delay is loaded, the procedure above takes place for a set of blade server capacity values. The capacity of a Core Node supporting a given group of BBU Pools will be computed in Data Processing depending on the capacity associated with its backhaul links. Finally, the number of blade servers required for each Core Node is computed for a certain blade server capacity. Core Dimensioning is thereby concluded.

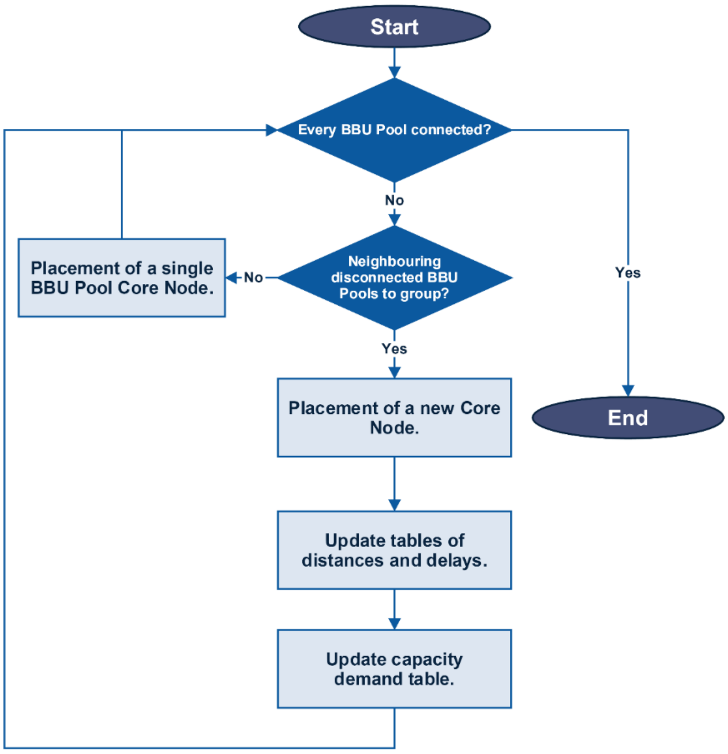
Start
Every BBU Pool connected?
No
Yes
Neighbouring disconnected BBU Pools to group?
No
Placement of a single BBU Pool Core Node.
Yes
Placement of a new Core Node.
End
Update tables of distances and delays.
Update capacity demand table.

Figure 3.9. Placement of extra Core Nodes (Core Dimensioning – Part III).

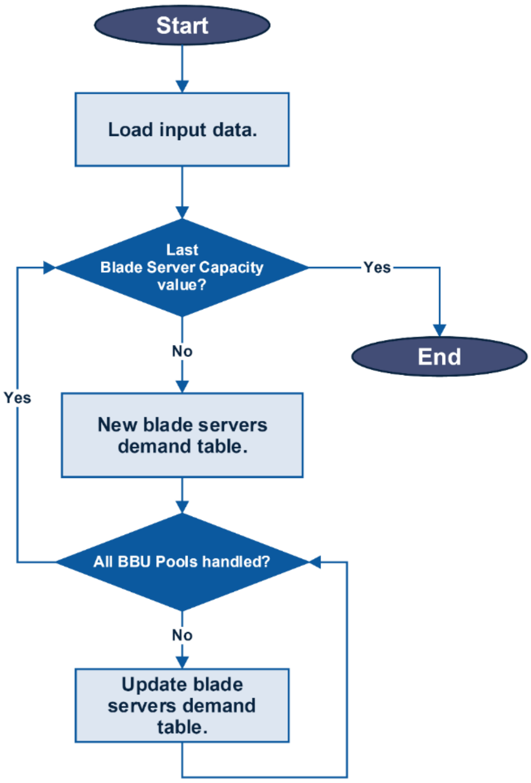
Start
Load input data.
Last Blade Server Capacity value?
Yes
End
No
Yes
New blade servers demand table.
All BBU Pools handled?
No
Update blade servers demand table.

Figure 3.10. Survey of blade servers' demand per BBU Pool, depending on blade server capacity (Core Dimensioning – Part IV).

3.3.3 Data Processing

The Data Processing stage takes the data acquired in RAN and Core Dimensionings and computes the outputs. Hence, tables containing fronthaul delays and distances between BBU Pools and their corresponding cell sites, fronthaul capacity demand tables and other important outputs of the RAN Dimensioning stage are analysed in order to obtain pertinent statistics (maximum, minimum, average and standard deviation) for the countrywide scenario and data insights, such as:

- The number of BBU Pools required countrywide;
- Statistics concerning the fronthaul delay and the fronthaul distance;
- Share of single site BBU Pools;
- Statistics concerning the number of sites and consequently the required number of blade servers per BBU Pool, for different values of the maximum fronthaul delay constraint;
- The evolution of the required number of blade servers per BBU Pool as the blade server capacity is changed.

The fronthaul delays, distances, capacity demand and blade servers demand tables have a column per BBU Pool, where in each cell is respectively the delay, distance, capacity demand and blade servers demand of a given cell site – as described in previous sections, the first three tables change with the maximum fronthaul delay constraint, while the blade server capacity demand table changes with the blade server capacity constraint. Through the data available in these tables, one can easily compute all the outputs listed above, regarding the RAN examination. Also, it must be noted that the averages computed at this point are simple arithmetic means.

After the processing of RAN-related data, a similar processing takes place for Core-related data. Here, tables containing backhaul delays and distances between Core Nodes and their corresponding BBU Pools, backhaul capacity demand tables and other important outputs of the Core Dimensioning stage are analysed in order to obtain relevant insights, namely:

- The number of Core Nodes required countrywide;
- Statistics concerning the backhaul delay and the backhaul distance;
- Share of single BBU Pool Core Nodes;
- Statistics concerning the number of BBU Pools and consequently the required number of blade servers per Core Node, for different values of the maximum backhaul delay constraint;
- The evolution of the required number of blade servers per Core Node as the blade server capacity is changed.

The backhaul delays, distances, capacity demand and blade servers demand tables have a column per Core Node, where in each cell is respectively the delay, distance, capacity demand and blade servers demand of a given BBU Pool – again, the first three tables change with the maximum backhaul delay constraint, while the blade server capacity demand table changes with the blade server capacity constraint. Through these tables, all Core-related output variables can be easily computed. For example, the required number of blade servers per Core Node is calculated by taking the sum, column by column, in the blade servers demand table.

3.3.4 Data Fitting

The Data Fitting phase of the implementation is carried out through the usage of Matlab's Curve Fitting Tool [Matl15] and considering the mathematical models presented in Table 3.4. The selection of the most suitable mathematical model for fitting the data is done in a first approach by analysing which has the highest correlation coefficient, *R*^{2}, which varies from 0 to 1, 1 being a perfect fit. Thus, *R*^{2} measures how successful the fit is in explaining the variation of the data, i.e., represents the similarity between the response values and the predicted ones. It is also called coefficient of multiple determination, and is given by:

$$\varSigma_{regression} = \sum_{i=1}^{n} (\widehat{y_i} - \mu_y)^2 \tag{3.10}$$

$$\varSigma_{\mu} = \sum_{i=1}^{n} (y_i - \mu_y)^2 \tag{3.11}$$

$$R^2 = \frac{\varSigma_{regression}}{\varSigma_{\mu}} \tag{3.12}$$

where:

- y_i – Observation i;
- μ_y – Average of the observations of variable y;
- $\widehat{y}_i$ – Estimated or predicted value of y_i;
- n – Number of observations;
- $\varSigma_{regression}$ – Sum of squares of the regression;
- $\varSigma_{\mu}$ – Sum of squares about the mean.

As explained in Matlab's documentation, when, e.g., *R*^{2} = 0.508, one can conclude that the model explains about 50% of the variability in the response variable. In order to further ensure the reliability of the fitting used, the RMSE is presented as an auxiliary measure of the goodness of fit – a lower value of this parameter corresponds to a better fitting, so it should be as close to zero as possible, in comparison with the values of the observations. This second statistic is given by:

$$\sqrt{\overline{\varepsilon^2}} = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \widehat{y_i})^2}{n}} \tag{3.13}$$

where:

- y_i – Observation *i*;
- $\widehat{y}_i$ – Estimated or predicted value of y_i;
- n – Number of observations.

Table 3.4. Models used for Data Fitting.

Model Name	Expression
Rational	$f(x) = \frac{a}{x+b} \ , \ a,b \in \mathbb{R}$
Linear	$f(x) = a \cdot x + b \ , \ a,b \in \mathbb{R}$
Quadratic	$f(x) = a \cdot x^2 + b \cdot x + c \ , \ a,b,c \in \mathbb{R}$
Exponential	$f(x) = a \cdot e^{bx} \ , \ a,b \in \mathbb{R}$

In the present work, a model with an R^2 above 0.95 is considered to be a very accurate description of the data set under study. When more than one of the mathematical models have very similar R^2 and RMSE, it is chosen for the fitting the one that is the most suitable from a theoretical point of view.

3.4 Model Assessment

During the model development, in order to validate its implementation, the obtained outputs were subjected to a set of empirical tests. Basically, as the scripts were under construction, a careful examination of all variables was made, in order to check if they were coherent and also accurate from a theoretical point of view. Table 3.5 presents a list of the most critical tests for RAN Dimensioning, done in order to validate the results in Chapter 4.

A similar set of tests was done for Core Dimensioning, since the positioning algorithm is identical to the one used in RAN Dimensioning – the main difference was that instead of dealing with BBU Pool positioning, these tests dealt with Core Node positioning. To further extend the model assessment, the model implementation was applied to one part of the country, Figure 3.11, thus, consisting of a subset of the complete set of cell sites provided by the mobile operator. The part of the country chosen, for no particular reason, was the one in between 39.5º and 41.5º of Latitude, which includes cities like Leiria and Porto, as well as rural areas.

So, the evolution of the more relevant variables related to the change on maximum fronthaul delay was compared with the set expected trends. It should be noted that the assessment of the behaviour of the number of blade servers per BBU Pool with the increase in blade server capacity is not presented, because it follows exactly the theoretical descending trend.

For testing the positioning algorithm and thus validate the model, the variables analysed were the total number of BBU Pools, the average fronthaul delay, the average fronthaul distance, the share of possible microwave links, the average number of sites per BBU Pool and the average number of blade servers per BBU Pool. As expected, the number of BBU Pools decreases with the increase in the maximum fronthaul delay – the same happens with the percentage of possible microwave links, apart from the last value. The rest of the tested variables present an expected ascending trend – the decrease in the average fronthaul distance and in the average fronthaul delay as the maximum fronthaul delay increases from 75 µs to 200 µs is negligible.

Table 3.5. List of empirical tests that were made to validate model implementation.

Number	Description
1	Validation of the .csv input file read, by verifying if the number of cell sites coordinates' pairs loaded to Matlab was equal to the number of rows of the file (4188).
2	Scatter plot of the cell sites' locations, in order to visually inspect their rightness.
3	Validation of the coverage areas: - Check if the set of square form coverage areas was not exceeding the geographical scope of the country. - Check if all empty square areas (with no cell sites inside) are lacking of BBU Pool, as they should.
4	Verification of the correct update of the disconnected sites list, i.e., if it happens every time that a single site square coverage area is found or when a BBU Pool placement within a square form coverage area leads to cell site disconnection.
5	Verification of the correct computation of BBU Pool locations, as the average of the geographical coordinates of the respective cell sites.
6	Validation of the distances and delay tables update: - Check if it is happening every time a new BBU Pool is placed. - Check if the values are correctly computed and stored.
7	Verification of cell site connection completion, i.e., if the number of loaded pairs of coordinates equals the number of connected cell sites.
8	Check if all sites selected as possibly supported by microwave technology are in fact 2 km of less away from the BBU Pool to which they are associated.
9	Verify if the process of handling disconnected sites referred in Figure 3.3Figure 3.4 and further described in Figure 3.4 and Figure 3.5 is correctly implemented: - Check if the entire list of disconnected sites is being envisaged – no sites left unexamined. - Check if the sites identified as disconnected only originate extra BBU Pools if and only if they can't connect to neighbouring BBU Pools.
10	Check correct computation of the number of blade servers for different values of blade server capacity and according to the number of supported cell sites by BBU Pool.
11	Verification of the correct plot of all outputs.

Table 3.6. Output variables' behaviour vs. expected trend.

Variable	Maximum Fronthaul Delay [µs]						Expected Trend	Validation
	75	100	125	150	175	200		
$\boldsymbol{N_{BBUPools}}$	116	68	52	37	29	22	Descending	OK
$\boldsymbol{\mu_{\delta_{fh}}}$ [µs]	31.84	41.86	54.01	62.18	76.31	75.43	Ascending	OK
$\boldsymbol{\mu_{d_{fh}}}$ [km]	6.36	8.37	10.80	12.43	15.26	15.08	Ascending	OK
$\boldsymbol{N_{mwlinks}}$ [%]	8.46	6.53	2.71	2.15	0.66	1.60	Descending	OK
$\boldsymbol{\mu_{N_{Sites/BBUPool}}}$	16	27	35	49	63	83	Ascending	OK
$\boldsymbol{\mu_{N_{Servers/BBUPool}}}$	29	50	65	91	116	153	Ascending	OK

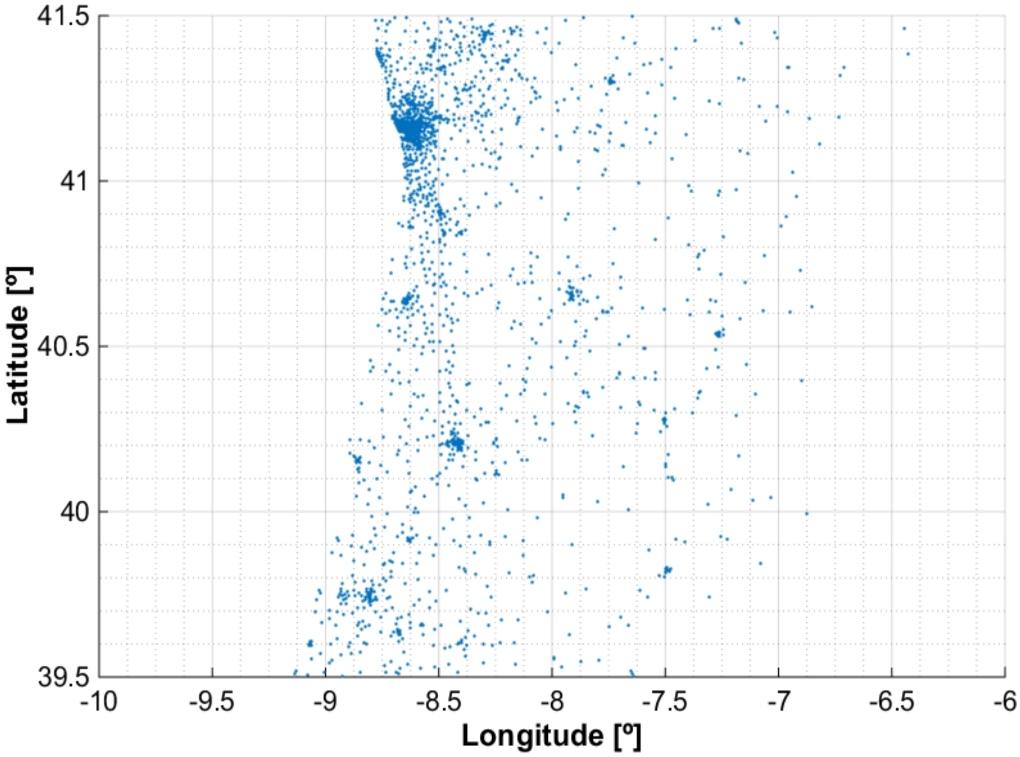

Figure 3.11. Set of cell sites considered for the Model Assessment.

Chapter 4

Results Analysis

This chapter presents the considered scenario along with the associated results and respective analysis. The scenario is defined in Section 4.1 and the results for the RAN and Core are presented in Sections 4.2 and 4.3, respectively. Lastly, an end-to-end latency estimation is given in Section 4.4.

4.1 Scenario

The scenario for this thesis is Portugal. In Figure 4.1, one presents in blue the cell sites from a mobile operator covering the entire continental area of the country, which were considered for the thesis.

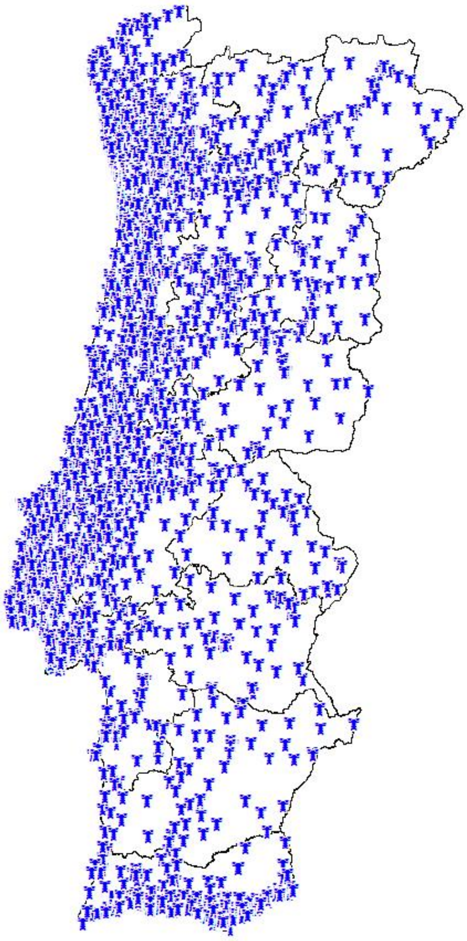

Figure 4.1. Base stations' positioning of a mobile operator in Portugal (extracted from [Corr14]).

Portugal has an area of approximately 92 090 km^2 and nearly 10.5 million inhabitants, wherein most of the population lives in coastal areas – in Lisbon's Metropolitan Area lives roughly a quarter of the people. This asymmetric distribution of the population within the country's area is reflected in the positioning of base stations – Figure 4.1 clearly shows a much lower density of sites in the countryside, in comparison with coastal areas.

