University of Brasília

Institute of Exact Sciences
Department of Computer Science

Characterizing and Improving Decision-making in Fog Radio Access Networks

Jonathan Mendes de Almeida

Thesis presented in partial fulfillment of the requirements for the degree of Master of Science in Informatics

Advisor
Prof. Dr. Célia Ghedini Ralha

Co-advisor
Prof. Dr. Marcelo Antonio Marotta

Brazil
2021

University of Brasília

Institute of Exact Sciences
Department of Computer Science

Characterizing and Improving Decision-making in Fog Radio Access Networks

Jonathan Mendes de Almeida

Thesis presented in partial fulfillment of the requirements for the degree of Master of Science in Informatics

Prof. Dr. Célia Ghedini Ralha (Advisor)
University of Brasília, Brazil

Prof. Dr. Luiz A. DaSilva
Virginia Tech, United States of America

Prof. Dr. Cristiano Bonato Both
Unisinos University, Brazil

Prof. Dr. Genaína Nunes Rodrigues
Coordinator of the Graduate Program in Informatics

Brazil, Brasília, February 26, 2021

Dedication

This dissertation is dedicated to my wife Juliana Mayumi Hosoume and professors whose support and encouragement have enriched my researcher spirit and inspired me to pursue and complete this work.

Acknowledgements

First of all, I thank God for helping me even when I had no faith in You. I am thankful for blessing me with a great family, professors, and friends. I have been privileged enough to work with people I like and respect so much.

I am grateful to my wife and best friend, Juliana Mayumi Hosoume, for her unconditional love and support throughout the past eight years. Thank you for always helping me to overcome my struggles and for encouraging me to face challenging situations. You make me capable of achieving more than I believe. Thank you for always pushing me to give my best in everything I do. You are my inspiration, my strength, and my soulmate.

I thank my family for everything they have done for me and for always being present in all stages of my life. In particular, I thank my parents for always being willing to support my studies and career choices. I also thank my father-in-law and mother-in-law for being like parents to me. It is a privilege to be part of the Hosoume family.

I would like to express my most profound appreciation to all professors and friends who inspired me and contributed to my personal, intellectual, and professional growth. I particularly thank Prof. Célia G. Ralha for her kind supervision since my undergraduate studies and for always being open to investigating different research topics. I also thank Prof. Marcelo A. Marotta for his friendship and guidance to become a better researcher. I learned a lot with him since last year and I would not have been able to complete this work without his guidance. He is more than an advisor. He is a loyal friend. Thanks for perfectly preparing me to be a great researcher in the future. Besides, I would like to give a special thanks to Prof. Luiz A. DaSilva, for believing in my potential and for accepting me as his prospective Ph.D. student. I thank him for his patience, dedication, and valuable comments/suggestions during the process of writing our articles. He has been an inspiration to me since our first collaboration. In addition, I thank Prof. Cristiano B. Both for always forcing me to improve every single part of our articles (including figures, tables, and references) and for his commitment towards all our collaborations. Last but not least, I thank all professors who supervised me in my undergraduate research experiences. I immensely thank Prof. Rosana Tidon for offering me the first opportunity to do research during my undergraduate studies. I also thank Prof. Cedric Chauve, who kindly received me during my internship in Canada and pushed me to learn new things.

Finally, I thank the Brazilian National Council for Scientific and Technological Development (CNPq) for the master's research grant and the Brazilian Higher Education Improvement Coordination (CAPES) for providing access to research articles.

Abstract

Fog Radio Access Networks (F-RANs) are the result of the application of fog paradigm to cloud radio access networks, inheriting components and aspects from both. Artificial Intelligence (AI) techniques can be applied to F-RANs to achieve enhanced energy efficiency, increased throughput, and/or decreased processing power consumption. Nonetheless, to select an appropriated AI technique to apply, it is necessary to take into account the different time granularities at which decision-making occurs in F-RANs. In the first part of this work, the benefits and challenges of implementing an AI-driven F-RAN considering three time granularities (hours, minutes/seconds, and milliseconds) are discussed. For each granularity, the key enabling AI techniques are highlighted, such as deep neural networks, reinforcement learning based algorithms, adaptive online learning, and classifiers. To enable integration between AI solutions from the same time granularity and from different granularities, a multi-agent architecture for F-RANs is proposed. Furthermore, a particular problem from the time granularity hours is explored. In this sense, opportunities for network operators to reduce their expenditures through optimal allocation of virtual Base Band Units (vBBUs) in F-RANs are investigated. The optimal allocation can generate additional revenue opportunities by leasing idle processing resources to Application Service Providers. In particular, the challenge of improving vBBUs allocation in terms of optimal assignment of the workloads of Remote Radio Heads (RRHs) to Micro Data Centers (MDCs) for cost minimisation is addressed, considering the trade-off between MDC and RRH distance and processing power consumption. Thus, an optimisation model to decide the assignments between MDCs to RRHs is proposed. The optimal solution is obtained through Binary Integer Linear Programming. The solution is evaluated by applying a real Call Detail Record data set, assessing different regions from Milan. K-means clustering was used to identify the Internet traffic behaviour of different regions in Milan. The results highlight opportunities for network operators to exploit their infrastructure usage and increase their gains.

Keywords: F-RAN, multiagent systems, time granularity, edge processing, vBBU allocation, processing power allocation, Milano grid dataset.

Resumo

Fog Radio Access Networks (F-RANs) são o resultado da aplicação de paradigmas de fog/edge computing em cloud radio access networks, herdando componentes e aspectos de ambos. As técnicas de Inteligência Artificial (IA) podem ser aplicadas às F-RANs para obter maior eficiência energética, maior rendimento e/ou menor consumo de energia de processamento, e melhor tomada de decisão em diferentes situações. No entanto, para selecionar uma técnica apropriada de IA a ser aplicada, é necessário levar em consideração as diferentes granularidades de tempo nas quais a tomada de decisão ocorre em F-RANs. Na primeira parte deste trabalho são discutidos os benefícios e desafios da implementação de uma F-RAN orientado a IA, considerando três granularidades de tempo. Para cada granularidade, são destacadas as principais técnicas de aprendizado, como redes neurais profundas, aprendizado de reforço, aprendizado on-line e classificadores. Para permitir a integração entre soluções de IA, é proposta uma arquitetura multiagente para F-RANs. Além disso, é explorado um problema específico a partir da granularidade de horas. Nesse sentido, são investigadas oportunidades para as operadoras reduzirem seus gastos através da alocação ideal de virtual Base Band Units (vBBUs). A alocação ideal pode gerar oportunidades de receita adicionais, alugando recursos de processamento ocioso para Application Service Providers (ASPs). Em particular, o desafio de melhorar a alocação de vBBU em termos da atribuição ideal das cargas de trabalho entre Remote Radio Heads (RRHs) e Micro Data Centers (MDCs), considerando o trade-off entre a distância entre MDC e RRH e o consumo de poder de processamento. Assim, é proposto um modelo de otimização para decidir as atribuições entre MDCs e RRHs. A solução ideal é obtida por meio de Binary Integer Linear Programming. A solução é avaliada aplicando um conjunto de dados Call Detail Records reais, simulando diferentes regiões de Milão. A técnica de agrupamento k-means foi utilizada para identificar o comportamento do tráfego de Internet em diferentes regiões de Milão. Os resultados destacam oportunidades para as operadoras explorarem sua infraestrutura e aumentarem seus ganhos.

Palavras-chave: F-RAN, sistemas multiagentes, granularidade de tempo, alocação de vBBUs, alocação de poder de processamento.

Contents

1 Introduction **1**

1.1 Research Study Design 5

1.2 Main Contributions 8

1.3 Document Outline 8

2 AI-driven F-RANs Overview: Decision-making, Time Granularities, and ML techniques **9**

2.1 Decision-making in Hours 9

2.1.1 Optimal allocation of MDCs 10

2.1.2 Processing Power Minimization 11

2.1.3 Cost reduction of vBBU allocation 11

2.1.4 Considerations 12

2.2 Decision-making in Minutes/Seconds 13

2.2.1 Optimal service placement at the edge 14

2.2.2 Enhanced Caching Hit Rate 14

2.2.3 Optimal usage of RRHs/Fog-RRHs 15

2.2.4 Considerations 15

2.3 Decision-making in Milliseconds 16

2.3.1 Optimal spectrum resource allocation 16

2.3.2 Enhanced CPU scheduling 17

2.3.3 Optimal RRH-UE assignments 18

2.3.4 Considerations 18

2.4 Mapping Between Decision-Making and AI techniques in F-RANs 19

3 Multiagent Architecture for AI-driven F-RANs **25**

3.1 Multiagent Architecture Proposal 25

3.1.1 Top Layer: Hours Granularity 26

3.1.2 Middle Layer: Minutes/Seconds Granularity 28

3.1.3 Bottom Layer: Milliseconds Granularity 29

3.2 Communication and Interaction Protocol 30

4 Use Cases: Granularity of Hours and Decomposed Time Granularities 32

4.1 System Model 32
4.2 Problem Formulation 36
4.3 Evaluation 38
4.3.1 Data Set 38
4.3.2 Scenario 42
4.4 Results and Analysis 46
4.5 Importance and Benefits of Decomposing and Integrating Time Granularities 51

5 Concluding Remarks and Future Work 54

5.1 Conclusions 54
5.2 Future Work 56

References 58

Appendix 63

A Publications 64

List of Figures

1.1 F-RAN architecture. 3
1.2 Decision-making in F-RAN, considered at different timescales. 7

2.1 F-RAN: resources and decision-making in hours. 10
2.2 F-RAN: resources and decision-making in minutes/seconds. 13
2.3 F-RAN: resources and decision-making in milliseconds. 19

3.1 Multiagent architecture proposal for AI-driven F-RAN 26
3.2 Relation betwenn distance between RRH and MDC on the minimal number of processing cores allocated and the processing cost per hour. 27

4.1 F-RAN: system model. 33
4.2 Spatial representation of Milano Grid data set. 39
4.3 Internet traffic activity during two weeks 40
4.4 Milan city center clusters 41
4.5 Elbow test results. 42
4.6 Internet traffic activity during a day. 43
4.7 Scenario representation. 45
4.8 Milan simulation scenarios. 46
4.9 Trade-off between distance and processing power. 47
4.10 Ratio of maximum income to cost of allocation for each time of the day, on weekdays and weekends for MDCs from macrocells and smallcells. 48
4.11 Ratio of maximum income to cost of allocation for each time of the day, on weekdays and weekends for MDCs from macrocells and smallcells considering the city of Milan and the province of Trento. 50
4.12 Number of migrations per Micro Data Center (MDC) per day in Milan and Trento (Italy), considering the trade-off between processing power and distance between MDC and Remote Radio Head (RRH) evaluated in different time granularities under three time intervals: (*i*) 10 minutes; (*ii*) 30 minutes; and (*iii*) 60 minutes. 53

4.13 RMSE for each trained model. 53

List of Tables

2.1 Highlighted works for decision-making on a timescale of hours/several minutes 21
2.2 Highlighted works for decision-making on a timescale of minutes/seconds . . 22
2.3 Highlighted works for decision-making on a timescale of milliseconds 22
2.4 Complete mapping of references from 2016 to 2018 considering decision-making on a timescale of hours, seconds and milliseconds. 23
2.5 Complete mapping of references from 2019 considering decision-making on a timescale of hours, seconds and milliseconds. 24

4.1 Notation 34
4.2 MDCs' specifications 44

List of Acronyms

ACL Agent Communication Language

AI Artificial Intelligence

ASP Application Service Provider

BBU Base-Band Unit

BILP Binary Integer Linear Programming

bps Bits per second

CDR Call Detail Record

CPU Central Processing Unit

CQI Channel Quality Indicator

C-RAN Cloud Radio Access Network

DRL Deep Reinforcement Learning

FEC Forward Error Correction

FIPA Foundation for Intelligent Physical Agents

F-RAN Fog Radio Access Network

GPP General Purpose Processor

HARQ Hybrid Automatic Repeat reQuest

Hz hertz

KB Knowledge Base

km Kilometer

LTE-A Long Term Evolution Advanced

LTE Long Term Evolution

MAS Multi-agent System

mbps Megabit per second

MCS Modulation and Coding Scheme

MDC Micro Data Center

ML Machine Learning

ms millisecond

QoS Quality of Service

QoE Quality of Experience

RAN Radio Access Network

RNN Recurrent Neural Network

RMSE Root Mean Square Error

RRH Remote Radio Head

UE User Equipment

vBBU virtual Base-Band Unit

VM Virtual Machine

3GPP Third Generation Partnership Project

Chapter 1

Introduction

The open interface used in Open Radio Access Networks (O-RANs) enables the possibility of any vendor's software to work on any open radio unit, *i.e.,* O-RAN aims to define and build Radio Access Network (RAN) solutions using general purpose hardware and software, disaggregating hardware and software. In this context, mobile operators are able to virtualize and disaggregate their RAN with open interfaces, which opens more options for network operators to optimize the deployment for different requirements with a reduced associated cost. This characteristic enables operators to implement different types of RANs according to their particular interests. From a practical aspect, the O-RAN architecture is gaining momentum and can be the future of RANs, since it was designed to flexibilize the deployment of RANs in a cost effective manner. Nonetheless, all instances of O-RANs will involve some form of Cloud Radio Access Networks (C-RANs) and/or Fog Radio Access Networks (F-RANs) [Singh et al., 2020], in which it is possible to implement a C-RAN or an F-RAN in a O-RAN. In particular, the work presented in this thesis employs an F-RAN architecture.

The virtualization and disaggregation of the RAN functions started with the C-RAN. Through the Base-Band Unit (BBU) pool it was possible to centralize the workload processing. A F-RAN is the result of the application of fog paradigm to a C-RAN [Yousefpour et al., 2019]. In this sense, the F-RAN architecture can be viewed as a direct evolution from C-RANs [Peng et al., 2016], seeking to achieve a balance between centralization and distribution of computational resources. This type of network inherits components and aspects from the C-RAN's centralized architecture, such as the BBU pool that processes the workload from geographically distributed RRHs, which, in turn,

communicate with User Equipments (UEs). Following a fog paradigm, F-RAN distributes core functions, such as computation, storage, communication, and control, extending the processing to the network edge by using MDCs placed alongside the RAN, turning a typical RRH into a Fog-RRH. Figure 1.1 illustrates the architecture of an F-RAN. There is an intersection between the RRHs and Fog-RRHs from the F-RAN and the centralized/decentralized units from the O-RAN (with a higher level of disaggegation). However, different from C-RANs and F-RANs, O-RANs have open interface between MDCs and RRHs.

Regardless the type of RAN architecture in use, it is noteworthy to mention that Application Service Providers (ASPs) such as Google and Facebook are interested in provisioning services in the cloud and also at the edge of the network, aiming to achieve low latency and improve their users' Quality of Service (QoS) [Chen et al., 2018a]. In this sense, network operators are gradually migrating their infrastructure to this architecture that incorporates the fog computing paradigm as a solution to decrease expenditures [Habibi et al., 2019][Yousefpour et al., 2019]. Concomitantly, cellular networks increasingly incorporate support for Artificial Intelligence (AI)-based network management and control. It is possible, for example, to employ AI-enhanced applications to collect network operation data and perform self-healing control using the analytics function capabilities present in 5G and beyond (B5G) [3GPP, 2020].

In particular, F-RANs can optimize network performance dynamically by taking advantage of processing power near the edge when available. Dynamic decisions that require up-to-date awareness of network conditions and resource availability result in several open challenges, including decision-making regarding edge caching [Peng and Zhang, 2016], virtual Base-Band Unit (vBBU) allocation [Yu et al., 2016], and power consumption minimization through resource management [Chien et al., 2019]. Moreover, F-RANs also inherit many challenges that are present in C-RANs, including Central Processing Unit (CPU) scheduling decisions [Wang et al., 2019], resource block allocation [Alqerm and Shihada, 2018], and RRH-UE assignments [Imtiaz et al., 2018]. The F-RAN's dynamism gives rise to many of these challenges and requires adaptable and smart decisions, rather than approaches that are either hard coded in software or strictly based on utility function maximization. The use of AI shows great promise in dealing with these challenges.

There are some important works that highlight the use of Machine Learning (ML) in F-RANs. Chien, Lai, and Chao [Chien et al., 2019] discuss time constraints related to

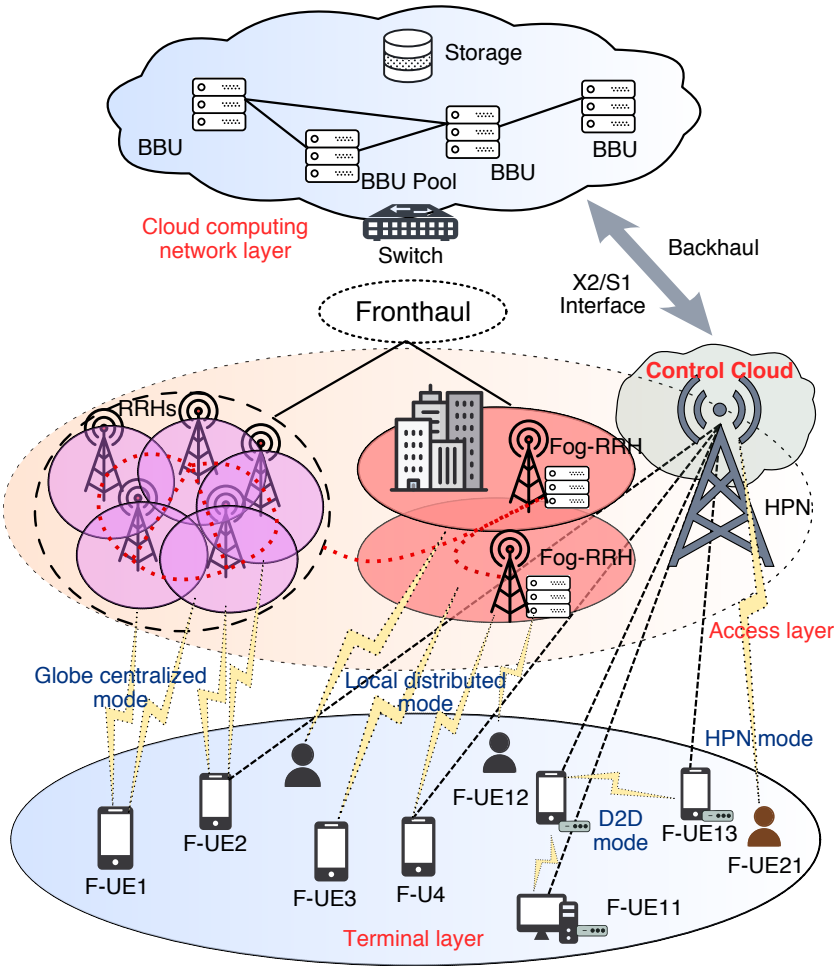

Figure 1.1: F-RAN architecture.

ML approaches, including the difficulty of applying learning techniques to environments with low latency requirements due to the complexity of the training process. Soliman and Leon-Garcia [Soliman and Leon-Garcia, 2016] discuss how constraints regarding information that all users must send to the scheduler in a time slot of one millisecond make the application of sophisticated ML algorithms unfeasible. Hence, the authors propose a simple learning approach to satisfy the time constraint. Nonetheless, the discussion regarding the time constraints for decision-making in AI-driven RANs is not well explored in the literature, being mentioned only by [Chien et al., 2019] and [Soliman and Leon-Garcia, 2016]. One of the goals of this work is to fill this gap in the literature to foster the development of novel and realistic solutions for AI-driven F-RANs.

Further, in RANs, the workload of RRHs is impacted by the density of UEs and their demand, which changes frequently, potentially leaving MDCs/processing resources unused [Tang et al., 2017]. Considering the combination of ASPs' interests and the underused resources left in F-RANs, an operator may exploit the opportunity to reduce processing power in use and generate additional revenue by leasing idle processing to ASPs, while minimising its own expenditures. To further explore this opportunity, it is necessary to improve the efficiency of processing power utilization by optimising the vBBU allocation in F-RANs. This work proposes and evaluates a mechanism to achieve this goal.

Decisions regarding the allocation of vBBUs, *i.e.,* virtual representations of BBUs considering virtual machines or containers, impact the processing resource availability in RANs [Aqeeli et al., 2018]. Note that efficient allocation of vBBUs may translate into improved resource availability in F-RANs and, consequently, enhance the opportunity for operators to generate revenue by leasing idle processing resources to ASPs. Given that the computing resources in the BBU pool are limited and the efficiency of RANs depends on the processing resources available in the BBU pool, the challenge is to find the most effective way to allocate computational resources among MDCs [Aqeeli et al., 2018]. Moreover, the minimum computational resources required to process the workload of an RRH is a function of its distance to the associated MDC, and this distance is a key factor for processing power allocation [Marotta et al., 2018]. The ultimate goal is to achieve optimality in vBBU allocation decisions, considering cost minimization and efficient usage of computational resources.

Some aspects of the allocation of vBBUs to improve RAN performance and reduce

operators' expenditures have been investigated in the literature. For instance, Aryal and Altmann [Aryal and Altmann, 2018] propose a solution that applies evolutionary computation to make decisions regarding the placement of vBBUs to process the RRHs' workload. Chien, Lai, and Chao [Chien et al., 2019] formulate the vBBU allocation as an RRH workload assignment problem and propose a solution employing a deep Recurrent Neural Network (RNN) to decide how to allocate MDCs in the RAN. Xia *et al.* [Xia et al., 2019] deal with vBBU allocation aiming to minimise the task execution and signal transmission delays by applying heuristic algorithms. Liu, Khoukhi, and Hafid [Dongqing Liu et al., 2017] propose a solution based on game theory to make decisions regarding data offloading for mobile cloud computing. Nonetheless, there is a gap in dealing with allocation decisions in terms of operational expenditures minimization by optimally decreasing the processing power in use through optimal assignment of RRHs' workload. Moreover, there is a lack of a proper model to perform allocation decisions considering the trade-off between processing power consumption and distance between MDC and RRH, which is a key factor to determine how to allocate vBBUs and how to assign RRHs' workload to MDCs in F-RANs.

1.1 Research Study Design

Fundamental Question: How to design an intelligent decision-making system able to address the challenges of F-RANs considering the distance between MDC and RRH as a key factor?

Hypothesis: AI techniques must be applied in order to improve decision-making considering the distance between MDC and RRH under different time constraints in F-RANs.

The four research questions (RQ) associated with the hypothesis are defined and presented to guide the investigations conducted in this thesis:

- RQ1 - What are the main decisions to be taken in an AI-driven F-RAN considering the different time granularities?
- RQ2 - How to integrate decision-making possibilities from different time granularities in an AI-driven F-RAN?

- RQ3 - How to formalize decision-making in F-RAN considering vBBUs allocation, the distance between MDC and RRH, and time constraints?

In this thesis, the relationship between ML approaches is discussed as well as the decisions associated with different timescales in F-RANs, as depicted in Figure 1.2. To drive the discussion, it is considered the F-RAN architecture proposed in [Peng et al., 2016] (*i.e.,* an F-RAN is composed of a BBU pool in the cloud and multiple RRHs and Fog-RRHs) under three granularities: hours, minutes/seconds, and milliseconds.

In the scope of hours, decisions regarding optimal allocation of vBBUs can be addressed with training during runtime, for instance, with online clustering and neural networks [Yu et al., 2016]. Within the range of minutes/seconds, service placement, caching, and routing decisions can be made through more sophisticated approaches, applying, for example, contextual reinforcement learning [Chen et al., 2018a], deep reinforcement learning agents [Xu et al., 2017], and adaptive online learning [Jiang et al., 2019]. Decisions that must be made in milliseconds, such as Modulation and Coding Scheme (MCS) prediction for transmission power allocation, resource block scheduling, and CPU scheduling, impose strict time constraints on the AI solutions that can be applied [Chien et al., 2019]. Moreover, this thesis examines how ML-based solutions can improve the performance of F-RANs through resource management decisions made at different time constraints. This discussion is fundamental to characterize the potential trade-offs between decision-making approaches at different time granularities, which highlights future opportunities on AI-driven F-RANs. Different from the aforementioned solutions, in which the applications is confined to a single kind of decision within one time granularity, the proposed architecture aims to provide an possible solution to implement the integration among decisions from different time granularities while characterizing trade-offs between decision-making possibilities.