Although no specific user distribution and traffic mix are considered, the network is hereby studied for the worst case scenario of traffic load – all RRHs with active users, using all resources available. The values shown in Table 3.3 for fronthaul capacity take this situation into consideration; based on these, three classes of trisector sites are used, see Table 4.1.

Table 4.1. Classes of sites and corresponding link capacity demand.

Site Class	Number of RRHs	Radio Access Technologies	Link Capacity Demand [Gbps]
1	15	2G (1 band) 3G (2 bands) LTE 20MHz 2x2 MIMO (2 bands)	18.506
2	15	2G (1 band) 3G (2 bands) LTE 20MHz 4x4 MIMO (2 bands)	33.251
3	9	2G (1 band) 3G (1 band) LTE 100MHz 8x8 MIMO (1 band)	149.373

For the maximum fronthaul delay impact study, all sites were considered as class 1, but in order to further aggravate the capacity demand, in the sections that follow, the three classes of sites were used – all sites are class 3 within a radius of 20 km around Lisbon and Porto, and all sites are class 2 within 10 km around the remaining district capitals. Figure 4.2 illustrates this division of sites into classes, used in Section 4.2.2 and onward – the blue dots represent the distribution of cell sites.

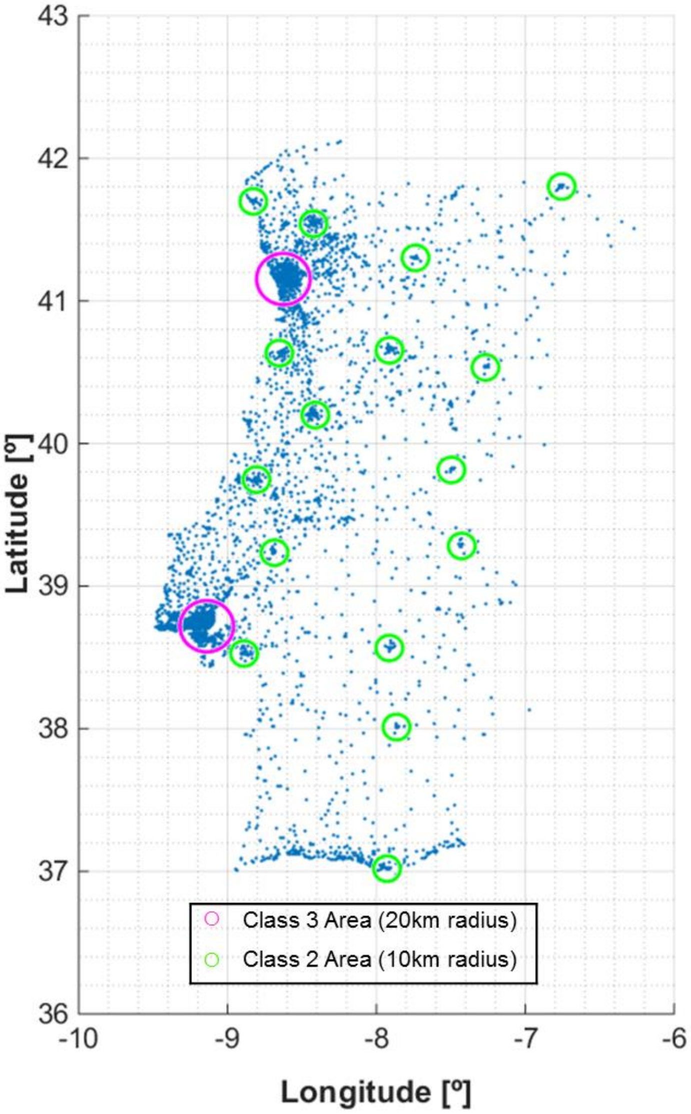

Figure 4.2. Classes of sites used in Section 4.2.2 and onward.

Although the fronthaul capacity demand of class 3 sites might seem exaggerated, the forecasted increase in video streaming and gaming traffic, along with M2M, may require 100 MHz bandwidth and 8x8 MIMO, particularly in urban regions like Lisbon and Porto, where the number of users and machines is bigger.

Two layers of data centres are considered in order to support the traffic demand faced by the set of cell sites previously illustrated, with the first layer hosting the BBU Pools and the second one hosting virtual machines responsible for Core operations, such as QoS control and billing. No capacity restrictions for the data centres are considered, meaning that BBU Pools can support a virtually infinite number of sites, and Core Nodes can support a virtually infinite number of BBU Pools, as long as the adequate number of blade servers is present.

Previous studies point that the fronthaul RTT spans in between 150 µs and 400 µs – Table 4.2 displays the maximum and minimum values. The values are expressed in terms of OWD, because it is the quantity that matters for computing the length of the fronthaul link.

Table 4.2. Processing and Fronthaul Propagation OWD values taken as input.

Fronthaul Propagation [µs]		**BBU + RRH processing [µs]**	
Minimum	Maximum	Minimum	Maximum
75	200	300	425

Also, one assumes that for situations when sites end up located 2 km or less away from their BBU Pool, the link between BBU Pool and cell site may be over microwave technology instead of fibre. This assumption is used only when all sites are considered to be class 1 and has the purpose of exploring situations where the installation of fibre links is too costly, and thus where microwave technology is appealing. Still, it must be highlighted that even current state of the art microwave technology would hardly meet class 1 site capacity requirements, and it certainly would not be capable of supporting class 2 and 3 ones.

Regarding blade server capacity, two different ranges are considered – one for BBU Pools and another for Core Nodes. While the capacity of a single blade server belonging to a BBU Pool is considered to be in between 10 Gbps and 60 Gbps, for a blade server belonging to a Core Node it is considered to be in between 50 Gbps and 100 Gbps. This assumption takes into consideration the fact that a Core Node has to handle a much larger amount of information then the one required for a BBU Pool, since it gathers a large number of data flows coming from the BBU Pools for which connectivity is being provided.

An estimate of the end-to-end delay is proposed in Section 4.4. Typical delay values for the RAN, EPC, Backhaul, Server and NSC, as well as various reference values for backbone delay – Table 4.3 shows these parameters.

Table 4.3. Delay parameters concerning Section 4.4.

$\boldsymbol{\delta_{RAN}}$ [ms]	$\boldsymbol{\delta_{bh}}$ [ms]	$\boldsymbol{\delta_{EPC}}$ [ms]	$\boldsymbol{\delta_{bb}}$ [ms]	$\boldsymbol{\delta_{server}}$ [ms]	$\boldsymbol{\delta_{NSC}}$ [ms]
10	12	4	Dependent on the server location	4	4

The goal in the last section of results is to analyse the different contributions for the total end-to-end

RTT. Twelve different server locations among five continents are tested, which results in twelve different backbone delays that are shown in Section 4.4. The delay value considered for NSC was obtained in laboratory by a manufacturer – values for commercial solutions are not available yet.

4.2 RAN Analysis

4.2.1 Maximum Fronthaul Delay Impact

To determine the effect of the imposed maximum fronthaul delay, several outputs were analysed for different values of this constraint, namely the number of BBU Pools, the fronthaul delay and distance, the percentage of links that can possibly work over microwave technology instead of fibre, the number of sites per BBU Pool and the number of blade servers per BBU Pool. The symbols μ and σ in the following plots refer to the average and the standard deviation, respectively. Figure 4.3 to Figure 4.8 show how the algorithm places the BBU Pools (in red) countrywide for 75 μs, 100 μs, 125 μs, 150 μs, 175 μs and 200 μs – the dots in blue represent the cell sites where the RRHs are placed, which in the present section are all regarded as class 1.

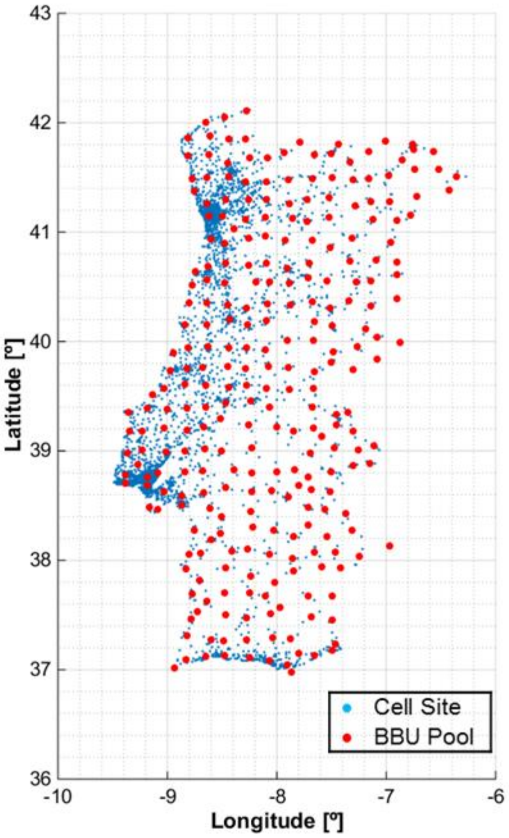

Figure 4.3. Placement of BBU Pools (in red) for a maximum fronthaul delay of 75 μs.

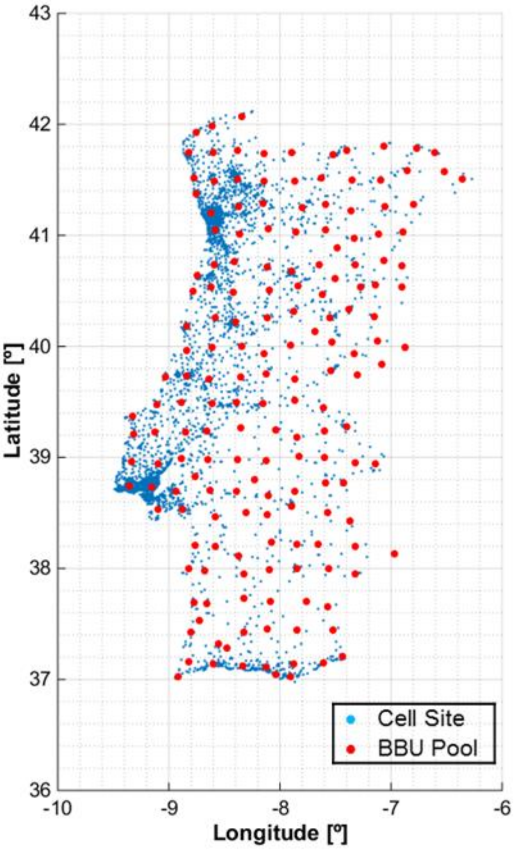

Figure 4.4. Placement of BBU Pools (in red) for a maximum fronthaul delay of 100 μs.

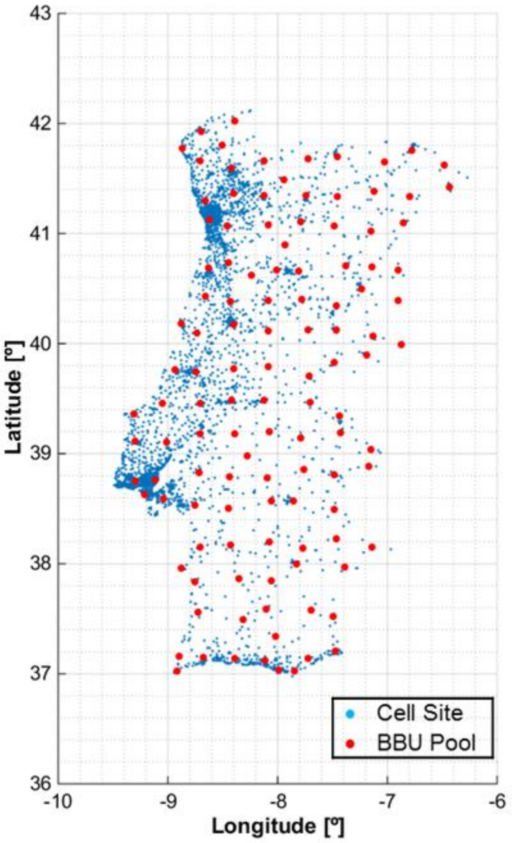

Figure 4.5. Placement of BBU Pools (in red) for a maximum fronthaul delay of 125 μs.

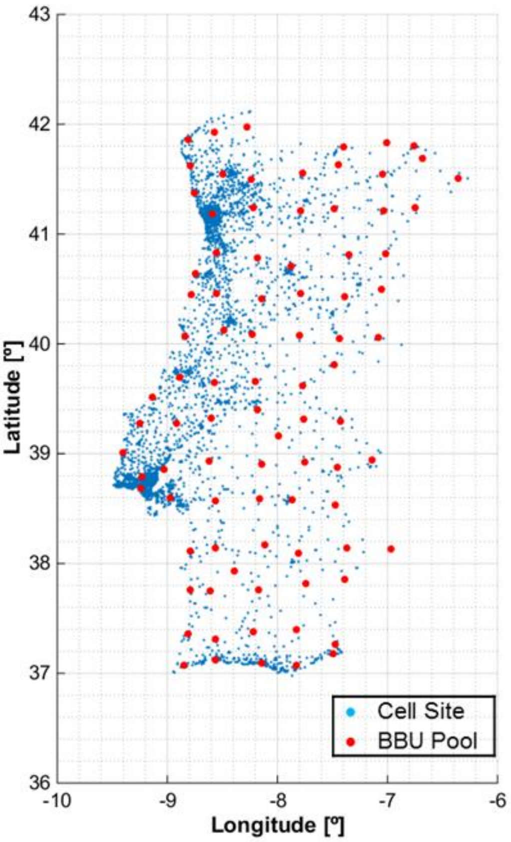

Figure 4.6. Placement of BBU Pools (in red) for a maximum fronthaul delay of 150 μs.

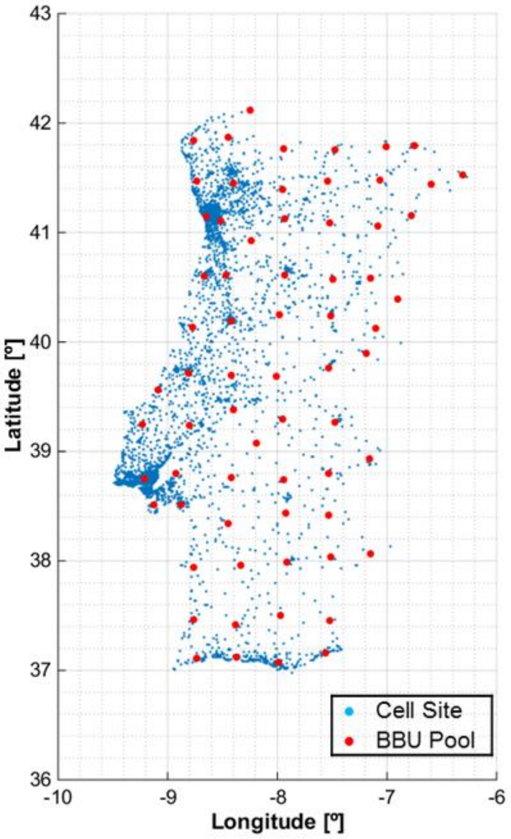

Figure 4.7. Placement of BBU Pools (in red) for a maximum fronthaul delay of 175 μs.

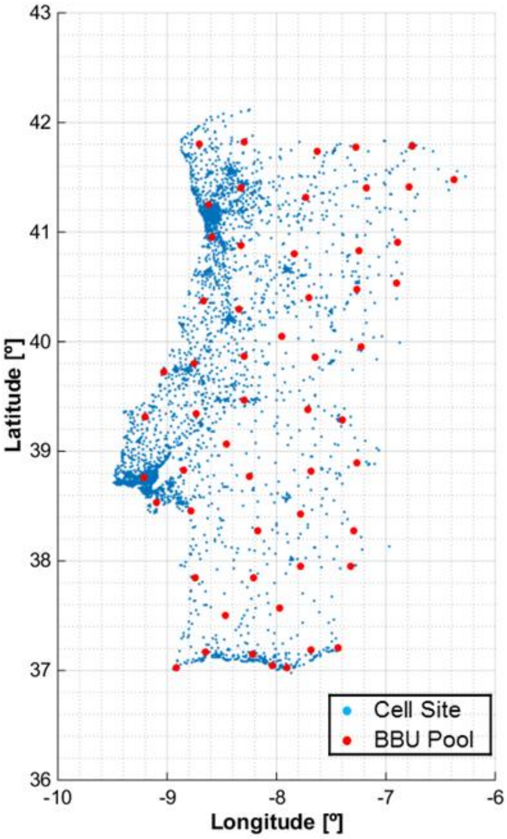

Figure 4.8. Placement of BBU Pools (in red) for a maximum fronthaul delay of 200 µs.

The first insight that one can extract from the previous figures is the expected decrease in the number of BBU Pools as the maximum fronthaul delay increases – Figure 4.9 depicts this evolution of the number of required BBU Pools, according to the positioning algorithm.