Further, a case study from the time granularity of hours is presented. In particular, this work presents a solution for vBBU allocation in F-RANs through the optimal assignment of RRHs' workload for cost minimization, considering the relation between processing power consumption and distance between MDCs and RRHs. The vBBU allocation problem is formulated as an optimization problem in terms of decisions regarding assignments of RRHs to MDCs. The objective function is defined in terms of the system's cost minimization, subject to allocation and assignment constraints. The optimization problem is formulated as a Binary Integer Linear Programming (BILP), which minimises

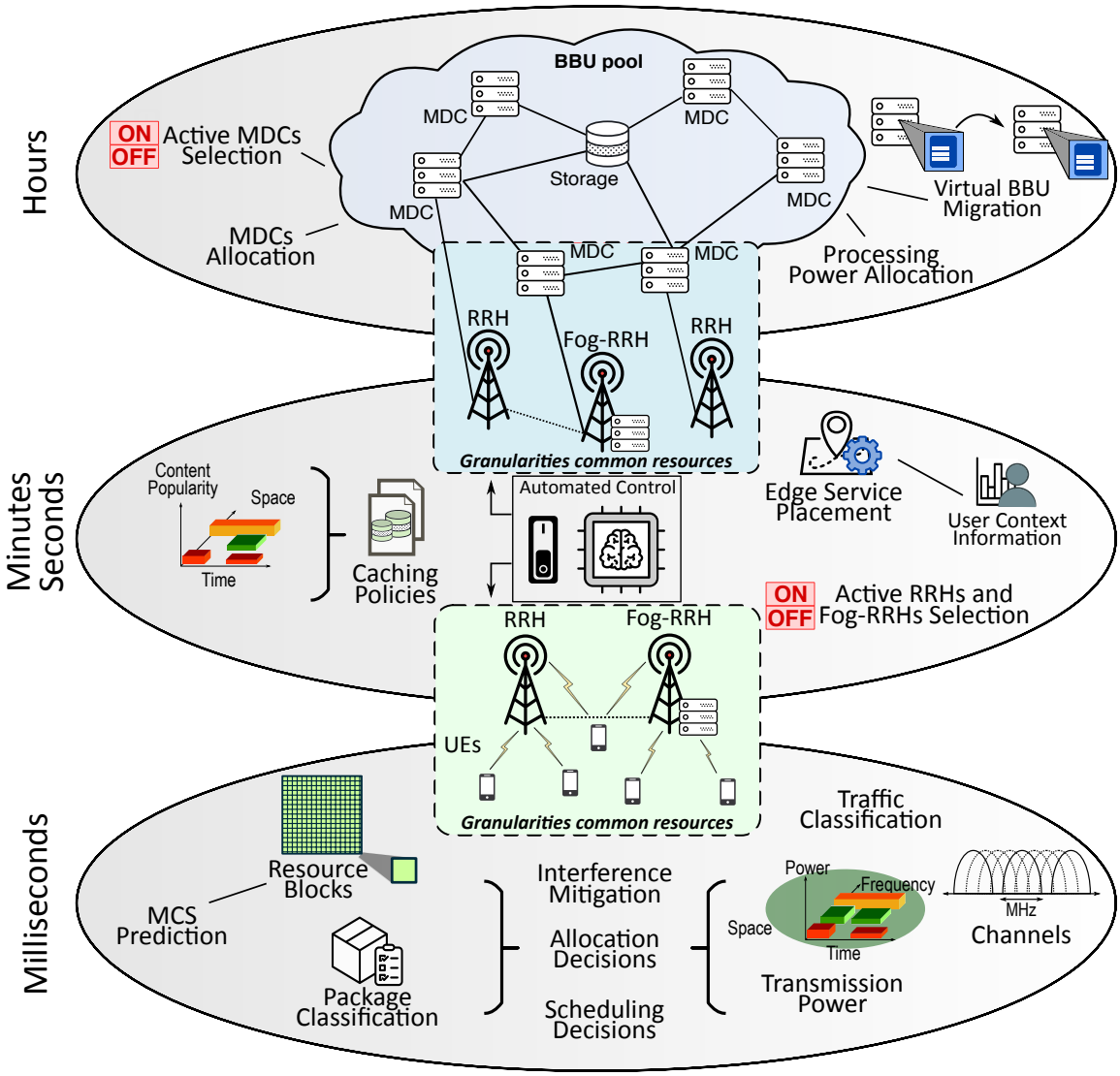

Figure 1.2: Decision-making in F-RAN, considered at different timescales.

the cost of vBBU allocation by optimally deciding the assignments of RRHs to MDCs required to meet the RRHs' workload demand, reducing operator's expenditures and increasing the processing resource availability in the F-RAN.

In this context, through simulations under different F-RAN scenarios, the solution's potential to achieve revenue gains to network operators considering the opportunity of leasing unused processing resources to others is evaluated. As future work, the optimal solution will be used to compare with other solutions using ML, which are currently under development. The ML-based solution will be developed addressing the problem of efficient resource allocation in F-RANs while considering the time constraints.

1.2 Main Contributions

Throughout the development of this work, many contributions were developed in regard to the state-of-the-art of AI in F-RANs. In short, the main contributions of this work include:

1. A systematic investigation of the application of AI techniques in F-RANs and its benefits, classified around three time granularities: hours, minutes/seconds, and milliseconds.
2. A mapping between the time granularity at which resource management decisions must be made in F-RANs and the most effective AI techniques that can be applied.
3. A multiagent-based architecture proposal for integration of AI solutions into F-RANs, at different time granularities.
4. A model to formulate the problem of vBBU optimal allocation, a specific goal from the time granularity of hours, as an RRH-MDC assignment problem for cost minimization, considering the relation between processing power and distance between MDCs and RRHs.
5. The evaluation of the potential operators' expenditure gains in terms of minimal processing power in use considering different scenarios and relying on real demand data provided by an operator.

1.3 Document Outline

This document is organized as follows. The concepts of decision-making and time granularities, presenting a literature review to describe each time granularity are presented in Chapter 2. The time granularities of hours, minutes/seconds, and milliseconds as well as the mapping between decision-making and AI techniques in F-RANs are presented. The multiagent architecture proposal for AI-driven F-RANs, a possible solution for decision-making integration among time granularities, is presented in Chapter 3. The description of the use case considering a time granularity of hours, including the definition of the system model, problem formulation, simulation results, and analysis are presented in Chapter 4. Finally, the conclusions and future works are presented in Chapter 5.

Chapter 2

AI-driven F-RANs Overview: Decision-making, Time Granularities, and ML techniques

This chapter presents the key concepts and a literature review related to AI in F-RANs, aiming to characterize ML-based solutions for decision-making in the network. Based on the literature review, a characterization and discussion regarding three time granularities (hours, minutes/seconds, and milliseconds) are presented. The relationship between AI approaches is discussed as well as the decisions associated with different timescales in F-RANs. Further, a mapping between decision-making and AI techniques in F-RANs is presented.

2.1 Decision-making in Hours

Decision-making that occurs on a timescale of hours/several minutes incorporates elements from cloud and fog, as depicted in Figure 2.1. The BBU pool, RRHs, and Fog-RRHs are the main components involved at this time granularity, and decisions associated with such resources can have significant impact on the RAN operation. For instance, decisions regarding MDC and vBBU allocation will influence all the attached communication, *i.e.,* will impact all the assigned RRHs/Fog-RRHs and any other equipment linked requiring reconfiguration and preparation according to different decision. In this context, decisions will result in service migration, which is a task that can be

set in the granularity of hours [Liu et al., 2018]. Considering such flexible time constraint, it is feasible to apply sophisticated ML techniques to improve the RAN's operation [Yu et al., 2016] [Chien et al., 2019] [Aryal and Altmann, 2018]. We highlight three relevant goals to guide the decision-making on a F-RAN on a timescale of hours: (*i*) optimal allocation of MDCs; (*ii*) processing power minimization; and (*iii*) cost reduction of vBBU allocation. For each goal, we present the main decision involved, ML techniques that can be employed for decision-making, and the reason these techniques are suitable.

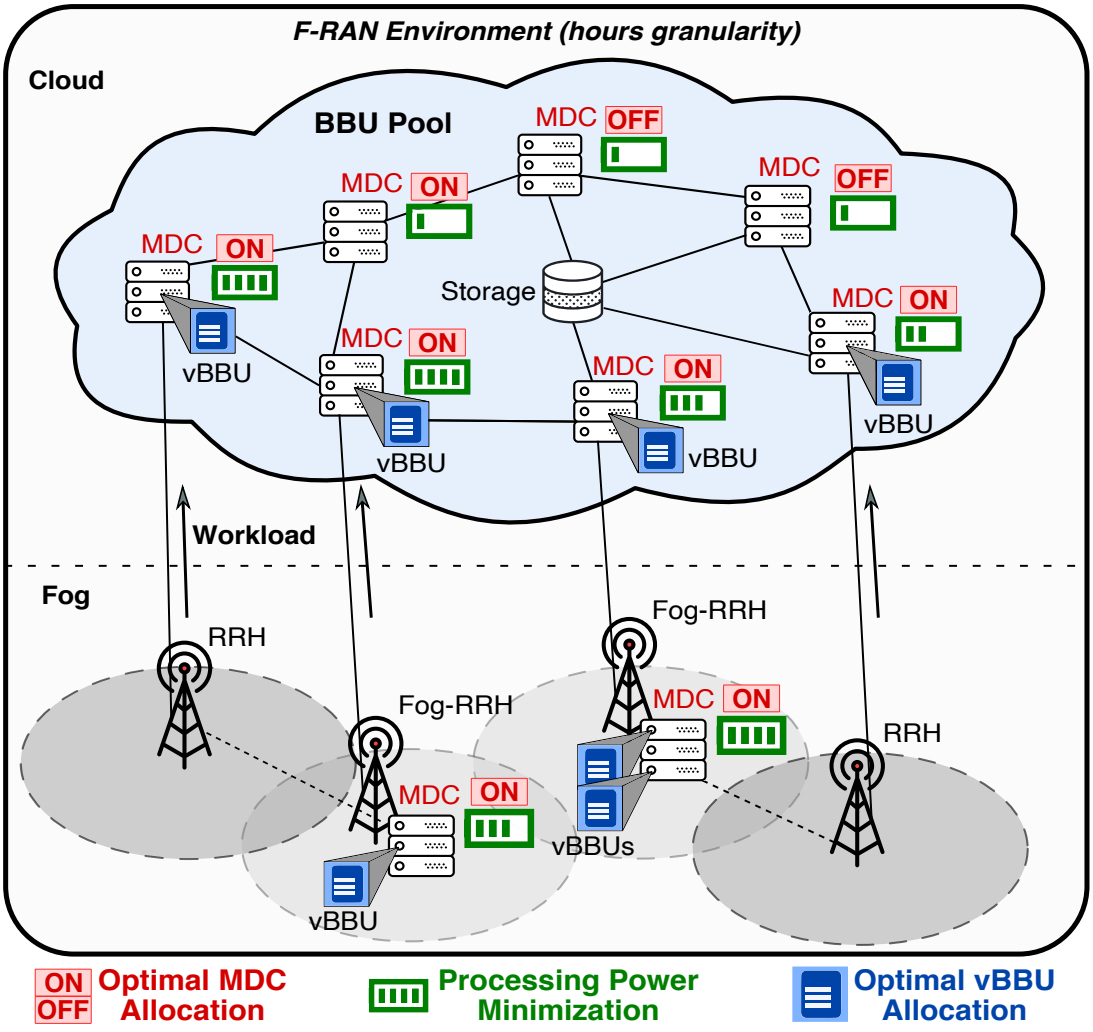

Figure 2.1: F-RAN: resources and decision-making in hours.

2.1.1 Optimal allocation of MDCs

Optimal allocation of MDCs can reduce energy consumption and cost. Decisions consist on selecting the minimal set of active MDCs in the BBU pool that meets the F-RAN

processing demand. Considering the dynamic characteristics of processing demand in F-RANs, accurate prediction of the processing power in use is a key factor to determine the MDCs' future loading. ML techniques capable of calculating regressions during run-time assume an important role in this context, taking several minutes to hours of training.

Deep RNNs are widely used for performing predictions through regressions of time series, which is directly applicable to demand prediction. Such techniques typically cannot process very long sequences and require a prolonged training time. Nonetheless, for the allocation of MDCs, in which the time interval for decision-making is large, the benefits of neural networks outstrip their disadvantages. As an application example, Chien, Lai, and Chao [Chien et al., 2019] presented an solution applying Long Short-Term Memory, a deep RNN, to predict throughput and to allocate MDCs considering a scale of hours.

2.1.2 Processing Power Minimization

Minimizing the processing power in use enables MDCs and Fog-RRHs to direct spare resources to serve new RANs or to be shared among operators. An F-RAN may serve simultaneously urban, residential, and rural areas that present distinct processing demand behavior. ML techniques able to detect and cluster the different patterns of demand can be beneficial for load balancing and processing power allocation. In this context, clustering and neural network techniques can process the data and make decisions that increase the accuracy of workload predictions.

Although neural networks are able to learn an effective model based on the training data and experience, the training process is costly and the number of parameters grows as the number of layers and perceptrons increases. Nonetheless, this technique is suitable for workload demand prediction in RANs, *i.e.,* the benefits of such a technique outweigh its disadvantages. For instance, Yu *et al.* [Yu et al., 2016] presented a solution applying K-means and Multi-Layer Perceptron, examples of clustering and neural network respectively, to detect specific demand patterns of each involved RAN and to minimize the processing power constrained to a scale of hours.

2.1.3 Cost reduction of vBBU allocation

Cost reduction of vBBU allocation enables operators to decrease operational expenditures in F-RANs. In this regard, decisions must be made to allocate the lowest cost set

of vBBUs, by placing them in MDCs that are closer to the edge in Fog-RRHs or in the distant cloud, to process the RRHs' workload. Market and operational factors introduce fluctuations in the price of processing resources. ML techniques able to search the space of potential solutions can make decisions that consider these price fluctuations.

Evolutionary computation is suitable for this purpose, since it can exploit the search space by generating a population of solutions and combining them to create more fitted ones, aiming to achieve optimality. As a counterpoint, the task of setting suitable heuristics and parameters (*e.g.,* population and generation) is not trivial. As consequence, the optimal solution may not be achieved at all times since the results of such techniques depend on the proper algorithm settings. Regardless these limitations, genetic algorithms can be successfully applied to decide how to allocate vBBUs in RANs. For instance, constrained to a timescale of hours, Aryal and Altmann [Aryal and Altmann, 2018] presented valuable results by applying a genetic algorithm, an evolutionary computation approach, to achieve better vBBU allocation while minimizing the costs of operation.

2.1.4 Considerations

Although ML techniques have been broadly applied in the context of RANs, there is still the opportunity to improve the learning process on this timescale by incorporating the output from finer-grained solutions. The different objectives involved in decision-making by F-RANs on a timescale of hours are not mutually exclusive and can be employed together according to the interests of network operators. In this case, F-RANs must present a flexible decision-making architecture capable of selecting and tailoring the AI techniques according to the objectives that are of highest priority. A solution based on multiagent systems enable the integration of different AI techniques in the same framework. To achieve optimal allocation of MDCs, processing power minimization, and vBBU allocation for cost minimization, some agents can implement neural networks or evolutionary algorithms, while others can apply meta-learning algorithms to decide which learning technique to use according to the current scenario, historical data, and the operators' interests. Besides, there is still the opportunity to improve the learning process on this timescale by incorporating the output from finer-grained solutions. More detail on the multiagent architecture as well as the integration among granularities is provided in Section 4.4. Next, some of the objectives and AI techniques that are relevant on a timescale of minutes/seconds are discussed.

2.2 Decision-making in Minutes/Seconds

Decision-making that occurs on a timescale of minutes/seconds involves resources closer to the fog, such as Fog-RRHs, RRHs, and UEs, as depicted in Figure 2.2. Decisions regarding these resources must be made considering unexpected circumstances, such as flash-crowd events, and content popularity variations according to time, space, and context. Considering such dynamic situations, a granularity of hours is no longer suitable for decision-making. In this scenario, ML techniques able to respond considering a time constraint on the scale of minutes/seconds can be effectively applied to support decision-making in RANs [Chen et al., 2018a] [Xu et al., 2017] [Jiang et al., 2019]. We focus on three germane objectives that affect decision-making in F-RANs at this timescale: (*i*) optimal service placement at the edge; (*ii*) enhanced caching hit rate; and (*iii*) optimal usage of RRHs/Fog-RRHs. For each of these objectives, we present the main decision and the related ML techniques that are most suited for those decisions.

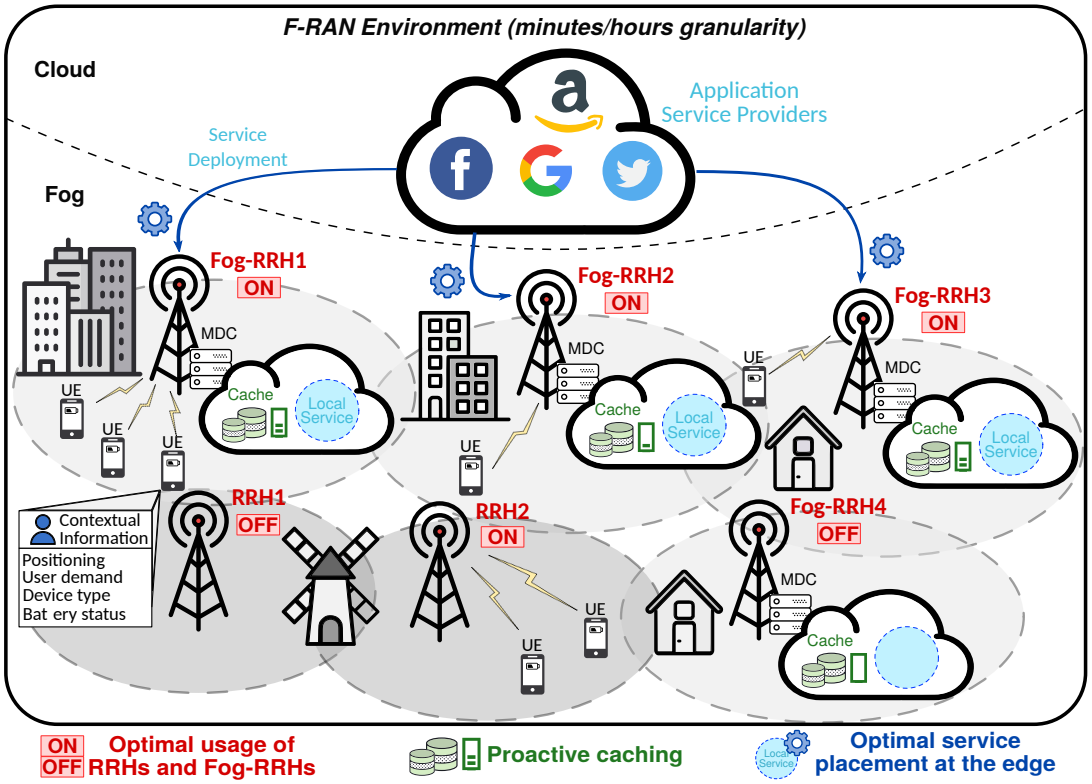

Figure 2.2: F-RAN: resources and decision-making in minutes/seconds.

2.2.1 Optimal service placement at the edge

Optimal service placement is an important goal for ASPs aiming to reduce service deployment costs and improve users' long-term satisfaction. To this end, an ASP must select the optimal set of Fog-RRHs required to deploy services at the edge, considering its limited budget. Nonetheless, Fog-RRHs' resource availability (*e.g.,* processing, memory, and storage) is always in flux and, concomitantly, UEs' contextual information (*e.g.,* connected RRH/Fog-RRH and content demand) also varies significantly on a timescale of minutes. Consequently, ASPs have to adapt their decisions to these scenario dynamics. ML techniques that are able to cross-reference contextual data, resource availability, and ASPs' interests, generating and updating their own model during runtime without several minutes of training, are well suited to this problem.

Contextual reinforcement learning algorithms are widely employed in recommendation systems to process different sources to maximize the average reward of a given objective. Such techniques are able to detect errors and make corrections during the learning process, which makes it highly suitable for environments where the data is constantly gathered. Potential disadvantages of a contextual reinforcement learning algorithm include: the difficulty to set a proper reward function for the algorithm and it requires constant computation and data acquisition. Nevertheless, Chen *et al.* [Chen et al., 2018a] implemented a solution using Bandit Learning, a context reinforcement learning algorithm, based on user feedback for optimal selection of Fog-RRHs in the range of minutes.

2.2.2 Enhanced Caching Hit Rate

Enhancing the caching hit rate in F-RANs can improve network latency and content acquisition time. The challenge is to decide whether to cache a content, via prediction of its popularity. Content popularity varies in space and time, so an accurate prediction considering these two variables is key. ML techniques that can learn and make predictions during runtime are appropriate for improving cache hit rate through caching policy updates in the F-RAN. In this context, adaptive online learning algorithms that also create their model during runtime but aiming for long-term optimality can be exploited.

Online learning algorithms are able to generate sparser solutions and the model can be efficiently updated during runtime, which makes it highly suitable for large-scale learning tasks with real data. When compared with offline approaches, online learning has

an important role in this context because it is able to track the environment changes in real time. However, it is difficult to evaluate the model and it is not trivial to achieve the correct behaviour automatically. Even considering such limitations, online learn algorithms can be effectively used to predict content popularity. As presented by Jiang *et al.* [Jiang et al., 2019], asymptotically optimal performance is achieved by employing Follow-The-Regularized-Leader (a novel optimization algorithm for training deep networks) to predict and track local popularity during runtime, and to update the caching policy within minutes, thus improving the caching hit rate in F-RANs.

2.2.3 Optimal usage of RRHs/Fog-RRHs

Optimal usage of RRHs/Fog-RRHs refers to the selection of the minimal set of active RRHs to serve UEs, meeting their throughput demand. UEs' requests vary significantly within a time interval of minutes or less. Consequently, a solution able to adapt promptly in consonance with this variation is necessary. For instance, reinforcement learning techniques are applied in many domains in which a system interacts with a dynamic environment and learns during runtime constrained to seconds.

Deep reinforcement learning is able to scale highly complex problems by automatically reducing and tuning features direct from inputs in runtime. Such a technique is able to estimate the possible states instead of computing every solution and, consequently, the solution space is pared down during the decision process. The limitation of this technique is that it requires a lot of data and computation in other to achieve effective results, *i.e.,* it will not perform well if there is few data. Nonetheless, the advantages outstrip the disadvantages when applied in the context of usage of RRHs/Fog-RRHs. For instance, Xu *et al.* [Xu et al., 2017] show that a deep reinforcement learning agent can be effectively applied to select the minimal set of active RRHs according to user requests and varying scenario dynamics considering a timescale of seconds.

2.2.4 Considerations

It is noteworthy that the varying behavior of F-RANs hinders the usage of techniques based on recurrent offline training, since the information cannot be well represented by historical data. Even with this limitation, there is still the opportunity to integrate different AI techniques at this time granularity using a multiagent architecture, given that

the presented objectives are not mutually exclusive. Similarly to the timescale of hours, the objectives can be divided among several agents implementing different AI algorithms. As an example, some agents implement reinforcement learning-based algorithms for optimal usage of RRHs/Fog-RRHs while others implement adaptive learning algorithms for content popularity prediction to enhance caching hit rate. More detail on multiagent architecture is presented in Section 4.4. The next sections presents the main objectives and AI techniques considering a timescale of milliseconds.

2.3 Decision-making in Milliseconds

Decision-making that occurs in milliseconds is directly related to the physical layer, involving spectrum and processing resources, including transmission and processing power and resource blocks, as depicted in Figure 2.3. Decisions regarding these resources must consider the high variability of channel conditions caused by interference, noise, and UE mobility. In this context, decisions regarding such low level resources must be made on the scale of milliseconds or even on a shorter timescale (*i.e.,* micro or nanoseconds) [Larsen et al., 2019]. However, ML techniques can be effectively exploited to assist decisions constrained to milliseconds [Wang et al., 2019] [Alqerm and Shihada, 2018] [Imtiaz et al., 2018]. For decisions within a granularity smaller than milliseconds (*e.g.,* nanoseconds), it is not practical to implement an ML solution since it will be restrained to quasi-statistical solutions with drastic computing limitations, approaching the processor's cycle time. In particular, we highlight three relevant goals for decision-making at this timescale: (*i*) optimal spectrum resource allocation; (*ii*) enhanced CPU scheduling; and (*iii*) optimal RRHs-UEs assignment. Considering each of these objectives, we present the main decision and the related ML techniques that are suitable for those decisions.

2.3.1 Optimal spectrum resource allocation

Optimal spectrum resource allocation impacts the F-RANs' energy consumption and spectral efficiency. In this context, the decision is made concerning resource blocks and transmission power allocated for communications between RRHs and UEs, considering QoS requirements. Since F-RANs must employ spectrum reuse in a dense scenario, appropriate resource block and transmission power allocation is needed to avoid inter-tier interference between RRHs/Fog-RRHs. An ML technique capable of acquiring current

state information to quickly assign the available resource blocks and adjust the transmission power to minimize interference is required.

Quasi-statistical model-free reinforcement learning algorithms are widely used to handle problems with stochastic transitions and rewards. A model-free algorithm that can learn without using the current policy nor accurate representation of the environment is suitable for problems where a model of the environment is not available, which is the case here. Additionally, when compared with other learning techniques (such as presented in Sections 2.1 and 2.2), this type of algorithm is considerably less complex in computation and in space. Nonetheless, since there is no defined model, previous experience is required to perform the training task. Alqerm and Shihada [Alqerm and Shihada, 2018] show that a Q-learning agent, a model-free learning approach, is suitable to perform joint allocation of resource blocks and transmission power while mitigating interference and maintaining QoS even with a strict time constraint.

2.3.2 Enhanced CPU scheduling

Enhanced CPU scheduling brings throughput improvements, while tailoring the processing resources to UEs' QoS requirements in F-RANs. Decisions regarding Fog-RRHs' CPUs scheduling may consider the application's delay budget, packet inter-arrival time, and packet length. In this context, ML classifiers that train a model beforehand without online updates are suitable to distinguish between different types of traffic, identifying their patterns in milliseconds. It is worth mentioning that the internal CPU operation (*e.g.,* cache coherence protocols and acceleration features) depends on its specific design/architecture. In particular, the Intel Atom P5900, a 5G chip, is equipped with a hardware-based network acceleration feature for integrated packet processing and a switch for inline cryptographic acceleration. From the perspective of scheduling algorithms development, a higher level context, such specific characteristics can be abstracted or included as parameters.

Support Vector Machines, a broadly applied ML classifier, is effective in high dimensional spaces and suitable for unstructured/semi-structured data. Although it is successfully applied in many domains, it is important to mention that such a technique is not suitable for large and noisy datasets. However, for the task of discriminating between different types of traffic, this technique shows great promise. As an example, constrained to a scale of milliseconds, Wang *et al.* [Wang et al., 2019] report throughput improve-

ments by applying Support Vector Machines to classify the traffic and adapt the CPU scheduling decisions based on the traffic classification and its QoS requirements, as well as the baseband signal processing load.

2.3.3 Optimal RRH-UE assignments

Optimal RRH-UE assignments can reduce the signaling overhead in F-RANs. Decisions are made regarding the best set of assignments among RRHs and UEs to perform communication without interfering with parallel transmissions. F-RANs are employed in very dense scenarios with high interference, resulting in frequent changes in channel state between RRHs and UEs and requiring frequent Channel Quality Indicator (CQI) transmissions. Considering that the primary use of CQIs is to determine the appropriate MCS, ML techniques bring the opportunity to assign RRHs to UEs while predicting the appropriate MCS. ML classifiers with an offline model trained are able to quickly predict MCS based on knowledge about UEs' positions, and past transmission beam, to assign RRHs to UEs, reducing signaling overhead.