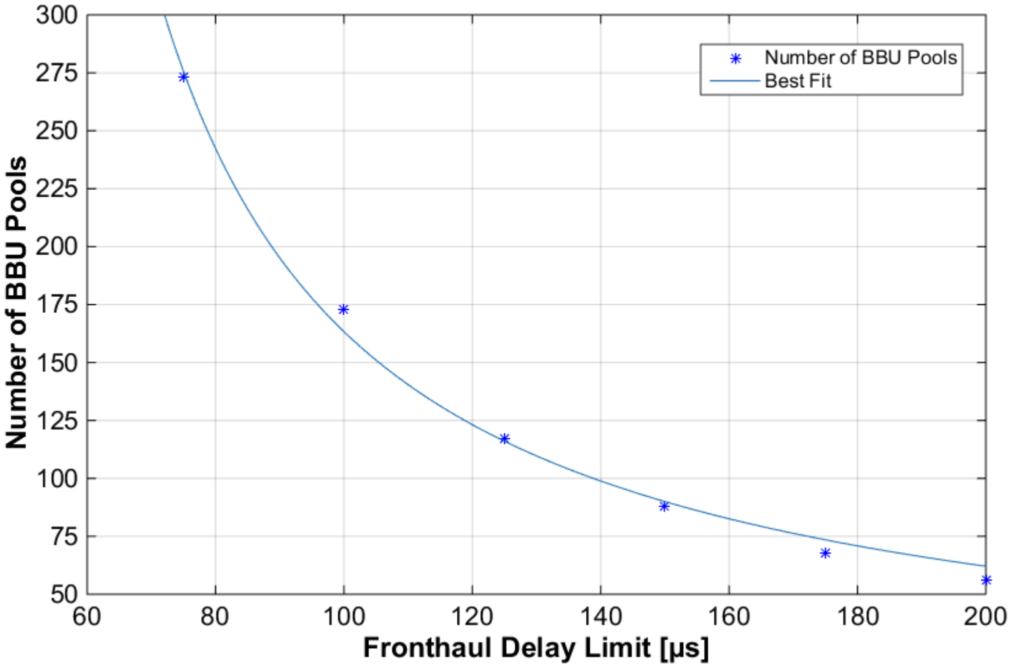

Figure 4.9. Number of BBU Pools vs. Maximum Fronthaul Delay, with best fit curve.

In fact, by increasing the maximum fronthaul delay, the maximum length of fronthaul links increases as well. This implicates an increase in the BBU Pool coverage area, which leads to a scenario where fewer BBU Pools are needed to cover the same total area – 56 for 200 μs as opposed to 273 for 75 μs. Here, a rational model is used for the fitting, Table 4.4. Although the quadratic and exponential models have similar correlation values in comparison with the rational one (higher than 0.98), the latter was chosen because, on the one hand, it monotonically decreases, which does not happen with the quadratic model, and on the other hand, the RMSE for the rational model is almost a half of the exponential model one.

Table 4.4. Mathematical characterisation of the best fit curve in Figure 4.9.

Model	Expression	R^2	RMSE
Rational	$N_{BBUPools} = \dfrac{10020}{\delta_{fh,\max\ [\mu s]} - 38.65}$	0.9948	6.574

Another important output is the number of single site BBU Pools, Table 4.5, which ideally would always be null in order to fully leverage the multiplexing gains brought by C-RAN, concerning for example energy efficiency. Hence, the algorithm was designed in order to decrease the percentage of single site BBU Pools.

Table 4.5. Number and Percentage of single site BBU Pools for different Maximum Fronthaul Delays.

Maximum Fronthaul Delay [μs]	75	100	125	150	175	200
Number of single site BBU Pools	32	14	7	3	3	4

The percentage of single site BBU Pools corresponding to Table 4.5 is depicted in Figure 4.10, where it can be seen that, due to the cell site and BBU Pool positioning algorithms, that percentage decreases down to 150 μs of maximum fronthaul delay, increasing for higher values of delay because the total number of BBU Pools decreases, while the absolute number of single site BBU Pools is approximately constant – around 3.

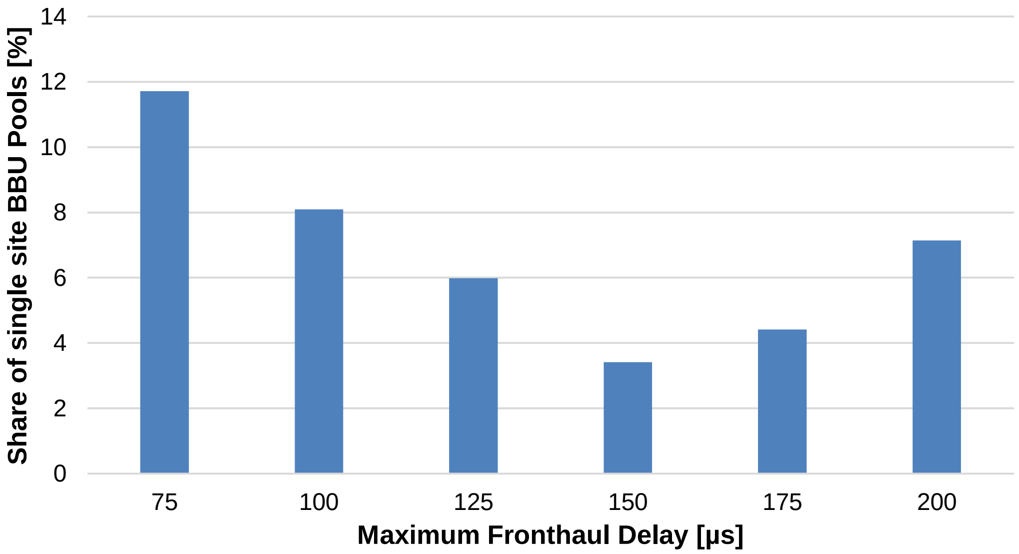

Figure 4.10. Percentage of single site BBU Pools vs. Maximum Fronthaul Delay.

The results in Table 4.5 show, to some extent, the efficiency of the algorithm, since only for a maximum fronthaul delay of 75 μs does the percentage of single site BBU Pools exceed 10% of the total number of BBU Pools. These results are also proof of the wide domain of application for the multiplexing capabilities provided by C-RAN.

As the imposed maximum fronthaul delay increases, a growth in the average fronthaul delay is also expected. Figure 4.11 confirms this conjecture. One can observe that the average delay (in blue) is significantly lower than the maximum delay bound, e.g., for a maximum delay of 200 μs the average one is nearly 80 μs. An increase in the standard deviation as the maximum delay increases is also noticeable (see red and green curves), which is coherent with the fact that BBU Pools provide connectivity to nearby sites as well as sites further away, causing a higher dispersion of delay values. Here, a linear model is used for the fitting in all three cases, Table 4.6. Although the model with higher correlation is the quadratic one, the linear model was chosen to fit data, because it makes more sense from a theoretical viewpoint – a quadratic approach would lead to a scenario where, if one decided to consider higher maximum fronthaul delays, the average fronthaul delay might even surpass the maximum fronthaul delay, which would not make sense.

Moreover, in Table 4.7, the maximum and minimum fronthaul delays for the range of imposed maximum fronthaul delay values are shown. Here, one can confirm that the maximum delay in the fronthaul is always less than the one being imposed. Also, one can observe that the minimum fronthaul delay is zero in all cases, which happens because, as seen before, there is always a percentage of single site BBU Pools, where the BBU Pool is co-located with the supported site, cancelling the fronthaul delay.

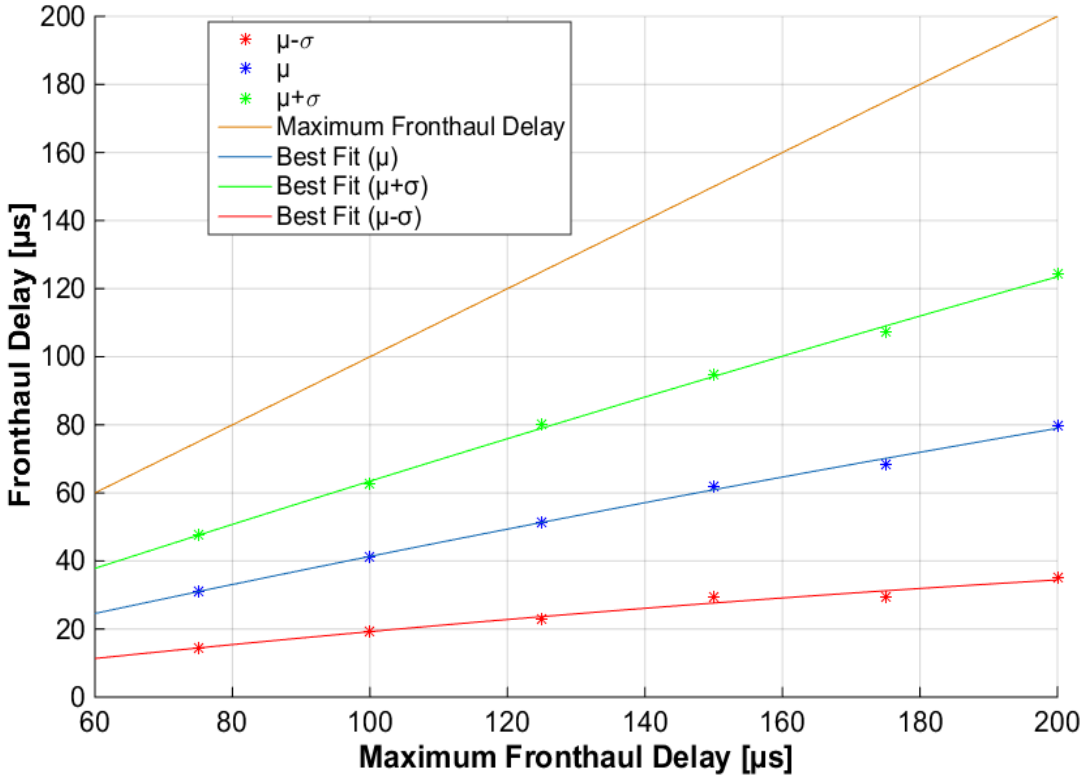

Figure 4.11. Fronthaul Delay vs. Maximum Fronthaul Delay, with best fit curves.

Table 4.6. Mathematical characterisation of the best fit curves in Figure 4.11.

Model	Fitted Data	Expression	R^2	RMSE
Linear	μ-σ	$\mu_{\delta_{fh}\,[\mu s]} - \sigma_{\delta_{fh}\,[\mu s]} = 0.16 \cdot \delta_{fh,\max\,[\mu s]} + 3.104$	0.9730	1.393
	μ	$\mu_{\delta_{fh}\,[\mu s]} = 0.3839 \cdot \delta_{fh,\max\,[\mu s]} + 2.822$	0.9960	1.278
	μ+σ	$\mu_{\delta_{fh}\,[\mu s]} + \sigma_{\delta_{fh}\,[\mu s]} = 0.6078 \cdot \delta_{fh,\max\,[\mu s]} + 2.54$	0.9982	1.339

Table 4.7. Maximum and minimum fronthaul delays for the range of imposed maximum delay values.

Maximum Fronthaul Delay (Imposed) [μs]	75	100	125	150	175	200
Maximum Fronthaul Delay [μs]	74.62	99.91	124.78	149.92	174.79	199.98
Minimum Fronthaul Delay [μs]	0					

Following the trend of average fronthaul delay linear growth with the increase of the maximum fronthaul delay, the restriction is the average fronthaul distance as a function of the maximum fronthaul distance. The increase in standard deviation with the maximum fronthaul distance is also noticeable, which again can be explained by the increase in BBU Pool coverage areas, leading to a situation where cell sites are spreader within those areas. Figure 4.12 illustrates these fronthaul distance results. Again, a linear model is used for the fitting in all three cases, Table 4.8.

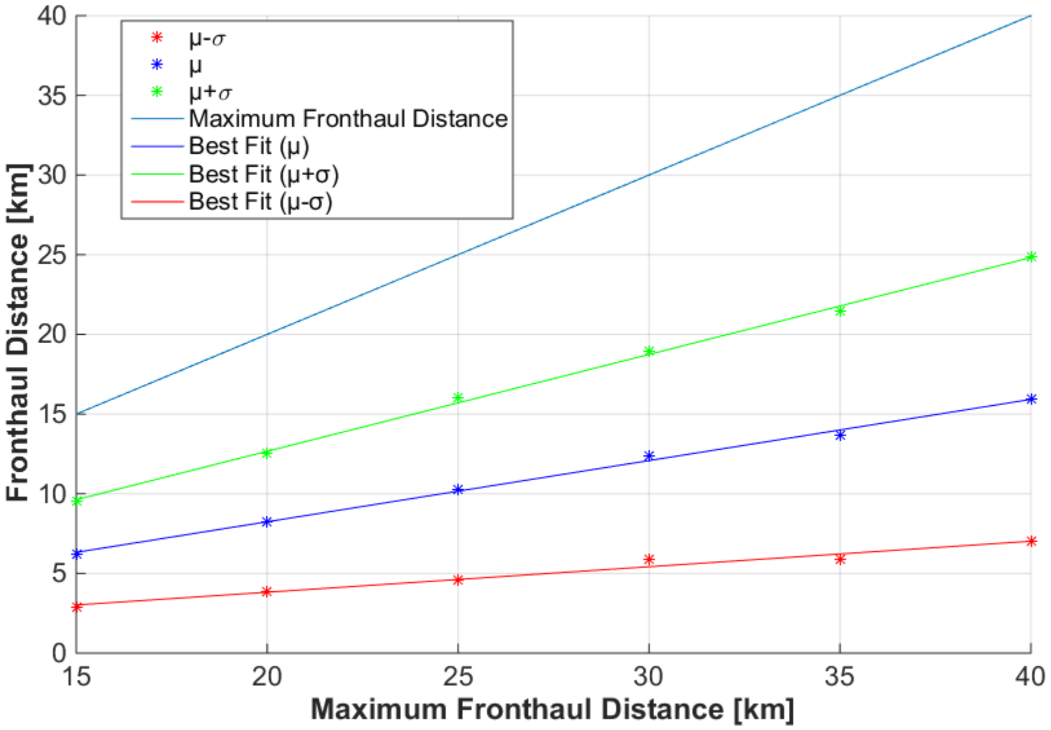

Figure 4.12. Fronthaul Distance vs. Maximum Fronthaul Distance, with best fit curves.

Furthermore, in Table 4.9., the maximum and minimum fronthaul distances for the range of imposed maximum fronthaul distance values are shown. Here, one can confirm that the maximum distance in the fronthaul is always less than the one being imposed. Also, one can observe that the minimum fronthaul distance is zero in all cases, which happens because, as seen before, there is always a percentage of single site BBU Pools.

Table 4.8. Mathematical characterisation of the best fit curves in Figure 4.12.

Model	Fitted Data	Expression	R^2	RMSE
Linear	μ-σ	$\mu_{d_{fh}\,[\text{km}]} - \sigma_{d_{fh}\,[\text{km}]} = 0.16 \cdot d_{fh,\max\,[\text{km}]} + 0.6208$	0.9730	0.2786
	μ	$\mu_{d_{fh}\,[\text{km}]} = 0.3839 \cdot d_{fh,\max\,[\text{km}]} + 0.5644$	0.9960	0.2556
	μ+σ	$\mu_{d_{fh}\,[\text{km}]} + \sigma_{d_{fh}\,[\text{km}]} = 0.6078 \cdot d_{fh,\max\,[\text{km}]} + 0.5079$	0.9982	0.2678

Table 4.9. Maximum and minimum fronthaul distances for the range of maximum distance values.

Maximum Fronthaul Distance (Imposed) [km]	15	20	25	30	35	40
Maximum Fronthaul Distance [km]	14.925	19.982	24.95	29.984	34.959	39.997
Minimum Fronthaul Distance [km]	0					

Another important insight that can be extracted from the RAN positioning results is how the cell sites are distributed within the coverage area of Rural, Suburban and Urban BBU Pools. Figure 4.13 and Figure 4.14 illustrate the positioning of BBUs from the three classes.

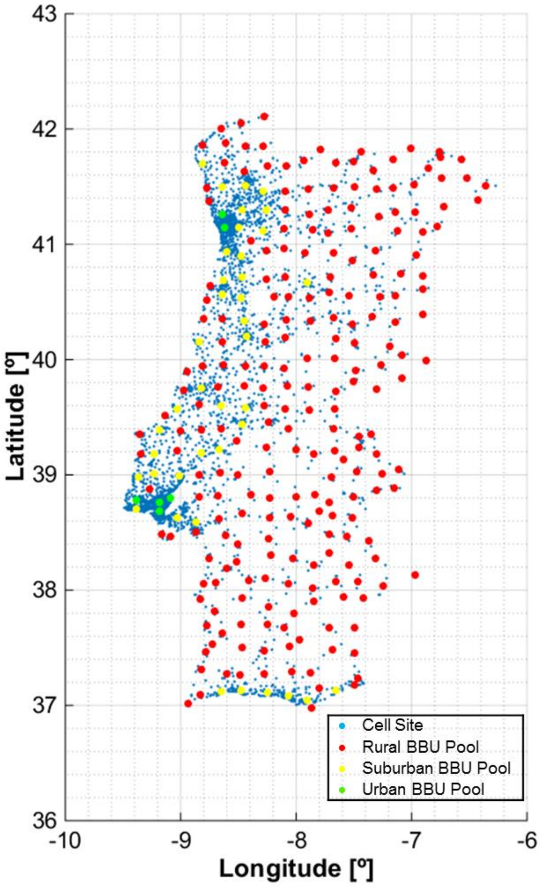

Figure 4.13. Classes of BBUs and their positioning, for 75 μs of maximum fronthaul delay.

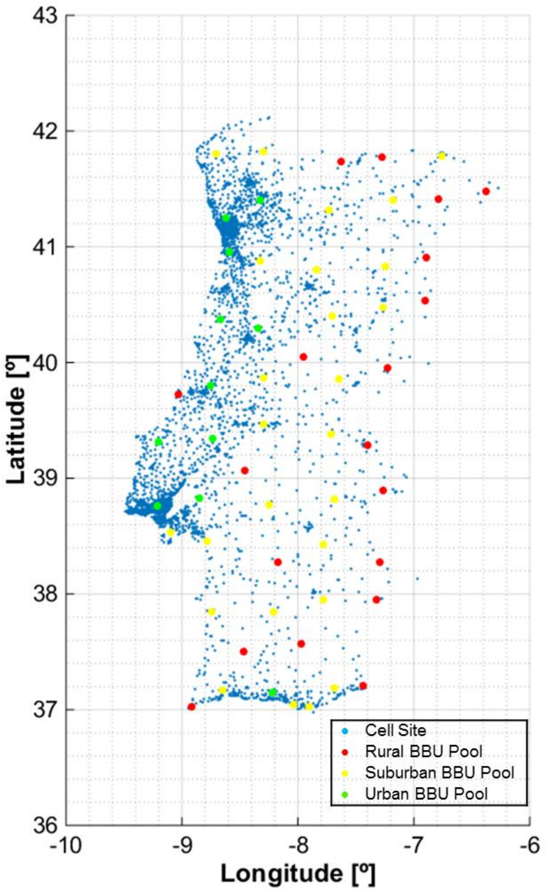

Figure 4.14. Classes of BBUs and their positioning, for 200 µs of maximum fronthaul delay.

The criterion used for classifying BBUs in the three classes is: Rural, with less than 20 cell sites within its coverage area; Suburban, with less than or equal to 100 sites; Urban, with more than 100 sites. Figure 4.15, Figure 4.16 and Figure 4.17, along with Figure 4.18, Figure 4.19 and Figure 4.20 depict, respectively, how the cell sites are spread within BBU Pool coverage areas for a maximum fronthaul delay of 75 µs and 200 µs, with 1 km interval – for 75 µs the fronthaul distance goes up to 15 km, and for 200 µs it goes up to 40 km. For Rural BBU pools, it is observed that the share of cell sites is distributed by distance in a nearly equitable way, only with smaller shares for distances superior to 10 km for the 75 µs case; this scenario was expected, because cell sites are more geographically scattered in rural areas. In the 200 µs case, this effect is less obvious, because of the small 1 km resolution that causes a broader scattering of the share of cell sites among the fronthaul distance categories. For Suburban BBU Pools, the share of cell sites distribution by distance resembles a Gaussian shaped curve, with bigger percentages for middle distances and gradually decreasing to smaller percentages for shorter and longer distances; this scenario indicates a decrease in the geographical scattering of cell sites, which was expected in comparison to the Rural case. For Urban BBU Pools, the larger shares of cell sites appear closer to the BBU Pool and not so much in middle distances like in the suburban case, which can be explained by the fact that in an urban scenario the cell sites tend to be more and more concentrated, as the number of users and corresponding traffic demand gets higher.

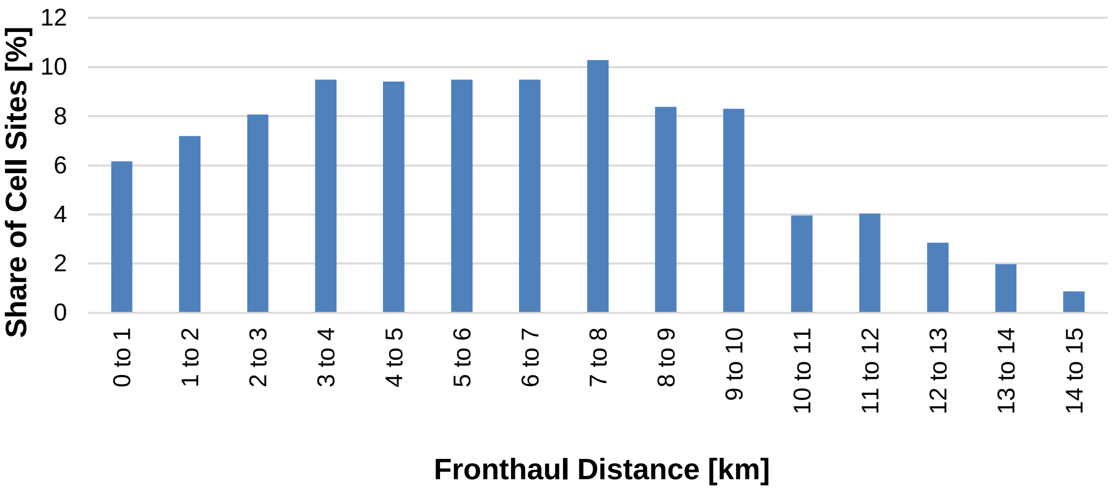

Figure 4.15. Percentage of cell sites at different fronthaul distances (1 km interval), for Rural BBU Pools and 75 µs of maximum fronthaul delay.

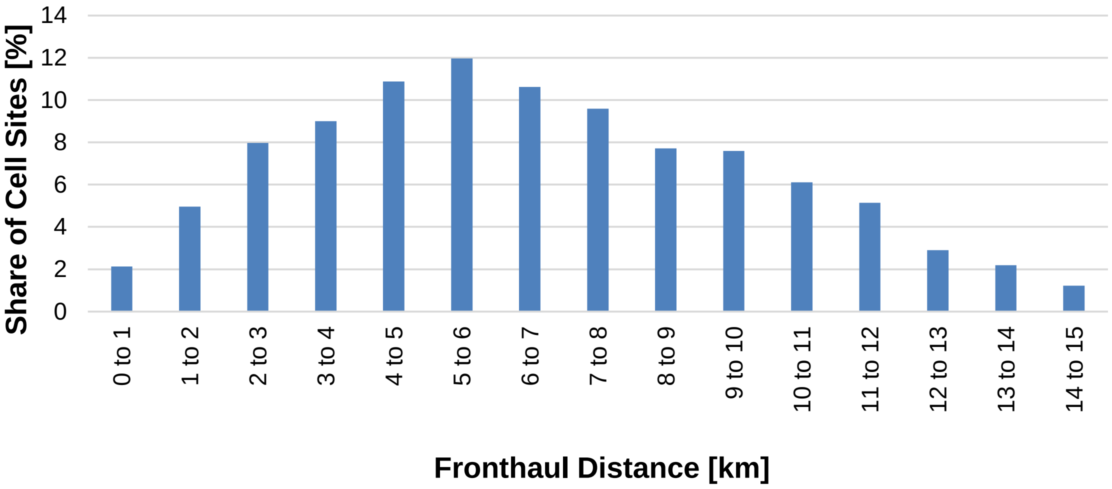

Figure 4.16. Percentage of cell sites at different fronthaul distances (1 km interval), for Suburban BBU Pools and 75 µs of maximum fronthaul delay.

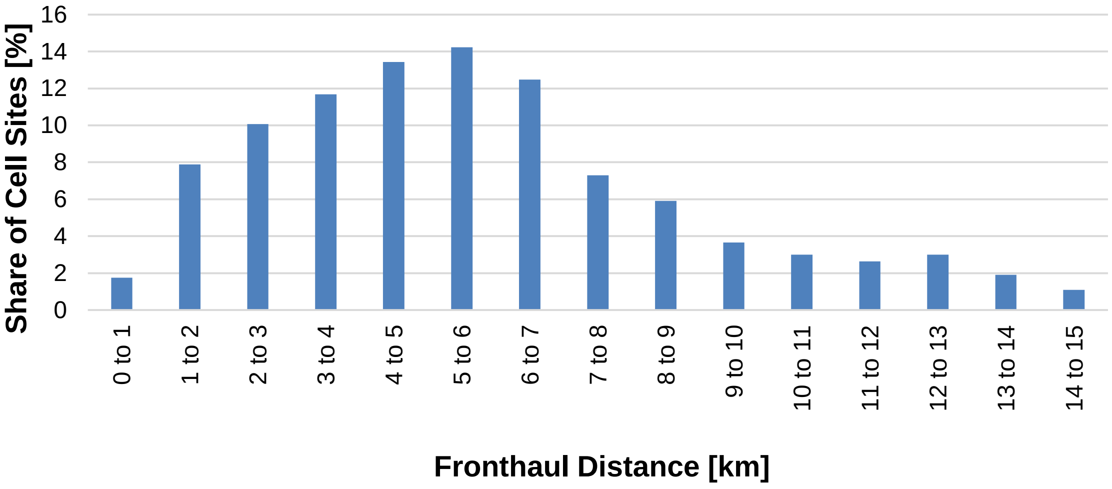

Figure 4.17. Percentage of cell sites at different fronthaul distances (1 km interval), for Urban BBU Pools and 75 µs of maximum fronthaul delay.

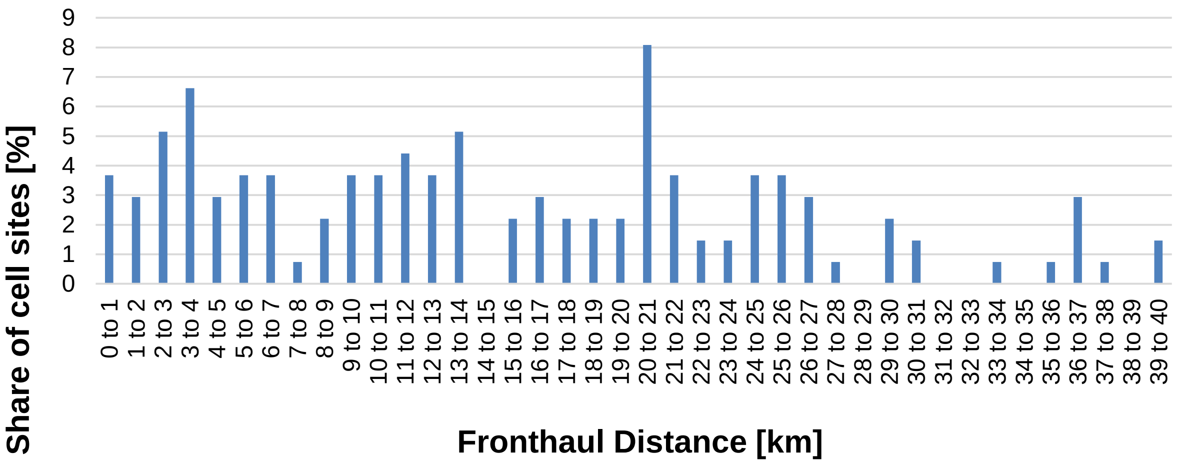

Figure 4.18. Percentage of cell sites at different fronthaul distances (1 km interval), for Rural BBU Pools and 200 µs of maximum fronthaul delay.

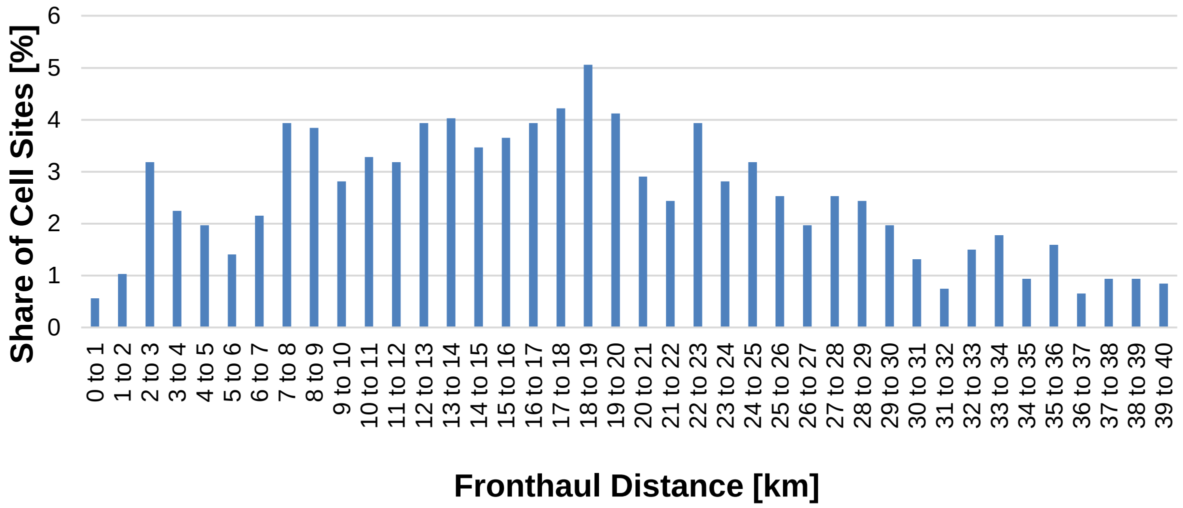

Figure 4.19. Percentage of cell sites at different fronthaul distances (1 km interval), for Rural BBU Pools and 200 µs of maximum fronthaul delay.

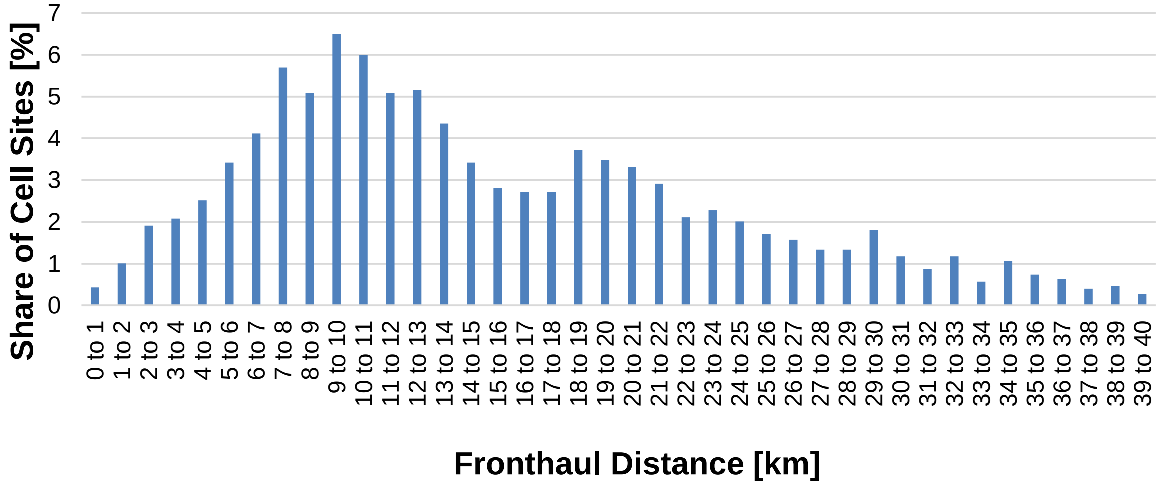

Figure 4.20. Percentage of cell sites at different fronthaul distances (1 km interval), for Urban BBU Pools and 200 µs of maximum fronthaul delay.

Increasing the maximum fronthaul delay is also expected to cause a decrease in the number of possible microwave links – an option that should be regarded in places where fibre links do not exist and the fronthaul distance is less than or equal to 2 km. In fact, Figure 4.21 shows that, as the maximum delay increases from 75 μs to 200 μs, the percentage of possible microwave links decreases from 9.8% to 1.6%. Here, a rational model is used for the fitting, Table 4.10, since it was the one with the highest correlation and lower RMSE. Only the transition from 150 μs to 175 μs is contrary to the descending trend, which is understandable, since although a higher maximum delay is imposed, it is not guaranteed that it will lead to less cell sites placed 2 km or less away from the corresponding BBU Pool – it depends on their spatial distribution. If one considers that microwave technology could only support the traffic demand of the considered cell sites for shorter distances (e.g., 1 km or less), it would cause these percentages to be smaller, but the curve should exhibit a similar behaviour.

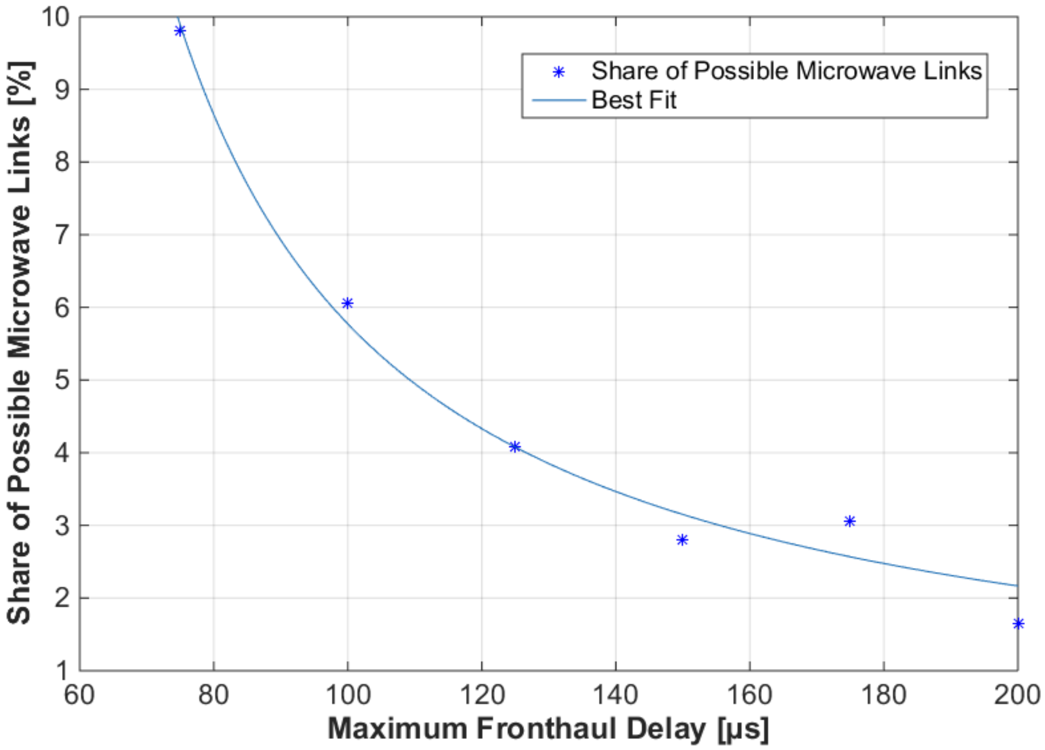

Figure 4.21. Possible microwave links vs. Maximum Fronthaul Delay, with best fit curve.

Table 4.10. Mathematical characterisation of the best fit curve in Figure 4.21.