The Random Forests algorithm, a supervised ML classifier, is able to create models based on few samples and deal well with missing data and unbalanced datasets. Nonetheless, the control on what the model does is highly limited (black box), turning difficult to improve the performance of the model. This disadvantage must be considered since such technique has a penchant towards over-fit. Additionally, it can consume a lot of memory (complex in space). Even so, it is possible to get valuable results for MCS classification. Imtiaz *et al.* [Imtiaz et al., 2018] proposed an solution based on Random Forests to improve resource allocation based on MCS prediction using UE's information on a scale of milliseconds.

2.3.4 Considerations

It is worth mentioning that, because the training task is performed beforehand, all ML classifiers highlighted above can be used on a timescale of milliseconds. Since the predictions are limited to well-defined categories (*e.g.,* traffic type and MCS indices), the recent data does not necessarily have a considerable impact on the classifier model, and an effective model can be achieved using labeled historical data. Again, there is the opportunity to adopt a multiagent architecture consisting of several agents using distinct

AI techniques. For instance, two agents with the same goal (*e.g.*, traffic classification) can achieve it by applying different algorithms, such as Boosted Trees and Naive Bayes. Furthermore, since this time granularity imposes a strong time constraint, heuristic approaches may become necessary. Thus, a multiagent solution can merge AI techniques and statistical/pure heuristic approaches. Lastly, agents from all the timescales can be integrated into one multiagent architecture. The discussion of such an architecture is presented in Chapter 3.

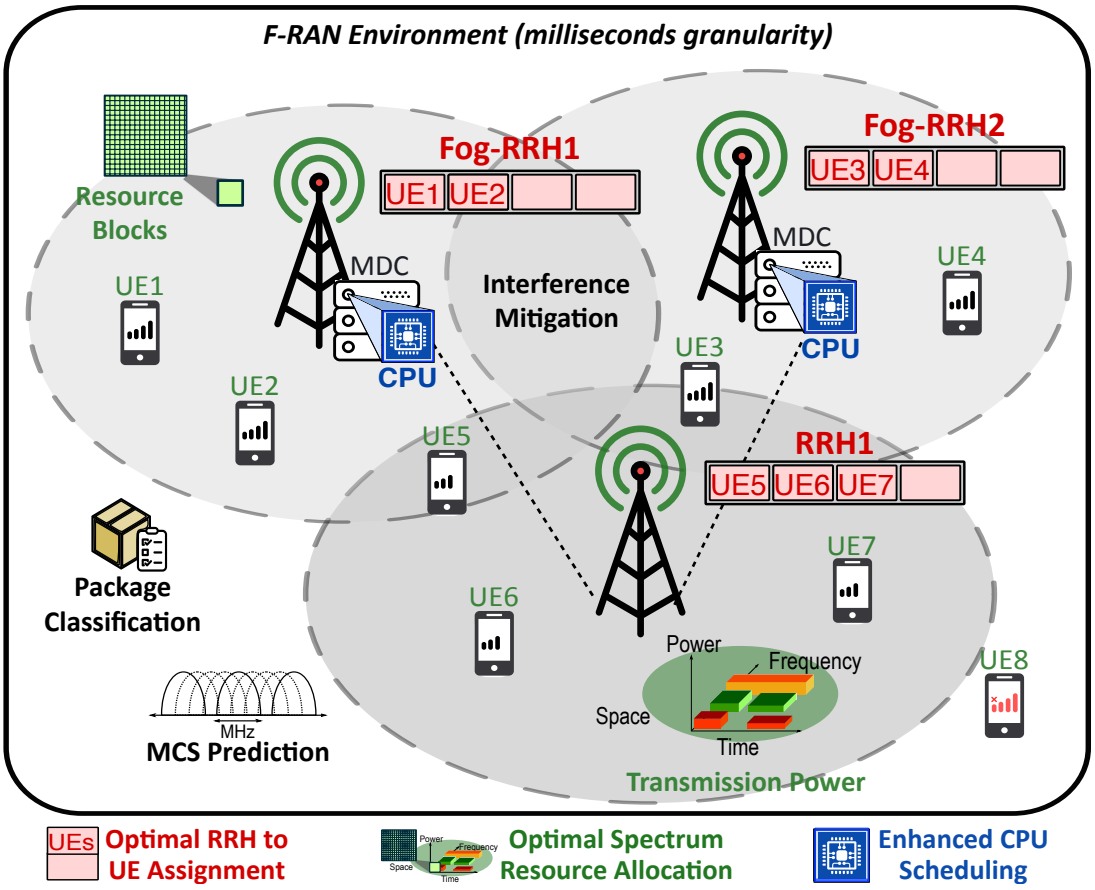

Figure 2.3: F-RAN: resources and decision-making in milliseconds.

2.4 Mapping Between Decision-Making and AI techniques in F-RANs

The discussion presented in Sections 2.1, 2.2, and 2.3 is the result of an extensive investigation of the literature, starting from over 100 articles (from 2016 to 2019), which we filtered down to 30 that specifically dealt with automated solutions for resource man-

agement in RANs. From those, we selected the three most relevant works for each time granularity. Publications highlighted in the previous sections are presented in Tables 2.1, 2.2, and 2.3, presenting a mapping between decision-making, and suitable ML technique for each time granularity. Moreover, these tables summarize the benefits that ML-based solutions bring to RAN operation and the advantages/disadvantages of each technique. Notice that all solutions were able to bring considerable benefits to RANs' performance despite their disadvantages. The complete list of selected articles is summarized in Tables 2.4 (references from 2016 to 2018) and 2.5 (references from 2019).

In Tables 2.1 to 2.3, the main benefits that ML solutions bring to RAN operation as well as the main advantages and disadvantages of each technique are summarized. At the granularity of hours, the main disadvantage of the techniques is the complexity of the training process. However, considering that there is a wide time window for the decision-making process, this disadvantage is compensated by their effectiveness in making high quality prediction. At the granularity of minutes/seconds, the main limitation of the techniques is the requirement of a large amount of data and, sometimes, constant computation/data acquisition. Nonetheless, such advantages are addressed by the plethora of information from RRHs and Fog-RRHs that are constantly gathered. For the three main goals from this granularity, the data from RRHs and Fog-RRHs are being collected constantly and, consequently, there is a plethora of data to be used in the training process. Therefore, considering the quantity of data available and the common advantage of updating and make corrections during run-time, the demand for data is not a great limitation to be considered in such context. Considering the time granularity of milliseconds, the main limitation is the lack of available time and, consequently, training models in real time is not practical. In this sense, this granularity overcome such limitation by applying quasi-statistical techniques or previous trained models. Nonetheless, considering the ML classifiers applied in the decision-making in milliseconds, the main limitations are the lack of performance dealing with large and noisy datasets and their complexity. Further, the complexity of training and model configuration is not a great problem considering that the model is going to be trained beforehand, *i.e.,* the solution will only apply a previous trained model.

Based on the investigation performed, it is possible to conclude that AI solutions have the capabilities to enhance the performance of F-RAN as long as time constraints are met. Moreover, there is an opportunity to integrate different solutions horizontally

Table 2.1: Highlighted works for decision-making on a timescale of hours/several minutes

Reference	**Decisions**	**Techniques**	**Benefits**	**Advantages**	**Disadvantages**
[Chien et al., 2019]	Optimal allocation of MDCs	Long-short term memory (recurrent neural network)	Energy consumption minimization and cost reduction	Can remember the information through time, which turns it a lot suitable for time series prediction	Can not process very long sequences; training process is complex
[Yu et al., 2016]	Cloud workload prediction for optimal vBBU allocation	Multi-layer perceptron (neural network) and k-means clustering	Processing power minimization	Can learn how to act based on the training data and experience; highly suitable for regression tasks	The number of parameters grows quickly; training process is complex
[Aryal and Altmann, 2018]	Allocate the lowest cost set of vBBUs	Genetic algorithm	Reduced RAN's operational expenditures	Can generate many different solutions in considerably short time of computation	Difficult to set a proper heuristic and parameters; it do not reach the optimal solution at all times

(solutions from the same time granularity) and vertically (solutions from different time granularities), which will be discussed in Chapter 3. Although several works have proposed valuable ML-based solutions in RANs, there is still opportunities on integrating different solutions, which are not discussed in any of these works. The next chapter presents the opportunities as well as the related challenges of integrating different solutions.

Table 2.2: Highlighted works for decision-making on a timescale of minutes/seconds

Reference	Decisions	Techniques	Benefits	Advantages	Disadvantages
[Chen et al., 2018a]	Select the optimal set of Fog-RRHs to be rented to deploy services at the edge	Bandit learning (reinforcement learning)	Optimal service placement; deployment costs reduction while attending QoS demand	Can detect errors and make corrections during the training process; suitable for environments where the data is constantly gathered	Hard to define the reward function; requires constant computation and data acquisition
[Jiang et al., 2019]	Decide whether or not to cache specific content via prediction of its popularity in Fog-RRHs	Follow the (proximally) regularized leader (adaptive online learning)	Enhanced caching hit rate	Generates sparse solutions; converges fast; the model can be efficiently updated during runtime	Difficult to evaluate; hard to achieve the correct behaviour automatically
[Xu et al., 2017]	Select minimal set of active RRHs/Fog-RRHs that meets UE requests in terms of throughput	Deep reinforcement learning	Energy consumption minimization	Can reduce the complexity of the solution, enabling to scale complex problems; features are automatically deduced and optimally tuned	Requires a lot of data and computation to overcome other techniques (even more than pure reinforcement learning algorithms)

Table 2.3: Highlighted works for decision-making on a timescale of milliseconds

Reference	Decisions	Techniques	Benefits	Advantages	Disadvantages
[Alqerm and Shihada, 2018]	Resource blocks and transmission power allocation	Q-learning	Enhanced spectral efficiency while mitigating interference and maintaining QoS	Can learn without an accurate representation of the environment; less complex in computation/space	Experience is required for training task; it does not how the dynamics of the environment affects the system
[Wang et al., 2019]	CPU scheduling based on the traffic classification	Support vector machines	Throughput improvements while meeting UE' QoS requirements	Effective in high dimensional spaces; suitable for unstructured and semi-structured data	Not suitable for large and noisy datasets; hard to select the kernel function and tune the parameters
[Imtiaz et al., 2018]	Select the best set of assignments among RRHs and UEs to perform communication via MCS prediction	Random forests	Better spectrum resources allocation; CQI signaling overhead reduction	Can create an effective model with few samples; deal with missing data and unbalanced datasets	Limited control on what the model does; proclivity towards over-fit; complex in space

Table 2.4: Complete mapping of references from 2016 to 2018 considering decision-making on a timescale of hours, seconds and milliseconds.

Reference	**Granularity**	**Decisions**	**Techniques**
[Soliman and Leon-Garcia, 2016]	Milliseconds	Decide if an user should be scheduled.	SVM and Decision Trees
[Xu et al., 2017]	Seconds	Select the minimal set of RRHs to be turned off while meeting UEs' demand.	Deep Reinforcement Learning (DRL) agent
[Nakayama et al., 2017]	Seconds	Traffic routing selection.	Markov Chain Monte Carlo Machine Learning (MCMC-ML)
[Shahriari et al., 2017]	Seconds	Proactive caching	Reinforcement Learning Agent
[Chen et al., 2017]	Minutes	Predict the distribution of requirements and decide contents to store.	Machine learning tools of Echo State Networks and sublinear algorithms
[Tinini et al., 2017]	Hours	Proccess scheduling between nodes.	Integer Linear Programming
[Huang et al., 2017]	Seconds	Infer the terminal (UE) type and effectively decide to allocate training tasks.	Support Vector Machines
[Zhang et al., 2017]	Hours	Minimize total amount of computing resources and load balancing	Genetic Algorithm
[Alqerm and Shihada, 2018]	Milliseconds	Decide spectral resource allocation according to user preferences	Q-Learning Agent
[Imtiaz et al., 2018]	Milliseconds	Select the best set of assignments among RRH and UE to perform communication via MCS prediction.	Random Forests
[Sun et al., 2018]	Seconds	Select and allocate resources for potential pairs of D2D users in the RAN.	Reinforcement Learning
[Chen et al., 2018a]	Minutes	Select the optimal set of Fog-RRHs to be rented to deploy services at the edge.	Bandit Learning (Reinforcement Learning)
[Balevi and Gitlin, 2018]	Hours	Determine the position of Fog-RRHs in order to maximize the throughput.	Clustering - distance-based K-means
[Du and Nakao, 2018]	Milliseconds	Classify the uplink/downlink packets using information about the aplication.	Deep Learning
[Chen et al., 2018b]	Seconds	Decides to offload the computation or decide to compute locally.	Deep Reinforcement Learning (DRL)
[Zhou et al., 2018]	Minutes	Proactive caching based on the previous uses' requisitions and preferences.	Deep Reinforcement Learning (DRL)
[Rahman et al., 2018]	Minutes	Decide the number of tasks to compute in the edge or in the cloud processors.	Heuristics
[Yan et al., 2018]	Minutes	Select which algorithm to apply for resource allocation	Heuristics

Table 2.5: Complete mapping of references from 2019 considering decision-making on a timescale of hours, seconds and milliseconds.