Model	Expression	R^2	RMSE
Rational	$N_{mwlinks\,[\%]} = 100 \cdot \frac{N_{mwlinks}}{N_{fhlinks}} = \frac{346.9}{\delta_{fh,\max\,[\mu s]} - 39.91}$	0.9835	0.4262

Figure 4.22 and Figure 4.23 depict respectively the evolution of the number of sites per BBU Pool and number of blade servers per BBU Pool, as the maximum fronthaul delay is increased. A very similar behaviour is observed in both cases, with an increasing trend of the average value for both quantities as well as of the standard deviation. This situation was expected, since as the maximum delay increases, the coverage areas of BBU Pools increase as well, causing these nodes to provide connectivity for more cell sites, which consequently origins a higher number of blade servers. Hence,

one can easily perceive that the number of blade servers required in a BBU Pool is deeply related to the number of sites that need to be supported. Quadratic models were used for the fitting of these four data sets, Table 4.11 and Table 4.12, since a higher correlation and a lower RMSE were verified, in comparison with the remaining mathematical models considered.

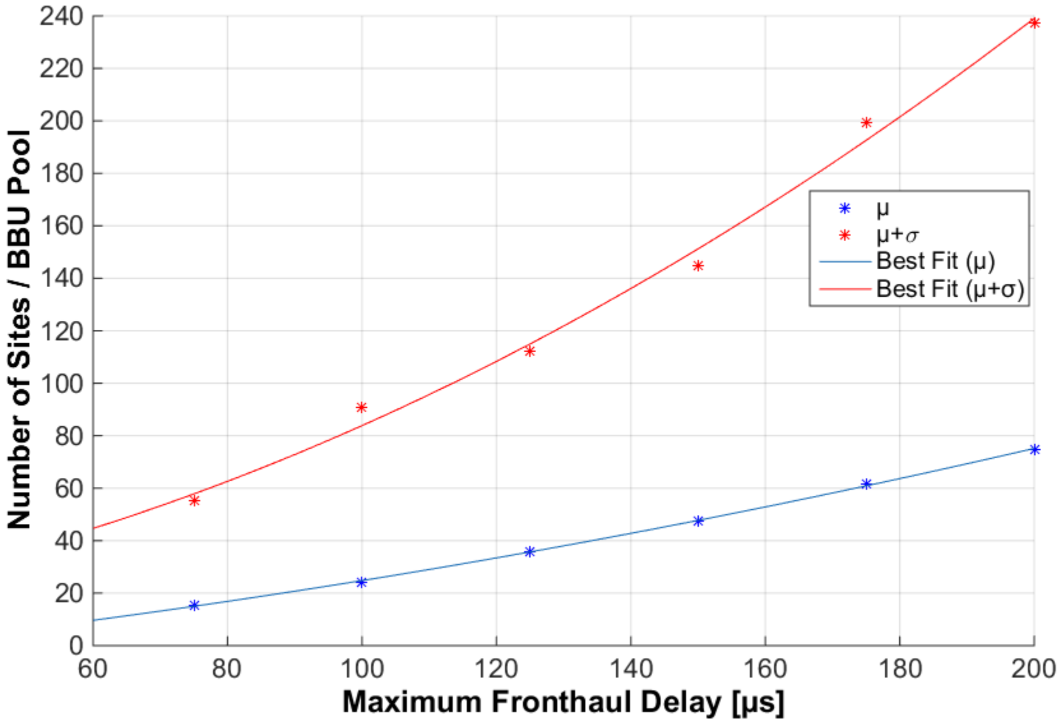

Figure 4.22 Number of Sites per BBU Pool vs Maximum Fronthaul Delay, with best fit curves.

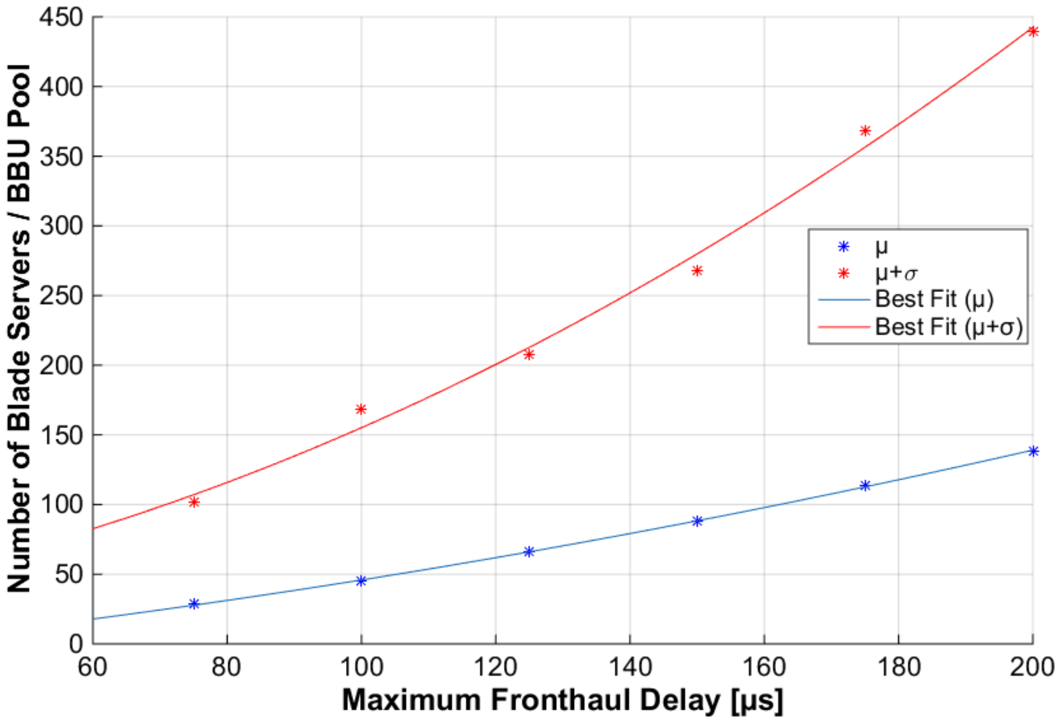

Figure 4.23 Number of Servers per BBU Pool vs. Maximum Fronthaul Delay, with best fit curves.

Table 4.11. Mathematical characterisation of the best fit curve in Figure 4.22.

Model	Fitted Data	Expression	R^2	RMSE
Quadratic	μ	$\mu_{N_{Sites/BBUPool}} =$ $= 8.941 \times 10^{-4} \cdot \delta_{fh,\max\ [\mu s]}{}^2 + 0.2355 \cdot \delta_{fh,\max\ [\mu s]} - 7.691$	0.9996	0.610
	μ+σ	$\mu_{N_{Sites/BBUPool}} + \sigma_{N_{Sites/BBUPool}} =$ $= 4.093 \times 10^{-3} \cdot \delta_{fh,\max\ [\mu s]}{}^2 + 0.3245 \cdot \delta_{fh,\max\ [\mu s]} + 10.48$	0.9934	7.186

Table 4.12. Mathematical characterisation of the best fit curve in Figure 4.23.

Model	Fitted Data	Expression	R^2	RMSE
Quadratic	μ	$\mu_{N_{Servers/BBUPool}} =$ $= 1.655 \times 10^{-3} \cdot \delta_{fh,\max\ [\mu s]}{}^2 + 0.4357 \cdot \delta_{fh,\max\ [\mu s]} - 14.23$	0.9995	1.129
	μ+σ	$\mu_{N_{Servers/BBUPool}} + \sigma_{N_{Servers/BBUPool}} =$ $= 7.574 \times 10^{-3} \cdot \delta_{fh,\max\ [\mu s]}{}^2 + 0.6005 \cdot \delta_{fh,\max\ [\mu s]} + 19.4$	0.9934	13.30

Furthermore, Table 4.13 and Table 4.14 show the maximum and minimum number of sites per BBU pool and servers per BBU pool, respectively, for the regarded set of maximum fronthaul delay values. In Table 4.13, the minimum number of sites per BBU Pool is consistent with the previous results that show a percentage of single site BBU Pools always higher than 0%. Also, the minimum number of servers per BBU Pool presented in Table 4.14 was already expected, because of the following factors: the minimum number of sites per BBU Pool is 1; all sites are hereby considered to be class 1 (18.506 Gbps of fronthaul link capacity); every blade server is hereby regarded as possessing 10 Gbps capacity.

Table 4.13. Maximum and minimum Number of Sites per Pool for the range of imposed Maximum Fronthaul Delay values.

Maximum Fronthaul Delay [μs]	75	100	125	150	175	200
Maximum Number of Sites per Pool	404	693	507	606	1 039	1 099
Minimum Number of Sites per Pool	1					

Table 4.14. Maximum and minimum Number of Blade Servers per BBU Pool for the range of imposed Maximum Fronthaul Delay values.

Maximum Fronthaul Delay [μs]	75	100	125	150	175	200
Maximum Number of Servers per Pool	748	1 283	939	1 122	1 923	2 034
Minimum Number of Servers per Pool	2					

4.2.2 Blade Server Capacity Impact

As pointed in Section 4.1, henceforward three classes of sites are considered in order to aggravate the scenario in terms of capacity, since adding sites classes 2 and 3, Figure 4.2 and Table 4.1, implicates a higher capacity requirements for the fronthaul, which in turn require higher capacities in the links and nodes upstream. The positioning used for the study made in this section is for a maximum fronthaul delay of 200 µs. The evolution of the number of blade servers per BBU pool with the enhancement of blade server capacity is depicted in Figure 4.24. The symbols µ and σ in the following plot refer to the average and the standard deviation, respectively.

With the increase of blade server capacity from 10 Gbps to 60 Gbps, the average number of blade servers needed decreases accordingly. In fact, the fitting shown for both curves is for a rational model with R^2 = 1 and RMSE equal to 0, Table 4.15, which proves that the two quantities are inversely proportional – this confirms what was intuitively expected. Also, it is observed that the standard deviation decreases as the blade server capacity increases, which is a reasonable result, since the latter leads to a significant decrease in the number of blade servers especially in the most loaded BBU Pools.

Moreover, Table 4.16 shows the maximum and minimum numbers of Blade Servers per BBU Pool for the set of Blade Server Capacity values in the plot, where one can see that for a Blade Server Capacity equal to or higher than 20 Gbps, the number of required Blade Servers can be as low as one single server.

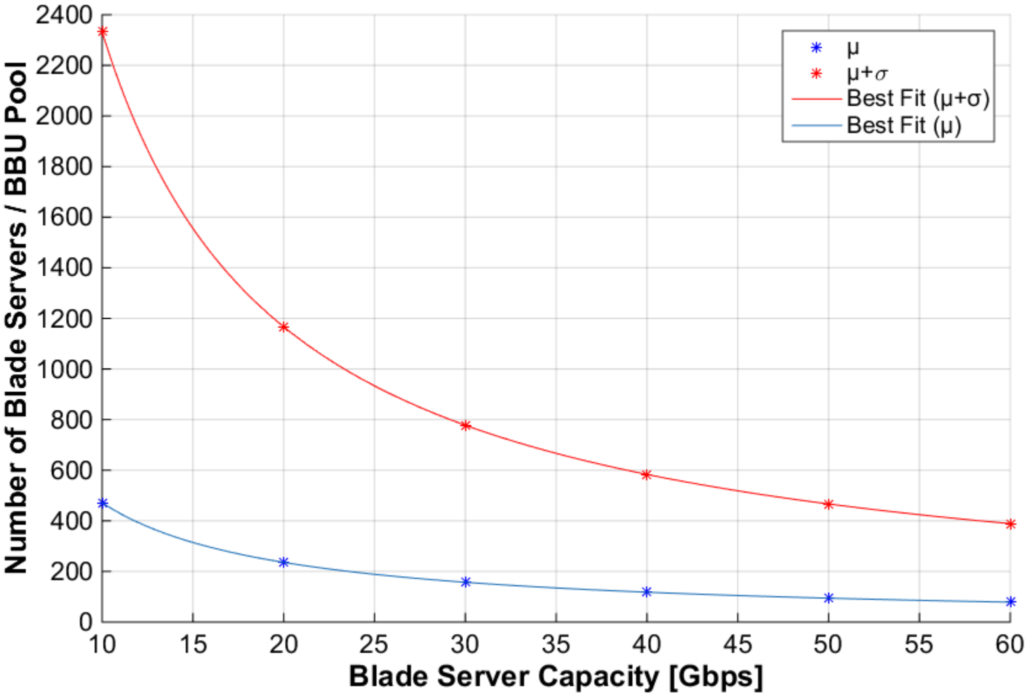

Figure 4.24. Number of Blade Servers per BBU Pool vs Blade Server Capacity, with best fit curves.

It should be noticed that for a Blade Server Capacity of 10 Gbps and a positioning associated with 200 µs of maximum fronthaul delay (both values seen in the previous section), here the average number of blade servers (472) as well as the maximum number of servers per BBU Pool (12 608) are significantly higher than the ones obtained in the previous section – 139 and 2 034, respectively. This is of course due to the introduction of more capacity demanding sites (classes 2 and 3), which highlights the need for scalable pooling resources, in order to cope with increasing data capacity demands.

Table 4.15. Mathematical characterisation of the best fit curve in Figure 4.11.

Model	Fitted Data	Expression	R^2	RMSE
Rational	µ	$\mu_{N_{Servers/BBUPool}} = \frac{4\,717}{C_{Server\,[\mathrm{Gbps}]}}$	1	0
	µ+σ	$\mu_{N_{Servers/BBUPool}} + \sigma_{N_{Servers/BBUPool}} = \frac{23\,340}{C_{Server\,[\mathrm{Gbps}]}}$	1	0

Table 4.16. Maximum and minimum Number of Blade Servers per BBU Pool for the range of Blade Server Capacity values.

Blade Server Capacity [Gbps]	10	20	30	40	50	60
Maximum Number of Servers per Pool	12 608	6 304	4 203	3 152	2 522	2 102
Minimum Number of Servers per Pool	2	1				

4.3 Core Analysis

4.3.1 Maximum Backhaul Delay Impact

A similar simulation to the one concerning the placement of BBU Pools for different maximum fronthaul delay values is made in this section – here, the goal is the placement of Core Nodes for different backhaul delay values (2 ms, 4 ms, 6 ms, 8 ms, 10 ms and 12 ms). The BBU Pools' positioning used is the one obtained for a maximum fronthaul delay of 200 µs.

The total number of Core Nodes is 2 for a maximum backhaul delay of 2 ms – for the remaining values, a single Core Node is enough to cover the entire country, from a delay restriction perspective. As a matter of fact, for resilience purposes, extra nodes must be present to provide redundancy, in a way that information and connectivity is not lost in case of failure of the original nodes. For 2 ms, a single Core Node is not enough, because the links connecting it to the three southernmost BBU Pools would not fulfil the referred backhaul delay constraint – the remaining values represent a lighter delay constraint, hence, for these there is no need for more Core Nodes. Figure 4.25 and Figure 4.26 show how the algorithm placed the Core Nodes (in green) countrywide for 2 ms and for 4 ms, 6 ms, 8 ms, 10 ms and 12 ms, respectively, along with the previously placed BBU Pools (in red) for a maximum fronthaul delay of 200 µs.

Regarding the average backhaul delay, it increases from approximately 781 µs in a scenario with 2 Core Nodes to 845 µs in a scenario with a single Core Node. Hence, in either case, the average backhaul delay remains significantly below the imposed constraint, which is 2 ms in the first case and in between 4 ms and 12 ms in the remaining ones. In what concerns the maximum and minimum backhaul delays that were verified for the set of imposed maximum backhaul delay values, the minimum was 142 μs for a maximum backhaul delay of 2 ms and 165 μs for the remaining values, while the maximum was 1.41 ms for a maximum backhaul delay of 2 ms and 1.46 ms for the remaining values. Translating this situation into distance, the average backhaul distance increases from approximately 156 km with 2 Core Nodes to 169 km with a single Core Node. In what concerns the maximum and minimum backhaul distances that were verified for the set of imposed maximum backhaul delay values, the minimum was 28 km for a maximum backhaul delay of 2 ms and 33 km for the remaining values, while the maximum was 283 km for a maximum backhaul delay of 2 ms and 292 km for the remaining values.

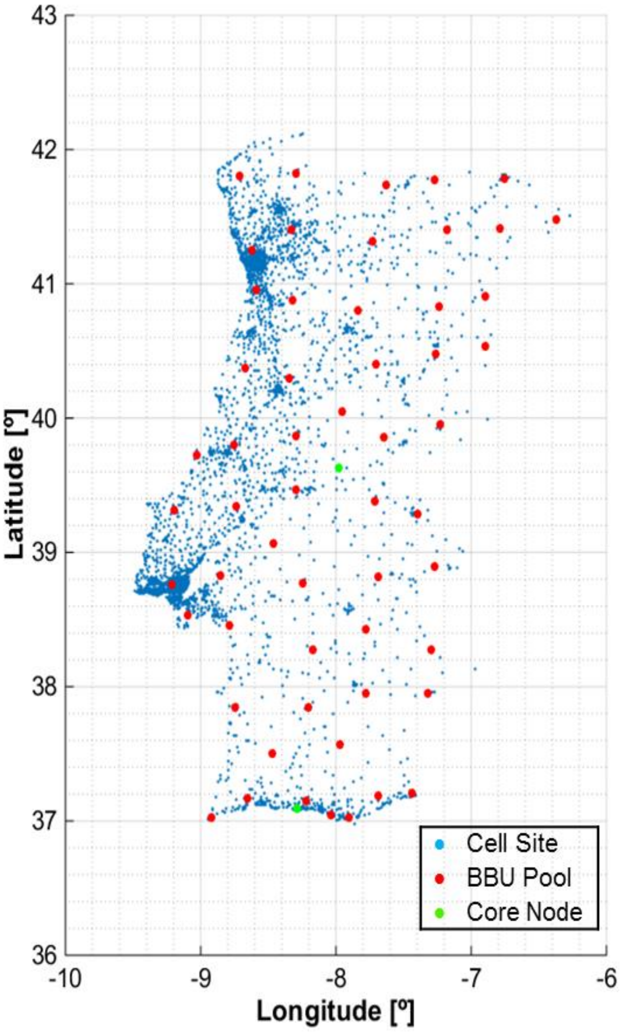

Figure 4.25. Placement of Core Nodes for a maximum backhaul delay of 2 ms.

For the situation with only one Core Node, all 56 BBU Pools stand connected to that node, requiring it to have at least 5 812 blade servers if one considers a blade server capacity of 50 Gbps. On the other hand, when two Core Nodes are at stake, and again for a blade server capacity of 50 Gbps, since the node in Algarve is allocated particularly to the 3 southernmost BBU Pools, it can have a minimum of 41 blade servers, while the other node must have at least 5 771 blade servers in order to support the remaining 53 BBU Pools.

It must be highlighted that none of the maximum backhaul delay values imposed in simulation caused a Core Node to support a single BBU Pool. That would lead to an inefficient usage of resources, since an absence of resource sharing would result in no pooling gains.

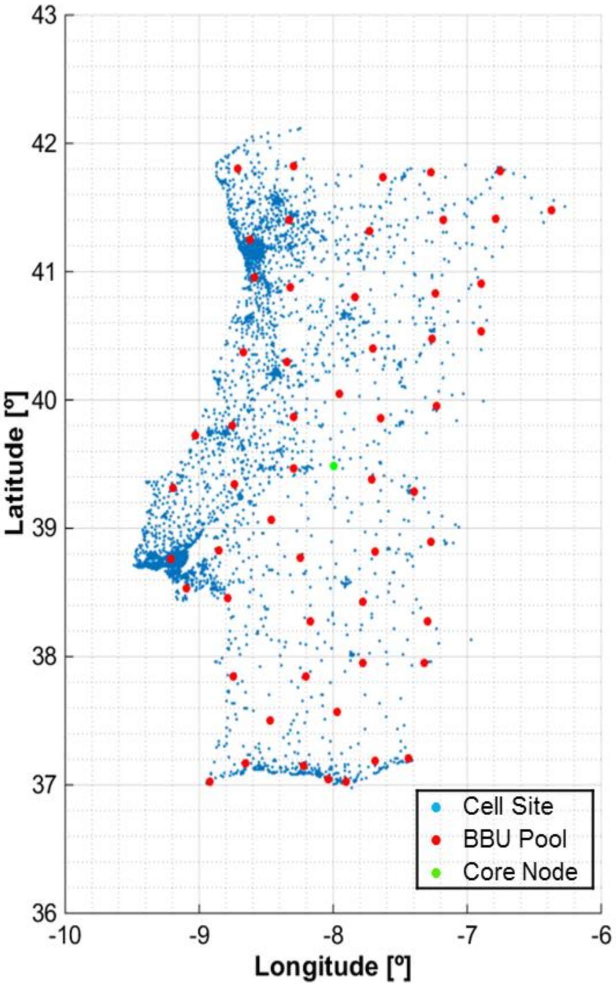

Figure 4.26. Placement of Core Nodes for a maximum backhaul delay of 4 ms, 6 ms, 8 ms, 10 ms and 12 ms.

4.3.2 Blade Server Capacity Impact

For evaluating how the number of blade servers per Core Node evolves as the blade server capacity is changed, it was chosen the case where a single Core Node exists, Figure 4.26. The behaviour observed in Figure 4.27 is identical to the one observed in Figure 4.24.

With the increase of blade server capacity from 50 Gbps to 100 Gbps, the number of blade servers needed in the Core Node decreases accordingly – from almost 5 812 blade servers to 2 906. In fact, the fitting represents a rational model with R^2 = 1 and a nearly null RMSE, Table 4.17, which proves that the two quantities are inversely proportional – this confirms what was intuitively expected.

Table 4.17. Mathematical characterisation of the best fit curve in Figure 4.27.

Model	Expression	R^2	RMSE
Rational	$N_{Servers/CoreNode} = \frac{290\ 600}{C_{Server\ [\mathrm{Gbps}]}}$	1	0

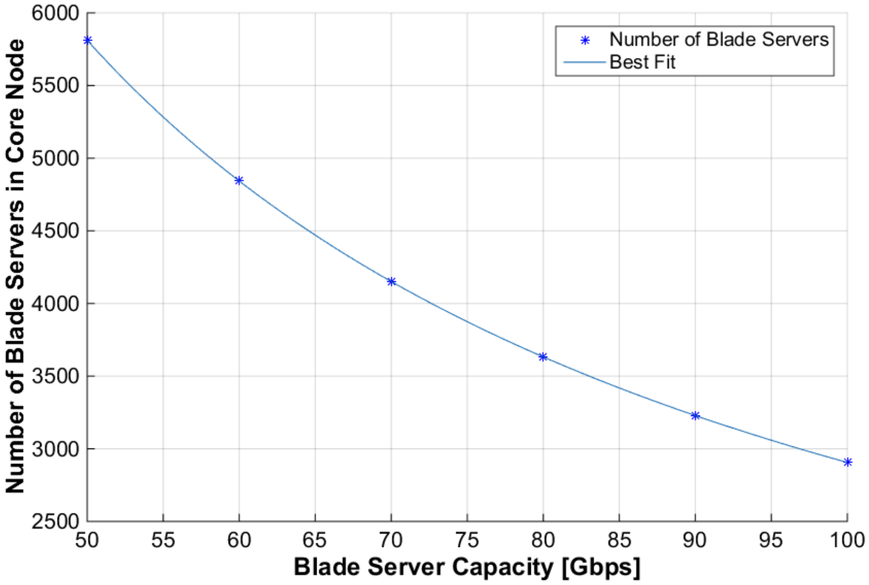

Figure 4.27. Number of Blade Servers per Core Node vs. Blade Server Capacity, with best fit curves.

4.4 End-to-End Delay Evaluation

The results from the implementation of the algorithm described in Chapter 3 were illustrated and discussed in previous sections. However, a broader view of the delay scenario adds value to the present work, hence, in this section one shows and discusses a set of estimates of the end-to-end delay in between Portugal and servers located in different regions of the world. In fact, while for non-interactive services the placement of the server is more or less irrelevant, for interactive services this placement can be critically important for guaranteeing a good QoE. The delay values regarded for the RAN, Backhaul, EPC and Server are fixed and presented in Table 4.18. Adding to these values is the delay due to the introduction of NSC whose impact is analysed herein.

The impact of NSC in terms of delay has just been studied in laboratory deployments, since very few commercial implementations are available. Hence, at this stage it is still not clear what values can be reasonable for wider scenarios. Anyway, for the purpose of this work, the delay due to NSC is considered to be 4ms. The server location is one of the factors that fundamentally determines backbone latency, along with congestion phenomena and the different network policies over the path. Two different sources were considered in order to estimate backbone latency values: [ClPi15] and [TeGe15].

Table 4.18. Fixed delay values contributing for the global End-to-end Delay.

Delay	Value [ms]
δ_{RAN}	10
δ_{bh}	12
δ_{EPC}	4
δ_{Server}	4
Total	30

First, the values in [TeGe15] were adapted, since they were associated with latency from the United Kingdom to the rest of the world, while in this thesis what matters is the delay from Portugal to the rest of the world. Then, the values from both sources were compared, and it followed that [TeGe15] was adopting an overly optimistic approach, hence, its values were raised by 30% for this study. After that, the server locations present in [ClPi15] that were missing in [TeGe15] have been added to the previously obtained set of values and adjusted in order to establish a coherent set of backbone latency estimates – Table 4.19 shows these estimates.

Table 4.19. Backbone RTT estimates, from Portugal to various world zones.

Server Location		$\boldsymbol{\delta_{bb}}$ [ms]
Africa	Tunisia	45.5
	South Africa	149.5
Asia	China	234
	India	143
	Japan	279.5
Europe	Germany	19.5
	United Kingdom	32.5
Middle East	United Arab Emirates	123.5
North America	USA (East Coast)	110.5
	USA (West Coast)	201.5
Oceania	Australia	266.5
South America	Brazil	221

Figure 4.28 depicts the global end-to-end delay, from a terminal located in Portugal to servers located in the previously referred world regions, with each of its components represented by a different colour. In Figure 4.28, one also presents 3 benchmarking lines that allow an assessment of where various kinds of services could be hosted, in order to fulfil widely adopted latency standards.

The first clear finding from the obtained absolute results is the dominance of backbone latency, in yellow, among the different components that constitute the end-to-end RTT. It should be noted once again that the NSC, processing in the Server, EPC, Backhaul and RAN delays are regarded as equal for every server location, which in reality is not very accurate, since it depends on the policies and infrastructure of different operators along the path between user terminal and server. Apart from backbone latency, the more relevant components for the end-to-end delay are RAN (10 ms, in light blue) and Backhaul (12 ms, in orange) delays, followed by EPC (4 ms, in grey), Server (4 ms, in dark blue) and NSC (4 ms, in green) delays.

The second important insight that can be extracted from the previous chart is the drastic difference in end-to-end delay between situations where the server is in Europe and servers located in other parts of the world – for Germany and United Kingdom the end-to-end delay is of 53.5 ms and 66.5 ms, respectively. The only value that can be comparable to the previous two is the end-to-end delay to Tunisia (79.5 ms), since it is the closest place to Portugal from the set of chosen locations. Hence, these locations would be the more suitable for hosting services supporting interactive applications.

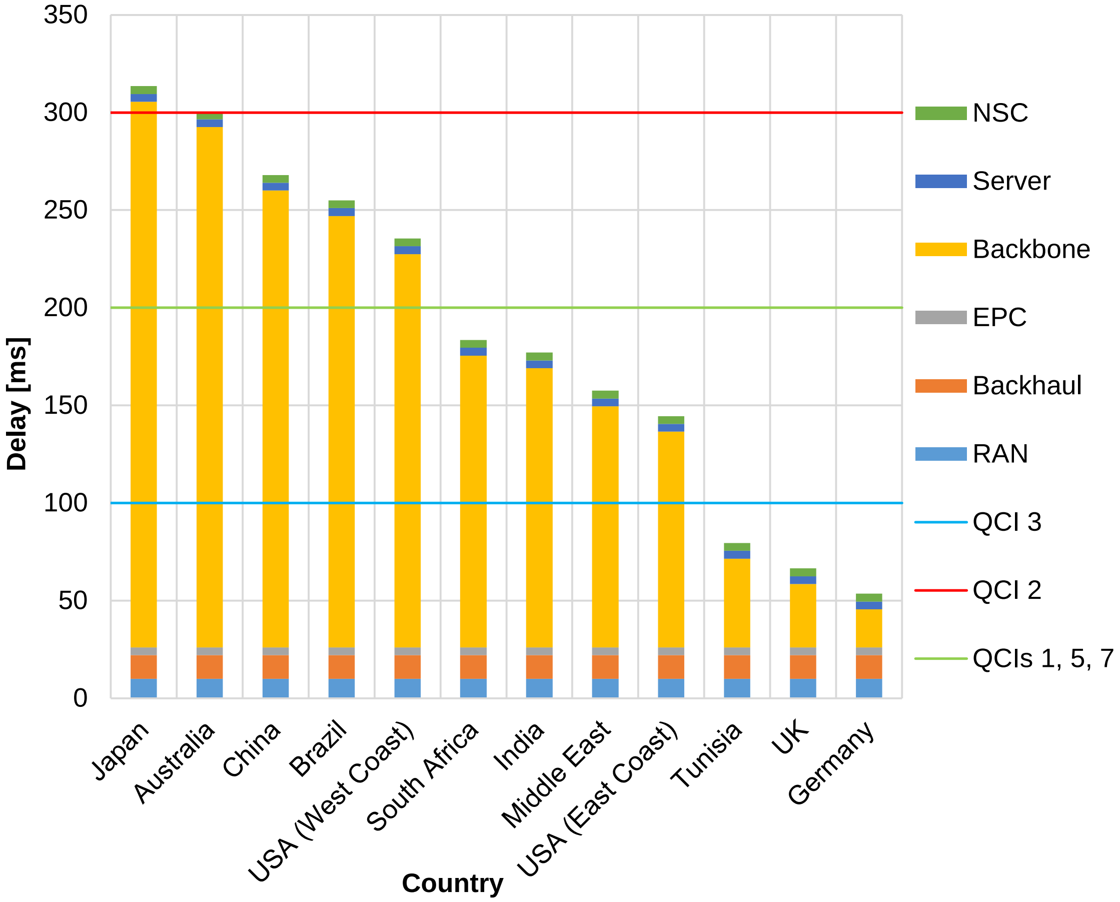

Figure 4.28. End-to-end RTT from Portugal to the World and its components (absolute values).

Next to these values are the ones for USA's East Coast (144.5 ms) and Middle East (157.5 ms), and then India (177 ms) and South Africa (183.5 ms). Lastly, the five server locations that cause a higher latency are, in this order, USA's West Coast (235.5 ms), Brazil (255 ms), China (268 ms), Australia (300.5 ms) and Japan (313.5 ms). These results reinforce the idea that interactive services and applications must be pushed as much as possible to the mobile network's edge, otherwise the QoE might be compromised. An important remark about the end-to-end delay to Brazil should be done: the fact that it is significantly higher than the delay to USA's East Coast is because nearly all submarine cables ensuring communication between Brazil and countries to the east, such as Portugal, are firstly attached in USA's East Coast before continuing to Brazil. There is already a plan to deploy at least one submarine cable directly between Portugal and Brazil, but currently it is still not in operation.

In what concerns the hosting of different types of data services, in Figure 4.28 is seen that QCI 3 (light blue line) services like Real Time Gaming should be located in areas relatively close to the end user in Portugal, like Germany, UK or Tunisia, since these are the most delay-sensitive of all. Following QCI 3 services in terms of delay sensitivity are services of QCIs 1, 5 and 7 (green line), e.g., Conversational Voice, IMS Signalling and Video (Live Streaming), which could be hosted in a vaster area including

South Africa, India, Middle East and USA (East Coast). Then, QCI 2 services (red line) like Conversational Video could be hosted in an even more extensive area, basically just excluding Japan. Finally, there are no lines representing services of QCIs 4, 6, 8 and 9 which include for example Non-Conversational Video and TCP-based applications, because due to its non-interactive nature and consequent delay insensitivity these could be hosted indifferently in any of the world zones. Figure 4.29 depicts the percentage of each end-to-end delay component for the set of server placements considered. Once again, the dominance of backbone latency (yellow segment) in the end-to-end RTT is clear – Table 4.20 explicitly illustrates this dominance, showing that in most cases (9 out of 12) the backbone delay is responsible for more than 75% of the global delay. The two cases where the backbone latency is less significant in percentage is for servers located in the UK and Germany, because for these cases its value is not so much higher than the remaining delay contributions, as opposed to the situations of Australia and Japan server locations where the percentage of end-to-end RTT due to backbone latency reaches almost 90%.

Table 4.20. Backbone Delay Percentage of the End-to-end-Delay.

Server Location		Backbone Delay Percentage of the End-to-end Delay [%]
Africa	Tunisia	57.23
	South Africa	81.47
Asia	China	87.31
	India	80.79
	Japan	89.15
Europe	Germany	36.44
	United Kingdom (UK)	48.87
Middle East	United Arab Emirates	78.41
North America	USA (East Coast)	76.47
	USA (West Coast)	85.56
Oceania	Australia	88.68
South America	Brazil	86.66

The second important insight from Figure 4.29 that must be highlighted is the low impact that NSC has on the global delay, compared to the sum of the remaining delay components. Table 4.21 further details this result.

Table 4.21. NSC Percentage of the End-to-end-Delay.

Server Location		NSC Percentage of the End-to-end Delay [%]
Africa	Tunisia	5.03
	South Africa	2.17
Asia	China	1.49
	India	2.25
	Japan	1.27
Europe	Germany	7.47
	United Kingdom (UK)	6.01
Middle East	United Arab Emirates	2.53
North America	USA (East Coast)	2.76
	USA (West Coast)	1.69
Oceania	Australia	1.33
South America	Brazil	1.56

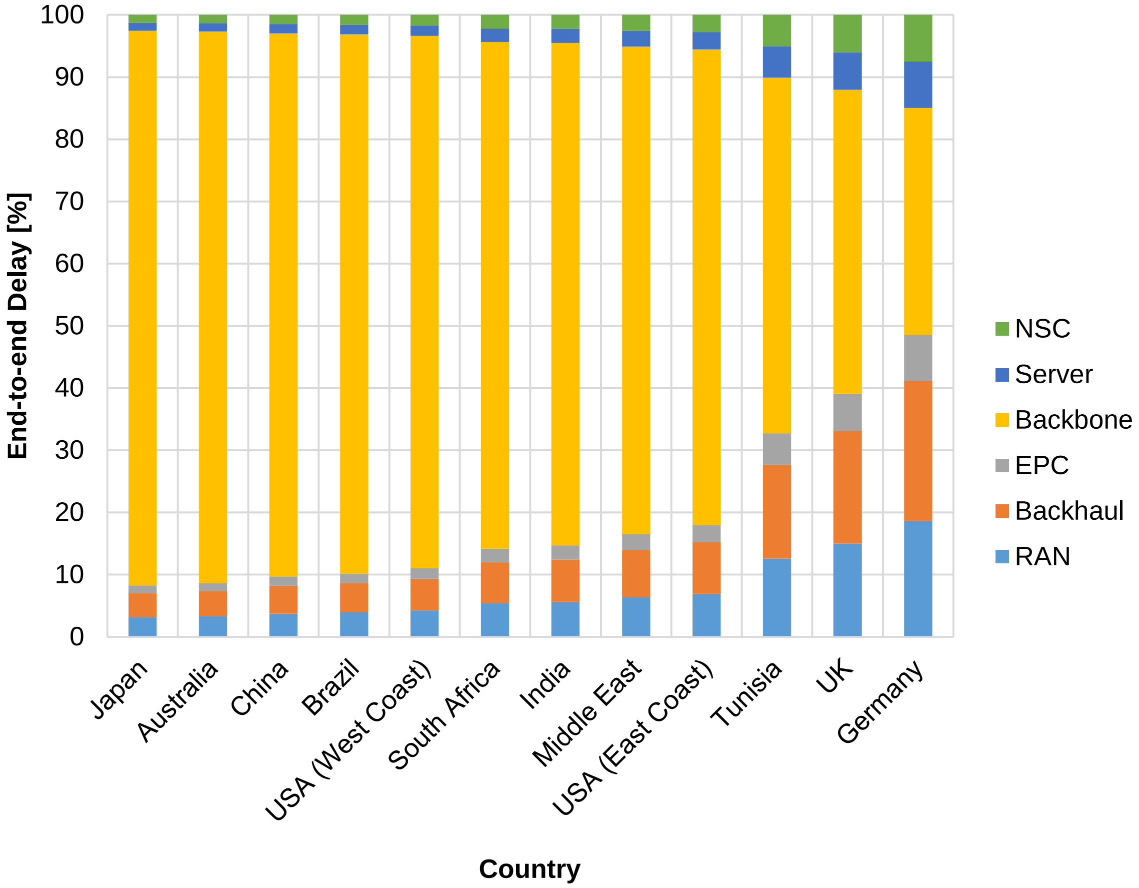

Figure 4.29. End-to-end RTT from Portugal to the World – percentage of each component.