Reference	**Granularity**	**Decisions**	**Techniques**
[Jiang et al., 2019]	Minutes	Proactive caching	FTRL-Proximal
[Sun et al., 2019a]	Hours	Minimize the number of active RRHs to minimize the energy consumption.	Relational Reinforcement Learning, Online K-Means Clustering
[Chien et al., 2019]	Hours	Optimal allocation of MDCs by predicting the future throughput and decide to turn off an BBU pool or load balancing.	Long Short-Term Memory (LSTM), Genetic Algorithm
[Sun et al., 2019b]	Minutes	Decide the UE modes (D2D or RAN). Decide processors' on/off states.	Deep Reinforcement Learning
[Gao et al., 2019]	Seconds	Positioning of BBUs Path selection.	Deep Reinforcement Learning (DRL)
[Lu et al., 2019]	Minutes	Find the optimal cache policy.	Q-Learning with value function approximation (Q-VFA-learning)
[Girgis et al., 2019]	Minutes	Proactive caching.	Statistical approach.
[Moon et al., 2019]	Minutes	Adaptive selection of backhaul and fronthaul transfer modes.	Online Reinforcement Learning
[Jiang et al., 2019]	Minutes	Content popularity prediction in order to decide to cache a content.	Deep Q-Learning
[Mao and Yan, 2019]	Milliseconds	Channel estimation.	Deep Learning
[Nassar and Yilmaz, 2019]	Minutes	Decide to serve an user locally or to refer it to the cloud.	Markov Decision Process with Reinforcement Learning

Chapter 3

Multiagent Architecture for AI-driven F-RANs

In this chapter, a proposal of a multiagent architecture integrating AI techniques at different time granularities to improve performance in F-RANs is presented. Section 3.1 introduces the multiagent architecture, presenting, for each time granularity, the impacts of the decision-making considering horizontal (at the same granularity) and vertical (at different time granularities) communication/integration. Section 3.2 discuss a communication and interaction protocol for the multiagent system.

3.1 Multiagent Architecture Proposal

A multiagent architecture can be designed to integrate AI techniques for multi-level decision-making F-RANs, as depicted in Figure 3.1. The proposed architecture comprises nine different types of intelligent agents within three time granularity layers: hours, minutes/seconds, and milliseconds. Each time granularity has three types of agents. The hours layer is composed of cloud managers, vBBU manager, and meta-learner. The minutes/seconds layer is composed of fog manager, QoS manager, and context manager. The milliseconds layer is composed of RRH-UE manager, CPU scheduling, and spectrum resources manager. Furthermore, there is a shared Knowledge Base (KB) in each layer to store all the information gathered from the environment as well as data regarding resources and decision-making on each timescale. Next, the architecture is described, considering the agents' behavior and capabilities.

3.1.1 Top Layer: Hours Granularity

In the top layer (hours), the cloud managers are responsible to allocate MDCs, selecting the optimal set of active MDCs. The vBBU manager is responsible for cost reduction in vBBU allocation, considering the distance between MDCs and RRHs. Both agents communicate with the meta-learner, which is responsible for selecting the most suitable AI technique to apply according to the current scenario. Therefore, the meta-learner does not act directly into the F-RAN environment but systematically observes the performance of different AI techniques in use at this layer to decide which one to apply. Although this kind of AI technique is promising for decision-making improvements, it is not feasible to be applied at finer time granularities, presenting a special opportunity on the timescale of hours.

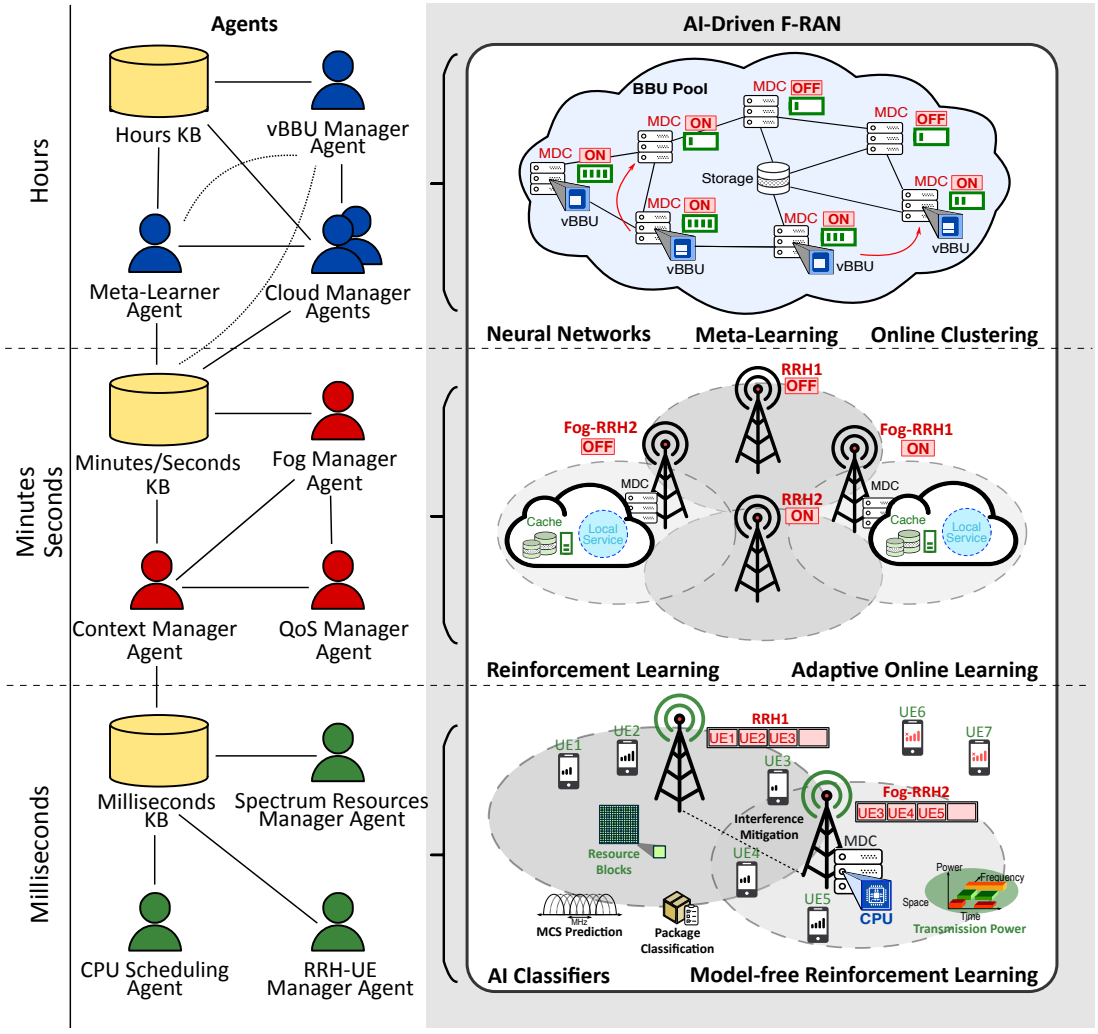

Figure 3.1: Multiagent architecture proposal for AI-driven F-RAN

In this layer, cloud and vBBU managers present different goals that may conflict due to the allocation trade-off regarding distance, cost, and number of cores in use, such as investigated in a previous work [Marotta et al., 2018], being depicted in Figure 3.2. To avoid such conflicts, agents must communicate with each other to arrive at a joint decision. The joint decision comprises vBBU allocation and migration operations, guided by the operator's business model. This business model can be represented as a utility function or weighted objective sum, to balance the interests of the operator aiming for equilibrium, at the intersection of the two curves in Figure 3.2. The two curves illustrate the influence of distance between RRH and MDC on the minimal number of processing cores allocated and the processing cost per hour, illustrating an important trade-off for F-RAN. Notice that these results consider the processing of seven iterations of a Forward Error Correction (FEC) decoder, an RRH with fixed upstream data rate of 10Mb/s, an MDC comprised of processors with an efficiency of 8 operations per cycle and 3.4 GHz per core, and allocation price of $0.0425/hour/core as defined by Amazon [Amazon Web Services, 2019].

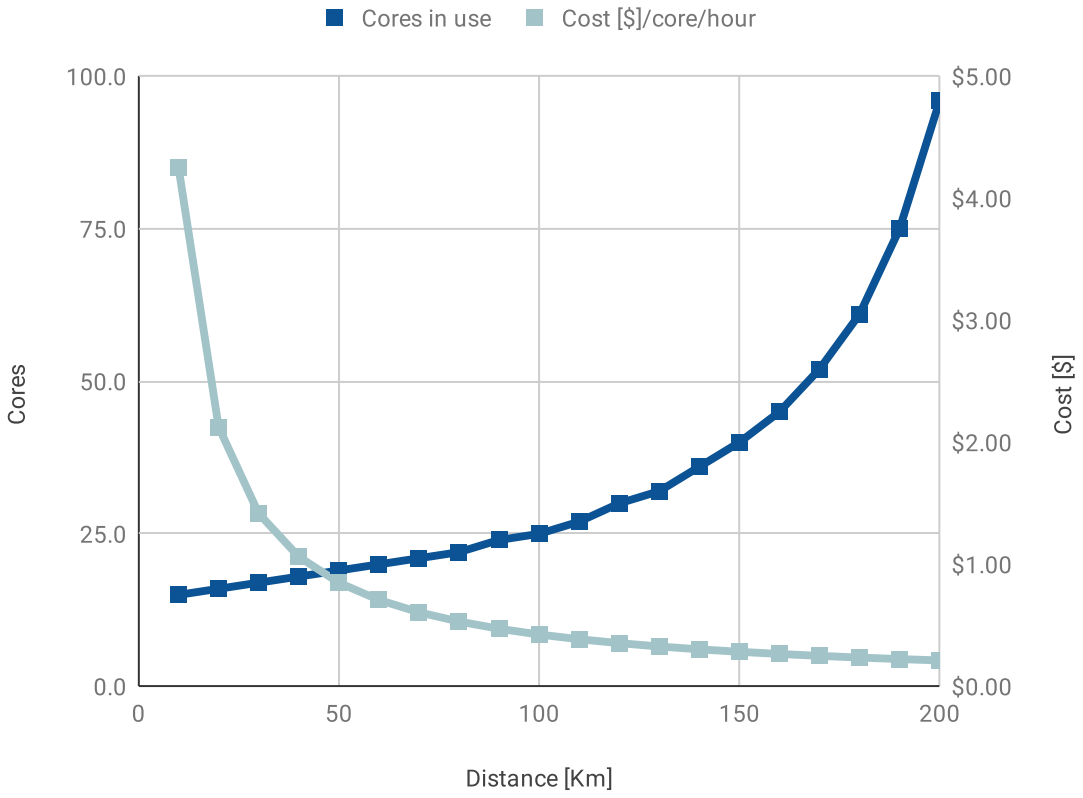

Figure 3.2: Relation betwenn distance between RRH and MDC on the minimal number of processing cores allocated and the processing cost per hour.

Further, all the mentioned agents are able to access the hours and minutes/seconds

KB to get data from both timescales, enabling horizontal and vertical integration. As an example of the potential of using data from another timescale, let us consider the situation of flash crowd events, such as a concert. While the occurrence and duration of such events is on a timescale of hours, once the event starts, further fine grained resource allocation decisions must be made on a timescale of minutes. In this case, the agents can get information from the minutes/seconds knowledge base for these finer-grained decisions.

3.1.2 Middle Layer: Minutes/Seconds Granularity

In the middle layer (minutes/seconds), the fog manager implements an AI technique able to select the minimal set of active RRHs to serve a certain area. The QoS manager is in charge of selecting MDCs and Fog-RRHs for ASPs to deploy their services, as well as of updating caching policies during run-time. These two agents require different parameters for decision-making, such as content popularity, UEs' information, and resource availability. Considering these parameters, the context manager is in charge of monitoring the scenario dynamics in terms of content, demand/popularity, and Fog-RRHs resource usage. More than a simple monitor, this agent's goal is to create predictions of content popularity, using adaptive online learning, and track contextual data, such as user mobility and positioning. Although this agent does not interact directly with F-RAN, it acts as an assistant to other agents from the same layer, enabling horizontal integration.

In this layer, the fog manager and the QoS manager goals may conflict. The fog manager may decide to turn off one or more RRHs that are near MDCs, which were selected by the QoSs manager to deploy the ASP's local services for their users. Communications between both agents enable system adaptation to arrive at a shared decision according to operators' interests, enabling a trade-off between decreasing energy consumption costs or increasing revenue through MDCs service hosting. For example, a Foundation for Intelligent Physical Agents (FIPA) Contract Net Interaction Protocol[1] may be settled during runtime by adjusting the price for service deployment to cover the energy costs related to sustain an RRH turned on.