Thus, for the majority of the server locations (9 out of 12), NSC accounts for less than 3% of the total end-to-end delay, it seems reasonable to conclude that in most situations one can leverage the QoE enhancements brought by NSC without compromising latency constraints. Table 4.22 details the percentages of end-to-end delay associated with the remaining components: RAN, Backhaul, EPC and Server, among which backhaul latency has the most important quote of total end-to-end delay.

Table 4.22. RAN, Backhaul, EPC and Server Percentages of End-to-end Delay.

Server Location		**End-to-end Delay Quote [%]**			
		RAN	**Backhaul**	**EPC**	**Server**
Africa	Tunisia	12.57	15.09	5.03	5.03
	South Africa	5.44	6.53	2.17	2.17
Asia	China	3.73	4.47	1.49	1.49
	India	5.64	6.77	2.25	2.25
	Japan	3.18	3.82	1.27	1.27
Europe	Germany	18.69	22.42	7.47	7.47
	United Kingdom (UK)	15.03	18.04	6.01	6.01
Middle East	United Arab Emirates	6.34	7.61	2.53	2.53
North America	USA (East Coast)	6.92	8.30	2.76	2.76
	USA (West Coast)	4.24	5.09	1.69	1.69
Oceania	Australia	3.32	3.99	1.33	1.33
South America	Brazil	3.92	4.70	1.56	1.56

Chapter 5

Conclusions

This chapter finalises this work by summarising the main conclusions obtained and pointing out aspects to be developed in future work.

The main goal of this thesis was to analyse how Software Defined Networking technology can improve network performance in LTE-A, by taking advantage of the separation between the control and data planes. Under this scope, a study of the impact of C-RAN and virtualisation techniques in an operator's network was made, namely in terms of the necessary number of storage and processing nodes and the links in between, taking into account latency and capacity constraints. The lack of information on how SDN technology performs out of laboratory environments did not enable an in depth quantification of its enhancements, in comparison with currently deployed solutions.

In the first chapter, a global view of the mobile communication systems evolution is presented along with a forecast for the ever increasing traffic demand, followed by the motivation for the present work and a short description of the contents in each chapter.

Chapter 2 provides a theoretical background on LTE's network architecture and radio interface, as well as on SDN fundamentals. A broad overview of the SDN paradigm is given, including its founding principles, generic architecture and a presentation of the benefits and drawbacks of its applicability to mobile communications. A high level description of the most prominent SDN protocol, the OpenFlow standard, is also given in this chapter. A background on Virtualisation and Cloud is provided, including the main benefits and characteristics of NFV and C-RAN, and how these technologies relate to SDN. In the services and applications part of this chapter, a set of the most used services is studied in terms of their QoS singularities, namely in what concerns bit rate, delay and packet loss. The chapter ends with a roundup of the state of the art, which contemplates the most relevant works developed in the SDN and mobile network virtualisation domains.

In Chapter 3, the model is described, starting with its delay and capacity parameters, followed by an overview and implementation details, and ending with a model assessment. So, the first part of this chapter consists in the mathematical foundations regarded while building the model, starting with delay, which is the constraint that conditions the positioning of BBU Pools and Core Nodes, followed by a description of how capacity aspects condition the dimensioning of processing nodes, in terms of the number of required blade servers. The model developed during this work was entirely developed from scratch, using the base station positioning of a mobile operator working in Portugal as test case and typical values for delay and capacity parameters seen in previous research, especially in what concerns the fronthaul and backhaul parts of the network.

In the model overview section, a systemic view of the model developed for the purpose of this thesis is depicted, in a way that one can easily identify which are the inputs and outputs. Moreover, the various modules that make up the model are presented: RAN Dimensioning, Core Dimensioning, Data Processing and Data Fitting. The concretisation of these modules is detailed in the model implementation section. RAN Dimensioning and Core Dimensioning are the first two implementation stages, consisting respectively of the positioning of BBU Pools and Core Nodes according to delay constraints, along with accounting for the capacity demand that must be satisfied in every node. These modules take into consideration the geographical limits of Portugal and implement a positioning strategy that uses a set of square form coverage areas representing an initial representation of data centres coverage areas, with dimensions dictated by the maximum front- and backhaul delays. The first step of

RAN Dimensioning positioned BBU Pools inside each of the referred areas by computing their coordinates as an average of the cell sites coordinates within each square area. Since this first step leads to disconnected cell sites within each area, the second step of this module has the goal of finding disconnected cell sites that can be connected to a BBU Pool present in one of the neighbouring square areas. The third step of RAN Dimensioning creates and positions extra BBU Pools for sites or groups of sites that ended up disconnected even after the first two steps. Finally, the last step of RAN Dimensioning accounts for the total capacity that needs to be satisfied in every BBU Pool, accounting for the number and class of supported cell sites. Core Dimensioning follows an identical algorithm to the one used in RAN Dimensioning, differing fundamentally in the delay constraint, which in this case is maximum backhaul delay instead of maximum fronthaul delay, and instead of having cell sites, BBU Pools and fronthaul links, one has BBU Pools, Core Nodes and backhaul links, respectively. Also, all backhaul links are assumed to work over fibre – no microwave links are regarded for this part of the network.

The Data Processing and Data Fitting blocks of the model aim at drawing meaningful insights and gathering trends for the data obtained from RAN Dimensioning and Core Dimensioning. So, Data Processing is responsible for analysing delay, distance and capacity demand tables and computing the relevant outputs – first, the ones related to maximum delay constraints in the fronthaul and backhaul, and then the ones concerning the number of required blade servers in BBU Pools and Core Nodes for different capacity values per unit. Data Fitting is responsible for fitting the data sets coming from the Data Processing stage with suitable mathematical models, in order to provide a clearer knowledge of the kind of behaviour presented by the various outputs. The final part of Chapter 3 is the model assessment, where the developed model was applied to a smaller area of the country, in order to test if it was correctly implemented. For this purpose, the model implementation was executed for the selected area and its outputs were analysed and compared with what was theoretically expected. The outputs' behaviour was aligned with expectations, hence, the implementation was validated, allowing to proceed for a countrywide analysis.

Chapter 4 starts by providing a description of the scenario used in this thesis, the continental part of Portugal, namely the geographical distribution of base stations of a mobile operator. Three classes of sites were defined, each with its own fronthaul link capacity requirement – this distinction was used in Section 4.2.2 and onward, in order to aggravate the capacity demand to be supported by the set of BBU Pools and nodes upstream. Furthermore, the range of values for maximum front- and backhaul delays used in simulation were defined, as well as the typical values for delays in the rest of the network, and the ranges for the capacity for blade servers in BBU Pools and Core Nodes.

In RAN Analysis, the results related to the RAN are presented and discussed, starting with the ones concerning the impact of the maximum fronthaul delay constraint, followed by a study of the blade server capacity impact. Then, results associated with the Core are shown and analysed, again starting with delay-related aspects and ending with blade server capacity impact. The last part of Chapter 4 concerns an end-to-end delay estimate, for a set of server locations around the globe, considering typical delay values for the radio access, backhaul, core and backbone segments of the network, as well as for server processing and NSC.

The first RAN output discussed is the number of BBU Pools required to support the set of cell sites positioned countrywide. The decrease of this output with the maximum fronthaul delay increase is confirmed, from 273 BBU Pools for 75 μs of maximum fronthaul delay to 56 for 200 μs – a rational model was chosen as the most appropriate for fitting the data and its expression was derived, resulting in a correlation of 0.995 and an RMSE of 6.574. Then, the share of single site BBU Pools was assessed and it indicates that the maximum pooling gains are achieved for 150 μs of maximum fronthaul delay. Next, the maximum, minimum and average fronthaul delay and fronthaul distance evolution were examined, as well as their standard deviation. For both quantities, a linear increase in their average and standard deviation with the maximum fronthaul delay is observed – correlation of 0.973, 0.996 and 0.998 for the three delay-related curves, as well as for the corresponding distance-related curves. These curves show that the average values are significantly below the imposed maximum constraint. Also, it must be noted that the minimum fronthaul delay is zero in every case, due to the ever positive share of single site BBU Pools, and consequently this exact scenario translates into an always null minimum fronthaul distance. Regarding fronthaul distance, a few considerations were also drawn on what concerns the spreading of cell sites along the BBU Pools coverage areas, where in rural areas the sites are more or less equally scattered, as opposed to urban areas, where sites tend to be more concentrated near the BBU Pool – for these purpose, BBU Pools were divided in three classes (Urban, Suburban and Rural) according to the number of sites within their coverage area.

The decrease in the percentage of possible microwave links with the increase in maximum fronthaul delay is confirmed, thus, obviously following the decrease in the coverage areas dimension – a rational model was chosen as the most appropriate for fitting the data, with a correlation of 0.984. As for the evolution of the number of sites per BBU Pool and number of blade servers per BBU Pool as the maximum fronthaul delay is increased, a very similar behaviour is observed in both cases, with an increasing trend of the average value for both quantities as well as of the standard deviation. This scenario was expected, since as the maximum delay increases these nodes to provide connectivity for more cell sites, consequently requiring a higher number of blade servers. Quadratic models were used for the fitting of these data sets, with a correlation of 0.999 and 0.993 for the sites per BBU Pool curves and of 0.999 and 0.993 for the servers per BBU Pool curves. The minimum number of sites per BBU Pool verified is 1, corresponding to a minimum of 2 blade servers per BBU Pool, for a unit capacity of 10 Gbps. Lastly, in what concerns the RAN results, it is observed that for a positioning of BBU Pools corresponding to 200 μs of maximum fronthaul delay, an increase of blade server capacity from 10 Gbps to 60 Gbps causes the average number of blade servers to decrease accordingly – the two quantities are inversely proportional. Here, a rational model was chosen as the most suitable for data fitting. The minimum number of blade servers per BBU Pool is 2 for a blade server capacity of 10 Gbps and 1 for a blade server capacity in between 20 Gbps and 60 Gbps.

For the Core Analysis, the BBU Pools positioning used is the one obtained for a maximum fronthaul delay of 200 μs. The maximum backhaul delay was the first constraint examined, ranging between 2 ms and 12 ms. Here, it is verified that the total number of Core Nodes required is 2 for a maximum backhaul delay of 2 ms – for the remaining values, a single Core Node is enough to cover the entire country, from a delay restriction perspective. Anyway, for resilience purposes, extra nodes must obviously be present to provide redundancy, but this aspect was not within the scope of the present thesis. For 2 ms, a single Core Node is not enough, because the links connecting it to the three southernmost BBU Pools would not fulfil the referred backhaul delay constraint. Regarding the average backhaul delay, it increases from approximately 781 µs in a scenario with 2 Core Nodes to 845 µs in a scenario with a single Core Node. Hence, in either case the average backhaul delay remains significantly below the imposed constraint. As for the maximum and minimum backhaul delays that were verified, the minimum is 142 µs for a maximum backhaul delay of 2 ms and 165 µs for the remaining values, while the maximum was 1.41 ms for a maximum backhaul delay of 2 ms and 1.46 ms for the remaining values. In terms of distance, the average backhaul distance increases from approximately 156 km with 2 Core Nodes to 169 km with a single Core Node. In what concerns the maximum and minimum backhaul distances, the minimum is 28 km for a maximum backhaul delay of 2 ms and 33 km for the remaining values, while the maximum is 283 km for a maximum backhaul delay of 2 ms and 292 km for the remaining values.

For the situation with only one Core Node, the supported BBU Pools require it to have at least 5 812 blade servers if one considers a blade server capacity of 50 Gbps. When two Core Nodes are considered and again for a blade server capacity of 50 Gbps, since the node in Algarve is allocated particularly to the 3 southernmost BBU Pools it can have a minimum of 41 blade servers, while the other node must have at least 5 771 blade servers in order to support the remaining 53 BBU Pools. For evaluating how the number of blade servers per Core Node evolves as the blade server capacity increases, it was chosen the scenario with a single Core Node. With the increase of blade server capacity from 50 Gbps to 100 Gbps, the number of blade servers needed in the Core Node decreased accordingly (from almost 5 812 blade servers to 2 906) – a rational model was used for the fitting.

The end-to-end delay evaluation ending Chapter 4 has the goal of providing a broader view of the delay scenario. Here, the delay values regarded for the RAN, Backhaul, EPC, Server and NSC were fixed, but the backbone delay was varied according to server locations worldwide. Furthermore, benchmarking lines that allow an assessment of where various kinds of services could be hosted were presented. A drastic difference in end-to-end delay between situations where the server is in Europe and servers located in other parts of the world was noted. For the majority of server locations (9 out of 12), NSC accounted for less than 3% of the total end-to-end delay, hence, it seems reasonable to conclude that in most situations one can leverage the QoE enhancements brought by NSC without compromising latency constraints.

There are various different subjects that can be tackled in future works following this thesis, since many aspects of SDN and C-RAN remain in need of further investigation regarding the efficiency gains that these paradigms bring in comparison with currently deployed technologies. The positioning algorithm developed for the purpose of this thesis provides a possible placement of all processing nodes, but it does not guarantee that it is optimal, so it would be interesting to add optimisation techniques to the model, in order to guarantee latency minimisation. Additional positioning restrictions for the data centres could be considered in the model, for example by excluding zones where fibre resources are not available. Also, for the fronthaul, a more thorough analysis of microwave technology viability would be valuable.

Although the implemented model is a reliable tool for providing a general picture of how C-RAN could be implemented in terms of the number and positioning of processing nodes and its consequences, it lacks temporal variation features that would allow a better understanding of the dynamic nature of mobile traffic influence on network resources utilisation. For example, with a varying number of users and respective traffic demand over time, one could study the potential savings introduced by SDN techniques, which could scale up and down the number of switched on blade servers as needed, leading to energy efficiency improvements and OPEX reduction.

Hereafter, further capacity constraints can be studied as well. For example, one can impose limits in data centre square area, imposing a maximum number of blade servers per data centre, hence causing a maximum number of sites that can be handled. Mostly on urban areas, this approach would lead to a higher number of small data centres to distribute within the territory, as opposed to having a macro data centre handling all sites in the urban zone under analysis. Load balancing among BBU Pools, by leveraging SDN control, is also a relevant study subject, taking into account zones where cell sites can be served by more than one nearby BBU Pool.

Moreover, the end-to-end latency estimate presented in Chapter 4 can be further explored, for example by focusing more in server positioning within the European zone, namely taking into account the actual placement of data centres where more popular content and processing resources are hosted – e.g., video contents and search engine resources. It would also be interesting to explore how data caching in edge data centres can contribute to minimise latency, leveraging SDN and NSC capabilities.

Annex A

District Capitals’ Coordinates

In this annex, the coordinates used in the present work for the district capitals are shown.

Table A.1. shows the geographical coordinates of the Portuguese district capitals, as obtained on Google Earth.

Table A.1. Coordinates of the Portuguese district capitals.

Name	Latitude [º]	Longitude [º]
Aveiro	40.6338	-8.6517
Beja	38.0113	-7.8648
Braga	41.5405	-8.4223
Bragança	41.8021	-6.7552
Castelo Branco	39.8149	-7.4967
Coimbra	40.1993	-8.4100
Évora	38.5659	-7.9129
Faro	37.0185	-7.9286
Guarda	40.5331	-7.2674
Leiria	39.7468	-8.8055
Lisboa	38.7187	-9.1399
Portalegre	39.2813	-7.4267
Porto	41.1533	-8.6249
Santarém	39.2344	-8.6846
Setúbal	38.5272	-8.8886
Viana do Castelo	41.6976	-8.8275
Vila Real	41.3016	-7.7371
Viseu	40.6536	-7.9122

Annex B

Goodness of Fit Parameters

In this annex, the Goodness of Fit Parameters of models used in Data Fitting are presented.

The tables in this annex present the Goodness of Fit Parameters obtained for each of the mathematical models considered for Data Fitting, for the various data sets fitted in Chapter 4. The rows in bold represent the mathematical model that was chosen for the fitting, from the set of tested models. In some tables, the set of four models is not complete because in more than one case some models do not make sense from a theoretical point of view, leading to absurd Goodness of Fit parameters like a negative R^2, or because a perfect fitting exists, allowing one to neglect the other models.

Table B.1. Goodness of Fit for the Number of BBU Pools vs. Maximum Fronthaul Delay.

Model		R^2	**RMSE**
Rational		**0.9948**	**6.574**
Polynomial	Degree 1	0.8696	33.07
	Degree 2	0.9906	10.24
Exponential		0.9846	11.36

Table B.2. Goodness of Fit for the Fronthaul Delay (μ) vs. Maximum Fronthaul Delay.

Model		R^2	**RMSE**
Rational		-	-
Polynomial	**Degree 1**	**0.9960**	**1.278**
	Degree 2	0.9968	1.315
Exponential		0.9743	3.226

Table B.3. Goodness of Fit for the Fronthaul Delay (μ+σ) vs. Maximum Fronthaul Delay.

Model		R^2	**RMSE**
Rational		-	-
Polynomial	**Degree 1**	**0.9982**	**1.3390**
	Degree 2	0.9985	1.4130
Exponential		0.9785	4.6670

Table B.4. Goodness of Fit for the Fronthaul Delay (μ-σ) vs. Maximum Fronthaul Delay.

Model		R^2	**RMSE**
Rational		-	-
Polynomial	**Degree 1**	**0.9730**	**1.3930**
	Degree 2	0.9783	1.4410
Exponential		0.9451	1.9870

Table B.5. Goodness of Fit for the Fronthaul Distance (μ) vs. Maximum Fronthaul Distance.

Model		R^2	**RMSE**
Rational		-	-
Polynomial	**Degree 1**	**0.9960**	**0.2556**
	Degree 2	0.9968	0.2629
Exponential		0.9743	0.6452

Table B.6. Goodness of Fit for the Fronthaul Distance (μ+σ) vs. Maximum Fronthaul Distance.

Model		R^2	**RMSE**
Rational		-	-
Polynomial	**Degree 1**	**0.9982**	**0.2678**
	Degree 2	0.9985	0.2826
Exponential		0.9785	0.9334

Table B.7. Goodness of Fit for the Fronthaul Distance (μ-σ) vs. Maximum Fronthaul Distance.

Model		**R^2**	**RMSE**
Rational		-	-
Polynomial	**Degree 1**	**0.9730**	**0.2786**
	Degree 2	0.9783	0.2883
Exponential		0.9451	0.3974

Table B.8. Goodness of Fit for the Share of Possible Microwave Links vs. Maximum Fronthaul Delay.

Model		**R^2**	**RMSE**
Rational		**0.9835**	**0.4262**
Polynomial	Degree 1	0.8502	1.2830
	Degree 2	0.9660	0.7059
Exponential		0.9671	0.6069

Table B.9. Goodness of Fit for the Number of Sites per BBU Pool (μ) vs. Maximum Fronthaul Delay.

Model		**R^2**	**RMSE**
Rational		-	-
Polynomial	Degree 1	0.9950	1.7870
	Degree 2	**0.9996**	**0.6100**
Exponential		0.9815	3.4280

Table B.10. Goodness of Fit for the Number of Sites per BBU Pool (μ+σ) vs. Max. Fronthaul Delay.

Model		**R^2**	**RMSE**
Rational		-	-
Polynomial	Degree 1	0.9829	9.9900
	Degree 2	**0.9934**	**7.1860**
Exponential		0.9862	8.9680

Table B.11. Goodness of Fit for the Number of Servers per BBU Pool (μ) vs. Max. Fronthaul Delay.

Model		**R^2**	**RMSE**
Rational		-	-
Polynomial	Degree 1	0.9937	3.3070
	Degree 2	**0.9985**	**1.1290**
Exponential		0.9815	6.3440

Table B.12. Goodness of Fit for the Number of Servers per BBU Pool (μ+σ) vs. Max. Fronthaul Delay.

Model		**R^2**	**RMSE**
Rational		-	-
Polynomial	Degree 1	0.9829	18.490
	Degree 2	**0.9934**	**13.300**
Exponential		0.9862	16.600

Table B.13. Goodness of Fit for the Number of Servers per BBU Pool (μ) vs. Blade Server Capacity.

Model		**R^2**	**RMSE**
Rational		**1**	**0**
Polynomial	Degree 1	-	-
	Degree 2	-	-
Exponential		-	-

Table B.14. Goodness of Fit for the Number of Servers per BBU Pool (μ+σ) vs. Blade Server Capacity.

Model		***R*²**	**RMSE**
Rational		**1**	**0**
Polynomial	Degree 1	-	-
	Degree 2	-	-
Exponential		-	-

Table B.15. Goodness of Fit for the Number of Servers in Core Node vs. Blade Server Capacity.

Model		***R*²**	**RMSE**
Rational		**1**	**0**
Polynomial	Degree 1	-	-
	Degree 2	-	-
Exponential		-	-

References

[3GPP14a] 3GPP, Technical Specification Group Services and System Aspects, *Policy and charging control architecture (Release 13)*, Report TS 23.203, V13.0.0, June 2014 (http://www.3gpp.org/ftp/Specs/html-info/23203.htm).

[3GPP14b] http://www.3gpp.org/About-3GPP, October 2014.

[3GPP14c] 3GPP, Technical Specification Group Services and System Aspects, *Quality of Service (QoS) concept and architecture (Release 12)*, Report TS 23.107, V12.0.0, September 2014 (http://www.3gpp.org/ftp/Specs/html-info/23107.htm).

[3GPP14d] 3GPP, Technical Specification Group Radio Access Network, *Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception (Release 12)*, Report TS 36.101, V12.5.0, October 2014 (http://www.3gpp.org/ftp/Specs/html-info/36101.htm).

[3GPP14e] 3GPP, Technical Specification Group Radio Access Network, *Feasibility study for evolved Universal Terrestrial Radio Access (UTRA) and Universal Terrestrial Radio Access Network (UTRAN) (Release 12)*, Report TR 25.912, V12.0.0, September 2014 (http://www.3gpp.org/ftp/Specs/html-info/25912.htm).

[Alme13] Almeida, D., *Inter-Cell Interference Impact on LTE Performance in Urban Scenarios*, M.Sc. Thesis, Instituto Superior Técnico, Lisbon, Portugal, 2013.

[ANAC12] ANACOM, *Information on multi-band spectrum auction (3)*, Public Consultation, Lisbon, Portugal, December 2012 (http://www.anacom.pt/render.jsp?contentId=1106646&languageId=1).

[Bast13] Basta, A., Kellerer, W., Hoffmann, M., Hoffmann, K., and Schmidt, E. D., “A Virtual SDN-enabled LTE EPC Architecture: a case study for S-/P-Gateways functions”, in *Proc. of SDN4FNS – IEEE Software Defined Networks for Future Networks and Services Conference*, Trento, Italy, November 2013.

[Bast14a] Basta, A., Kellerer, W., Hoffmann, M., Morper, H. J., and Hoffmann, K., “Applying NFV and SDN to LTE Mobile Core Gateways; The Functions Placement Problem”, in *Proc. of ACM 4th Workshop on All Things Cellular: operations, applications, & challenges*, New York, NY, USA, August 2014.

[Bast14b] Basta, A., Blenk, A., Hoffmann, M., Morper, H. J., Hoffmann, K., and Kellerer, W., “SDN and NFV Dynamic Operation of LTE EPC Gateways for Time-varying Traffic Patterns”, in *Proc. of EAI 6th International Conference on Mobile Networks and Management*, Würzburg, Germany, September 2014.

[ClPi15] http://www.cloudping.info/ (Latency estimation tool for Amazon Web Services), April, 2015.

[CMRI10] Chen, K., Duan, R., *C-RAN: The Road towards Green Radio RAN*, White Paper, China Mobile Research Institute, China, 2010.

[CoCr11] Copeland, R., and Crespi, N., “Modelling multi-MNO business for MVNOs in their evolution to LTE, VoLTE & advanced policy”, in *Proc. of 15th IEEE International Conference on Intelligence in Next Generation Networks (ICIN),* Berlin, Germany, October 2011.

[Corr14] Correia, L.M., *Mobile Communication Systems – Lecture Notes*, Instituto Superior Técnico, Lisbon, Portugal, 2014.

[Cost14] Costanzo, S., Xenakis, D., Passas, N., and Merakos, L., “OpeNB: A framework for virtualizing base stations in LTE networks”, in *Proc. of IEEE International Conference on Communications (ICC)*, Sydney, Australia, June 2014.

[Csas13] Császár, A., John, W., Kind, M., Meirosu, C., Pongrácz, G., Staessens, D., and Westphal, J., "Unifying Cloud and Carrier Network", in *Proc. of Distributed Cloud Computing Workshop,* Dresden, Germany, December 2013.

[Damo13] Damouny, N., *Inside NFV, SDN & the Emerging Network*, EE Times, June 2014 (http://www.eetimes.com/author.asp?section_id=36&doc_id=1320009).

[DDGF12] Desset, C., Debaillie, B., Giannini, V. and Fehske, A., "Flexible power modeling of LTE base stations", in *Proc. of IEEE Wireless Communications and Networking Conference*, Shanghai, China, April 2012.

[Eric14] Ericsson, *Ericsson Mobility Report*, Public Consultation, Stockholm, Sweden, June 2014 (http://www.ericsson.com/mobility-report).

[Eric15] Ericsson, *Characteristics Requirements for LTE Backhaul*, Stockholm, Sweden, 2015.

[ETSI12] Chiosi, M., *Network Functions Virtualisation: An Introduction, Benefits, Enablers, Challenges & Call for Action*, White Paper, ETSI, Germany, October 2012.

[Gude08] Gude, N., Koponen, T., Pettit, J., Pfaff, B., Casado, M., McKeown, N., and Shenker, S., “NOX: towards an operating system for networks”, in *Proc. of ACM SIGCOMM Computer Communication Conference*, New York, NY, USA, July 2008.

[Gudi13] Gudipati, A., Perry, D., Li, L. E., and Katti, S., “SoftRAN: Software defined radio access network”, in *Proc. of the 2nd ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking*, New York, NY, USA, 2013.

[Habe12] Haberland, B., and Rehm, W., “Concept for load balancing in a radio access network”, *United States Patent Application 14/111,386*, USA, 2012.

[HoTo11] Holma,H. and Toskala,A., *LTE for UMTS: Evolution to LTE Advanced (2nd Edition)*, John Wiley & Sons, Chichester, UK, March 2011.

[HSMA14] Hawilo, H., Shami, A., Mirahmadi, M., and Asal, R., “NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC)”, *IEEE Network*, Vol. 28, No. 6, November 2014, pp. 18-26.

[InRe14] Webb, R., *Macrocell Backhaul Strategies: Global Service Provider Survey*, Infonetics Research, London, United Kingdom, October 2014.

[Jamm14] Jammal, M., Singh, T., Shami, A., Asal, R., and Li, Y., “Software Defined Networking: State of the Art and Research Challenges”, Survey Paper, Elsevier, Canada, June 2014.

[Jarr14] Jarraya, Y., Madi, T., and Debbabi, M., “A survey and a layered taxonomy of software-defined networking”, *IEEE Communications Surveys & Tutorials*, Vol. 16, No. 4, April 2014, pp. 1955-1980.

[Jars14] Jarschel, M., Zinner, T., Hoßfeld, T., Tran-Gia, P., and Kellerer, W., "Interfaces, attributes, and use cases: A compass for SDN", *IEEE Communications Magazine,* Vol.52, No. 6, June 2014, pp. 210-217.

[JLVR13] Jin, X., Li, L. E., Vanbever, L., and Rexford, J., "Softcell: scalable and flexible cellular core network architecture", in *Proc. of the 9th ACM Conference on Emerging networking experiments and technologies*, New York, NY, USA, December 2013.

[John13] John, W., Pentikousis, K., Agapiou, G., Jacob, E., Kind, M., Manzalini, A., and Meirosu, C., "Research directions in network service chaining", in *Proc. of SDN4FNS – IEEE Software Defined Networks for Future Networks and Services Conference*, Trento, Italy, November 2013.

[Kara14] Karagiannis, G., Jamakovic, A., Edmonds, A., Parada, C., Metsch, T., Pichon, D., and Bohnert, T. M., "Mobile Cloud Networking: Virtualisation of Cellular Networks", in *Proc. of the 21st International Conference on Telecommunications*, Lisbon, Portugal, May 2014.

[Kemp12] Kempf, J., Johansson, B., Pettersson, S., Luning, H., and Nilsson, T., "Moving the mobile Evolved Packet Core to the cloud", in *Proc. of IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Barcelona, Spain, October 2012.

[Kemp14] Kempf, J., Korling, M., Baucke, S., Touati, S., McClelland, V., Mas, I., and Backman, O., "Fostering rapid, cross-domain service innovation in operator networks through Service Provider SDN", in *Proc. of IEEE International Conference on Communications (ICC),* Sydney, Australia, June 2014.

[Khan14] Khan, F. A., "Virtualised EPC: Unleashing the potential of NFV and SDN", in *Proc. of the 25th European Regional Conference of the International Telecommunications Society (ITS)*, Brussels, Belgium, June 2014.

[LiMa12] Li, L. E., Mao, Z. M., and Rexford, J., "Toward software-defined cellular networks", in *Proc. of European Workshop on Software Defined Networking (EWSDN)*, Darmstadt, Germany, October 2012.

[LiMR12] Li, L. E., Mao, Z. M., and Rexford, J., *CellSDN: Software-defined cellular networks*, Technical Report, Princeton University, 2012.

[LiYu14] Liang, C., and Yu, F. R., "Wireless Network Virtualisation: A Survey, Some Research Issues and Challenges", *IEEE Communications Surveys & Tutorials*, Vol. 17, No. 1, August 2014, pp. 358-380.

[Mart13] Martins, P., *Analysis of Wireless Cloud Implementation in LTE-Advanced*, M.Sc. Thesis, Instituto Superior Técnico, Lisbon, Portugal, November 2013.

[MaSe14] Toktam M., and Seetharaman S., *On Using a SDN-based Control Plane in 5G Mobile Networks*, King's College, London, UK, 2014.

[Matl15] http://www.mathworks.com/help/curvefit/evaluating-goodness-of-fit.html (Matlab's Support Documentation), May, 2015.

[Nika11] Nikaein, N., and Krea, S., "Latency for real-time machine-to-machine communication in LTE-based system architecture", in *Proc. of IEEE 11th European Wireless Conference,* Vienna, Austria, April 2011.

[ODLP14] http://www.opendaylight.org/project/technical-overview (OpenDaylight Project), December 2014.

[ONFo12] Open Networking Foundation, *Software-Defined Networking: The New Norm for Networks*, White Paper, April 2012.

[ONFo13] Open Networking Foundation, *OpenFlow Switch Specification: Version 1.4 (Wire Protocol 0x05)*, Technical Report, October 2013.

[Pent13] Pentikousis, K., Wang, Y., and Hu, W., "Mobileflow: Toward software-defined mobile networks", *IEEE Communications Magazine,* Vol. 51, No. 7, July 2013, pp. 44-53.

[Phil12] Philip, V. D., Gourhant, Y., and Zeghlache, D., "OpenFlow as an Architecture for e-Node B Virtualisation", in *e-Infrastructure and e-Services for Developing Countries*, Springer, Berlin, Germany, 2012.

[Pizz13] Pizzinat, A., Chanclou, P., Le Clech, F., Reedeker, T. L., Lagadec, Y., Saliou, F., and Galli, P., "Optical fibre solution for mobile fronthaul to achieve Cloud Radio Access Network", in *Proc. of IEEE Future Network and Mobile Summit*, Lisbon, Portugal, July 2013.

[Rigg14] Riggio, R., Gomez, K., Goratti, L., Fedrizzi, R., and Rasheed, T., "V-Cell: Going beyond the cell abstraction in 5G mobile networks", in *Proc. of IEEE Network Operations and Management Symposium (NOMS),* Krakow, Poland, May 2014.

[Said13] Said, S. B. H., Sama, M. R., Guillouard, K., Suciu, L., Simon, G., Lagrange, X., and Bonnin, J. M., "New control plane in 3GPP LTE/EPC architecture for on-demand connectivity service", in *Proc. of IEEE 2nd International Conference on Cloud Networking (CloudNet)*, San Francisco, CA, USA, November 2013.

[Sama14] Sama, M. R., Ben Hadj Said, S., Guillouard, K., and Suciu, L., "Enabling network programmability in LTE/EPC architecture using OpenFlow", in *Proc. of IEEE 12th International Symposium on Modeling and Optimisation in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, Hammamet, Tunisia, May 2014.

[SaSt14] Savu, D., Stancu, S., “Software Defined Networking – technology details and Openlab research overview”, IT Technical Forum, CERN, Switzerland, February 2014.

[SeTB11] Sesia,S., Toufik,I. and Baker,I., *LTE - The UMTS Long Term Evolution: From Theory to Practice (2nd Edition)*, John Wiley & Sons, Chichester, UK, August 2011.

[Seze13] Sezer, S., Scott-Hayward, S., Chouhan, P. K., Fraser, B., Lake, D., Finnegan, J., and Rao, N., "Are we ready for SDN? Implementation challenges for software-defined networks”, *IEEE Communications Magazine,* Vol. 51, No. 7, July 2013, pp. 36-43.

[Sher09] Sherwood, R., Gibb, G., Yap, K. K., Appenzeller, G., Casado, M., McKeown, N., and Parulkar, G., *Flowvisor: A network virtualisation layer*, Technical Report, OpenFlow Switch Consortium, USA, 2009.

[Sund13] Sundaresan, K., Arslan, M. Y., Singh, S., Rangarajan, S., and Krishnamurthy, S. V., "FluidNet: a flexible cloud-based radio access network for small cells", in *Proc. of ACM 19th international conference on Mobile computing & networking*, New York, NY, USA, September 2013.

[TeGe15] http://submarine-cable-map-2015.telegeography.com/, April 2015.

[Yang13] Yang, M., Li, Y., Jin, D., Su, L., Ma, S., and Zeng, L., "OpenRAN: a software-defined ran architecture via virtualisation", in *Proc. of ACM SIGCOMM Computer Communication Conference*, New York, NY, USA, August 2013.

[YapK10] Yap, K. K., Kobayashi, M., Sherwood, R., Huang, T. Y., Chan, M., Handigol, N., and McKeown, N., "OpenRoads: Empowering research in mobile networks", in *Proc. of ACM SIGCOMM Computer Communication Conference*, New York, NY, USA, January 2010.

[Zhou14] Zhou, X., Zhao, Z., Li, R., Zhou, Y., Chen, T., Niu, Z., and Zhang, H., "Towards 5G: When Explosive Bursts Meet Soft Cloud", *IEEE Network Magazine*, Vol. 28, No. 6, November 2014, pp. 12-17.