The fog manager is also able to access the KB from the layer of milliseconds, which has information regarding traffic type classification and can enable vertical integration between these two timescales. This agent is able to access the KB from the layer of milliseconds, which has information regarding traffic type classification and can enable

[1] Available in `http://www.fipa.org/specs/fipa00029/SC00029H.html`, last accessed on 01-18-2020.

vertical integration between these two timescales. As an example, consider that the context manager can get data regarding traffic type classification from the bottom layer's KB and use that information to improve its contextual data about the scenario in which it operates. Therefore, traffic classification assists both the context manager in monitoring the environment and, consequently, the QoS manager for decision-making regarding local service placement. For instance, if there is a location with high traffic for a specific kind of content, such as streaming, the QoSs manager may adapt its decision to deploy streaming services at MDCs from this location.

3.1.3 Bottom Layer: Milliseconds Granularity

In the bottom layer (milliseconds), the spectral resources manager is in charge of performing joint allocation of resource blocks and transmission power. The RRH-UE manager decides the assignment of RRHs and UEs based on MCS prediction. The CPU scheduling agent's goal is to classify the type of traffic to make scheduling decisions that increase overall throughput. Considering the limited time available for decision-making, all these agents can access only KB from the layer of milliseconds (*e.g.,* a shared memory or storage from an MDC).

In this layer, the RRH-UE manager and spectral resources manager share a trade-off in common regarding resource block allocation, enhanced channel capacity, and number of served UEs. Spectrum is a finite resource, sometimes requiring decisions of whether to serve a new UE or to provide enhanced channel capacity to already connected ones. At least one of the agents must receive information from the other to reach a shared decision regarding the resource block assignment and UEs being served. The shared decision must be ranked according to a performance objective (*e.g.,* the highest overall throughput or the fairest scheduling). Since communications between agents at this time granularity is not feasible, they must store their outcome in KBs (blackboard protocol). One of the agents can use the output of the other, readjusting its objective to aim for the best outcome.

The strong time constraint of a few milliseconds prevents any direct interaction between the agents at this granularity and the higher ones. In this sense, only indirect interactions are available, considering higher time granularity outcomes as information stored in the shared KB. For instance, let us assume that a fog manager from the minutes/seconds granularity determines the shutdown of an RRH. For the RRH-UE manager

at the milliseconds granularity, it will be considered a scenario parameter gathered from the milliseconds KB that will prevent the association with that RRH, when executing the Q-learning algorithm.

3.2 Communication and Interaction Protocol

To communicate, exchange, and understand messages in a multiagent environment, agents use a common set of terms through an Agent Communication Language (ACL), the standard language proposed by the FIPA, or other languages such as the knowledge query and manipulations language (KQML), knowledge interchange format (KIF), or more formal specifications such as ontologies. Considering the architecture presented in Figure 3.1, horizontal exchange of messages is accomplished by using FIPA ACL to inform and request basic performatives within the three layers. For example, in the hours layer, an inform performative with the statement content exchanged between the cloud manager and the vBBU manager is used to share information regarding the set of active MDCs. The vertical integration is accomplished through the stored agents' decisions/predictions at the shared KB from each layer.

An interaction protocol must be used for the structured exchange of messages between agents. Autonomous agents can have conflicting goals or simply be self-interested. In this case, utility functions are used to maximize payoff, *e.g.,* vBBU allocation and migration operations. In other instances, agents can have similar goals, so the objective is to maintain globally coherent performance without violating the autonomous behavior of agents, by determining shared goals and common tasks, avoiding unnecessary conflict, and pooling knowledge and evidence. For instance, the context manager sends contextual information and content popularity predictions to assist the decision-making of the QoS manager. There are many interaction protocols used in multiagent systems, including coordination, cooperation, contract net, and negotiation.

Considering the contract net protocol, a widely applied task-sharing approach, there are five stages: recognition, announcement, bidding, awarding, and expediting. Suppose there are two agents that cannot achieve the goal in isolation (*e.g.,* typically because of solution quality or deadline). In such cases, specific roles are assigned to the two agents: (*i*) the manager, which is responsible to announce a task, receive/evaluate bids, award a contract, and receive results; (*ii*) the contractor, which is responsible to receive

task announcements, evaluate/respond/decline/perform the task, and report results. For example, the contract net protocol can be used to setup an agreement between the fog and QoS managers such that the cost of leaving RRHs turned on is paid with the income adjustment of hosting ASPs services. To perform this process among middle layer agents, it is possible to use a cloud storage service, while for the bottom layer the communications occurs in the shared memory at MDC of an F-RAN, since the time interval available for decision-making is highly constrained.

It is worth mentioning that increasing the number of agents has its consequences, and the communication among agents may become a bottleneck. Therefore, the horizontal and vertical integration may be compromised, mainly in the bottom layer. To mitigate this issue, the integration among time granularities through KB sharing (*i.e.,* a shared cloud storage for decision-making and its related information) is proposed. This shared KB enables agents from coarser granularities to access the finer-grained data without compromising their performance with a complex communication and interaction mechanism.

Chapter 4

Use Cases: Granularity of Hours and Decomposed Time Granularities

In this chapter, an use case from the time granularity of hours is presented. First, the system model is defined followed by the problem formulation and evaluation followed by results analysis. Lastly, another use case is presented in order to illustrate the benefits of decomposing time granularities.

4.1 System Model

The defined system model of an F-RAN is depicted in Figure 4.1 and the notation is presented in Table 4.1. An F-RAN is composed of S = $|\mathcal{S}|$ MDCs, where $\mathcal{S}$ is the set containing MDCs from macrocells and from smallcells, *i.e.,* MDCs from Fog-RRHs. Note that the operator $|\cdot|$ represents the cardinality of a set. Each MDC $s \in \mathcal{S}$ has g_{is} General Purpose Processors (GPPs) for different classes of vBBUs $i \in \mathcal{I}$ responsible for the remote processing of the workload of M = $|\mathcal{M}|$ RRHs, where $\mathcal{M}$ is the set containing RRHs from macrocells and RRHs from small cells that are serving UEs in the RAN within a defined time horizon $\mathcal{T} = \{1, 2, \cdots\}$. RRHs from macrocells are separated by a distance d_{macro} (or d_{sm}), in meters, where $\frac{d_{macro}}{2}$ determines the coverage radius of each macrocell. The total workload demand of an RRH for one second of operation is represented as $\Gamma_m(t)$, in bits.

The different classes of vBBUs within MDCs are responsible for processing the RRHs' workload using different types of GPPs. For each class $i \in \mathcal{I}$ of vBBU, the technical

specifications of each GPPs used are the following: ceiling processing power f_{is} in hertz (Hz), efficiency e_{is} operations per cycle, number of cores p_{is}. Moreover, each class of vBBU has a proportion of cost k_{is} per time slot t, according to its individual specification. A vBBU class within an MDC performs processing for signal demodulation, radio resource demapping, precoding, and channel coding. It is noteworthy that the largest component of the processing workload is due to the decoding function of the FEC [Bhaumik et al., 2012]. In this context, w stands for the number of operations per bit for channel decoding processing.

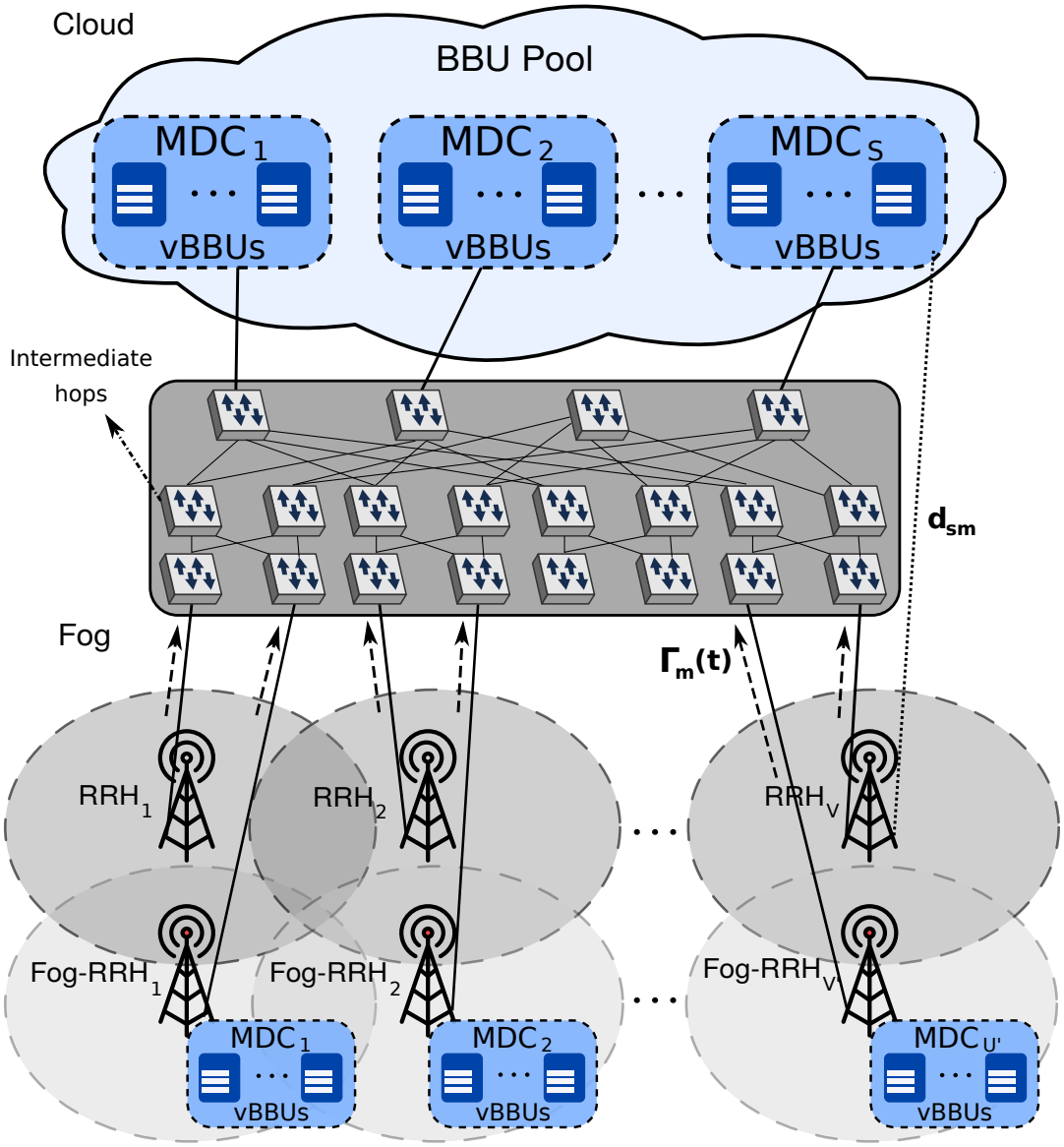

Figure 4.1: F-RAN: system model.

Table 4.1: Notation

Symbol	Description
$\mathcal{S}$	Set of MDCs from macrocells and MDCs from smallcells
$\mathcal{I}$	Set of classes of vBBUs
$\mathcal{M}$	Set of RRHs and Fog-RRHs
$\mathcal{T}$	Finite horizon of operation
c	Speed of light in meters per second
d_{macro}	Distance between RRH from macrocells in meters
w	Number of instructions for channel decoding process
φ	Delay budget according to the technology in use
h	Distance for signal regeneration in meters
λ	Delay per hop in seconds
g_{is}	Number of processors per MDC
f_{is}	Total processor frequency in Hz
e_{is}	Processor efficiency in operations per cycle
p_{is}	Total processor cores
k_{is}	Processor cost per hour in USD
k_{lease}	Operator's leasing price per core per hour in USD
d_{sm}	Link distance between MDC and RRH
$\Gamma_m(t)$	Workload per RRH per slot t in Megabit per second (mbps)
$n_{ism}(t)$	Minimum number of cores to process a workload from an RRH per slot t
$n'_{ism}(t)$	Minimum number of vBBUs to process a workload from an RRH per slot t
$a_{ism}(t)$	Decision variable for assignment among vBBU class, MDC, and RRH per slot t
$A_{cores}(t)$	Total number of cores in the system per slot t
$U_{cores}(t)$	Total number of idle cores in the system per slot t
$K_{alloc}(t)$	Cost of allocation per slot t
$G_{lease}(t)$	Maximum income (by leasing idle cores) for each slot t

FEC is processed at MDCs for the total data received from all assigned RRHs. To set an assignment between an MDC s and an RRH m at a time slot t, the assigned MDC must be able to process the workload demand $\Gamma_m(t)$ of the RRH m, *i.e.,* MDC must have GPPs available to allocate the required number of vBBUs. From Marotta *et al.* [Marotta et al., 2018], it is possible to obtain the minimum number of cores $n_{ism}(t)$ of an MDC s required to process the workload $\Gamma_m(t)$ of an RRH m transmitted in one second of operation as a function of their distance d_{sm} at a time slot t:

$$n_{ism}(t) = \begin{cases} \frac{\Gamma_m(t)w}{f_{is}e_{is}\left(\varphi - \frac{3d_{sm}}{c} - \frac{2\lambda d_{sm}}{h}\right)}, & \text{if } \left(\varphi - \frac{3d_{sm}}{c} - \frac{2\lambda d_{sm}}{h}\right) > 0 \\ -\infty, & \text{otherwise} \end{cases} \tag{4.1}$$

where φ stands for the delay budget according to the technology in use (*e.g.,* Hybrid Automatic Repeat reQuest (HARQ) imposes 3 milliseconds for the Long Term Evolution (LTE) processing [China Mobile Research Institute, 2011]); d_{sm} stands for the distance, in meters, between each MDC $s \in \mathcal{S}$ and RRH $m \in \mathcal{M}$; h stands for the distance, in meters, for signal regeneration, which determines the number of intermediate nodes (hops) required; λ stands for the delay per hop in seconds; and c is the speed of light in meters per second. Notice that $n_{ism}(t)$ is set to minus infinity when the distance between the RRH and MDC is too long, considering that narrow delay constraints dictate the maximum distance between an RRH and MDC to process to its workload [Musumeci et al., 2016]. In this case, the minus infinity is to represent that is not possible to process the RRH's workload in the MDC, *i.e.,* the distance limits the area that an MDC can serve.

Considering that each GPP has a specific number p_{is} of cores, it is possible to divide $n_{ism}(t)$ by p_{is} to get the minimum number of vBBUs $n'_{ism}(t)$ required to process the RRH's workload transmitted per time slot averaged for one second of operation:

$$n'_{ism}(t) = \begin{cases} \frac{\Gamma_m(t)w}{f_{is}e_{is}p_{is}\left(\varphi - \frac{3d_{sm}}{c} - \frac{2\lambda d_{sm}}{h}\right)}, & \text{if } \left(\varphi - \frac{3d_{sm}}{c} - \frac{2\lambda d_{sm}}{h}\right) > 0 \\ -\infty, & \text{otherwise} \end{cases} \tag{4.2}$$

Each MDC can support at most g_{is} vBBUs to run in the same time slot t, *i.e.,* $n_{ism}(t) \leq g_{is}, \forall i \in \mathcal{I}, \forall s \in \mathcal{S}, \forall m \in \mathcal{M}, \forall t \in \mathcal{T}$. Hence, the problem of vBBU allocation is addressed considering the distance between MDC and RRH when deciding which vBBU class to allocate to process the RRHs' workload. Lastly, the total number of cores for the entire

RAN is given by $A_{cores}(t) = \sum_{i=1}^{\mathrm{I}} \sum_{s=1}^{\mathrm{S}} p_{is} g_{is}$, which $A_{cores}(t)$ is the total number of cores of all GPPs from all MDCs at each time slot t. Considering the system model presented, the problem of optimal vBBU allocation in F-RANs is formulated in the next section.

4.2 Problem Formulation

The goal of this part of the work is to optimize the operator's allocation decisions to minimize their expenditures by deciding how to best assign RRHs' workload to MDCs. Note that the distance between RRH and MDC is the factor that most impacts the vBBU allocation cost since it determines the minimum computation resources required to process an RRH's workload. Besides, it is important to consider limitations regarding horizontal allocation, *i.e.,* allocation of MDCs to process workloads in parallel, and vertical allocation, *i.e.,* processing allocation required to process RRHs' workload considering delay thresholds from the wireless stack [Marotta et al., 2018]. Furthermore, the workload of an RRH must always be served by an MDC, *i.e.,* it will always be possible to assign an RRH's workload to an MDC. Therefore, the optimization problem to realize the proposed strategy is formulated as:

$$\min_{\forall t \in \mathcal{T}} \sum_{i=1}^{\mathrm{I}} \sum_{s=1}^{\mathrm{S}} \sum_{m=1}^{\mathrm{M}} a_{ism}(t) n'_{ism}(t) k_{is} \tag{4.3a}$$

$$s.t. \qquad \sum_{m=1}^{\mathrm{M}} n'_{ism}(t) a_{ism}(t) \leq g_{is} p_{is} \qquad \forall i \in \mathcal{I}; \forall s \in \mathcal{S} \tag{4.3b}$$

$$\frac{\Gamma_m(t) w}{\left(\varphi - \frac{3 d_s m}{c} - \frac{2 d_s m}{\lambda}\right)} - a_{ism}(t) f_{is} e_{is} n'_{ism}(t) \leq 0 \tag{4.3c}$$

$$\forall i \in \mathcal{I}; \forall s \in \mathcal{S}; \forall m \in \mathcal{M}$$

$$\sum_{i=1}^{\mathrm{I}} \sum_{s=1}^{\mathrm{S}} a_{ism}(t) = 1 \qquad \forall m \in \mathcal{M} \tag{4.3d}$$

In this case, $a_{ism}(t)$ is a decision assignment binary variable such that

$$a_{ism}(t) = \begin{cases} 1, & \text{if a vBBU class } i \text{ within an MDC } s \\ & \text{is assigned to an RRH } m \text{ at a slot } t \\ 0, & \text{otherwise} \end{cases}$$

and k_{is} is a scenario parameter for the cost of allocation for each class i of vBBU within an MDC s.

Equation 4.3a (the objective function) aims to minimize the sum of the system's processing power allocation cost for each vBBU class within each MDC that is computing RRHs' workloads per time slot t, subject to three constraints. Equation 4.3b represents the constraint regarding horizontal allocation, *i.e.,* an MDC must be able to process all the assigned workload, which restricts the vBBU allocation according to the available computational power considering the combination of workload assignment for each RRH m, vBBU class i, and MDC s. Equation 4.3c represents the constraint regarding vertical allocation, *i.e.,* the processing power required to compute the workload must be less than or equal to the available processing power within the system's MDCs, considering each vBBU class, each MDC, and each RRH. Finally, Equation 4.3d is the constraint to assure the assignment of all RRHs' workloads to at least one MDC, ensuring that all the RAN's workload is going to be processed, for all RRHs.

From the decision variable $a_{ism}(t)$, an equation for the total number of idle cores $U_{cores}(t)$ at each time slot t is formulated, which is obtained by subtracting the total number of assigned cores to compute RRHs' workload from $A_{cores}(t)$:

$$U_{cores}(t) = \sum_{i=1}^{\mathrm{I}} \sum_{s=1}^{\mathrm{S}} \sum_{m=1}^{\mathrm{M}} A_{cores}(t) - a_{ism}(t) \left(p_{is} n'_{ism}(t)\right) \qquad \forall t \in \mathcal{T} \qquad (4.4)$$

Also from $a_{ism}(t)$, an equation for the total cost of vBBU allocation $K_{alloc}(t)$ at each time slot t is formulated, such that:

$$K_{alloc}(t) = \sum_{i=1}^{\mathrm{I}} \sum_{s=1}^{\mathrm{S}} \sum_{m=1}^{\mathrm{M}} a_{ism}(t) \left(k_{is} n'_{ism}(t)\right) \qquad \forall t \in \mathcal{T} \qquad (4.5)$$

which gives the total cost of allocation required to serve all the system's RRHs for each time slot t, achieved by multiplying the number of cores allocated to serve the system's

workload demand by the cost of allocation per core per unit time.

Following the problem formulation presented, a real demand data provided by a network operator is applied to evaluate the optimal allocation of vBBUs. Next, details of the simulations performed in this work are presented as well as the results for vBBU allocation in terms of cost minimisation.

4.3 Evaluation

In this section, the data set used in the experiments is described and the simulation scenarios are characterized.

4.3.1 Data Set

The Milano Call Detail Record (CDR) data set [Telecom Milano,] is employed in the experiments. This is a real telecommunications data set from Telecom Italia, a cellular network operator. The data is geo-referenced and is processed from CDRs of their subscribers residing in a metropolitan area of Milan, Italy. It is temporally divided into 62 files, one for each day (from November 1, 2013 to January 1, 2014). This data set provides temporal and spatial information, including SMS, call and Internet traffic activity data during a two-month period for the city of Milan. The city of Milan is represented as a grid containing 10,000 squares (100x100 grid), each with an area of 55,225 m^2. The data set is spatially represented in Figure 4.2 using geojsonio [Chamberlain and Teucher, 2020] with OpenStreetMap [OpenStreetMap contributors, 2017].

For each grid, the following CDR information is presented in the original files:

- Square ID: indicates the identification number of the square grids.
- Time Stamp: indicates the data recorded over an interval of 10 min.
- Inbound Call Activity: indicates the duration of the inbound call at a particular grid within time slot of 10 minutes.
- Outbound Call Activity: indicates the duration of the outbound call at a particular grid within time stamp of 10 minutes.
- Inbound SMS Activity: indicates the duration of the inbound SMS at a particular grid within time slot of 10 minutes.

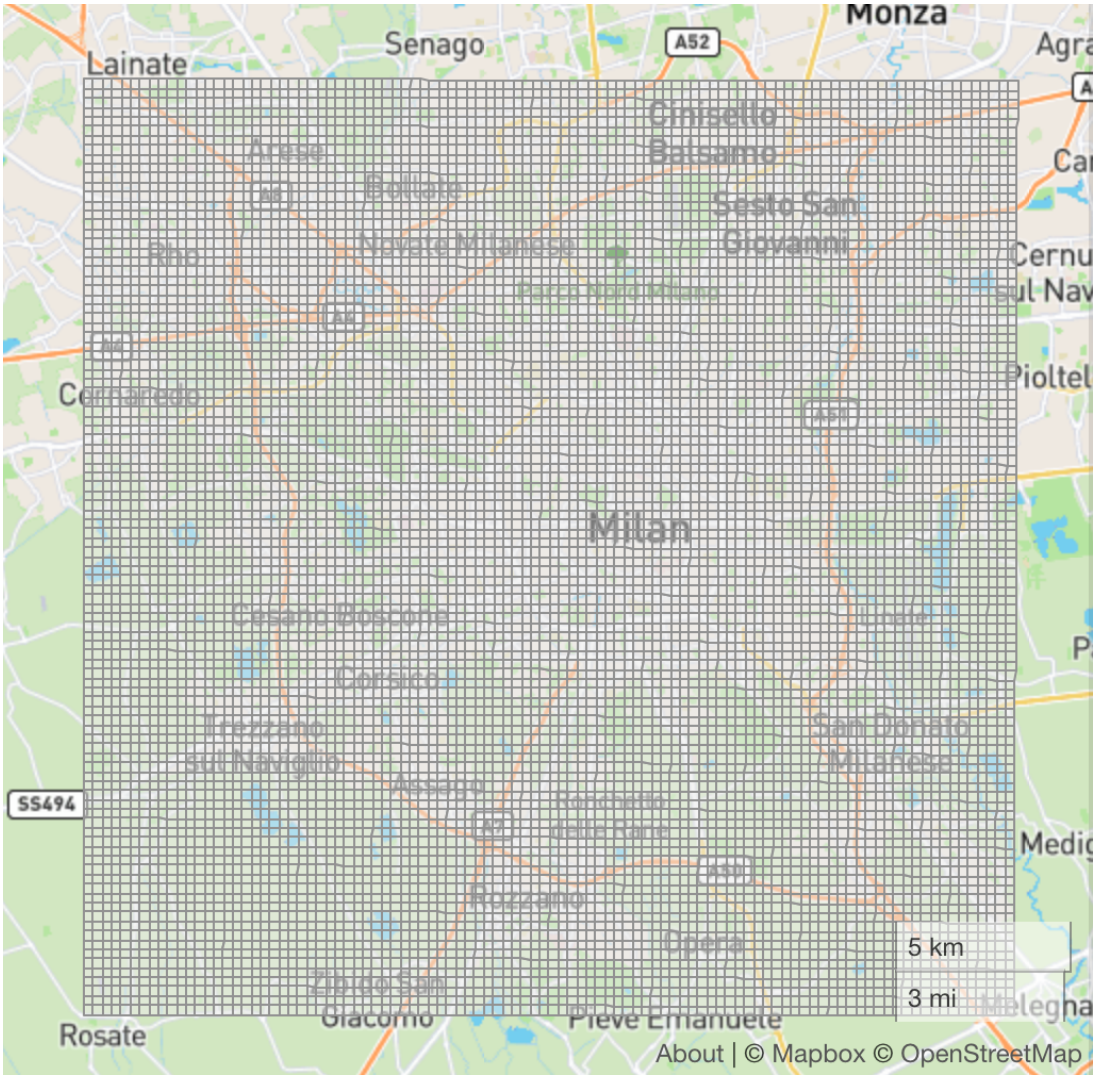

Figure 4.2: Spatial representation of Milano Grid data set.

- Outbound SMS Activity: indicates the duration of the outbound SMS at a particular grid within time slot of 10 minutes.
- Internet Activity: indicates the duration of the Internet activity at a particular grid within time slot of 10 minutes.

After preprocessing and cleaning, the CDR information used in the data set are the following:

- X Position: indicates the X position for a particular region.
- Y Position: indicates the Y position for a particular region.
- Date: indicates the date in the format day/month/year.

- Time Activity: indicates the hour of the day of data collection
- Internet Activity: indicates the duration of the Internet activity at a particular grid within time slot of 60 minutes (one hour).

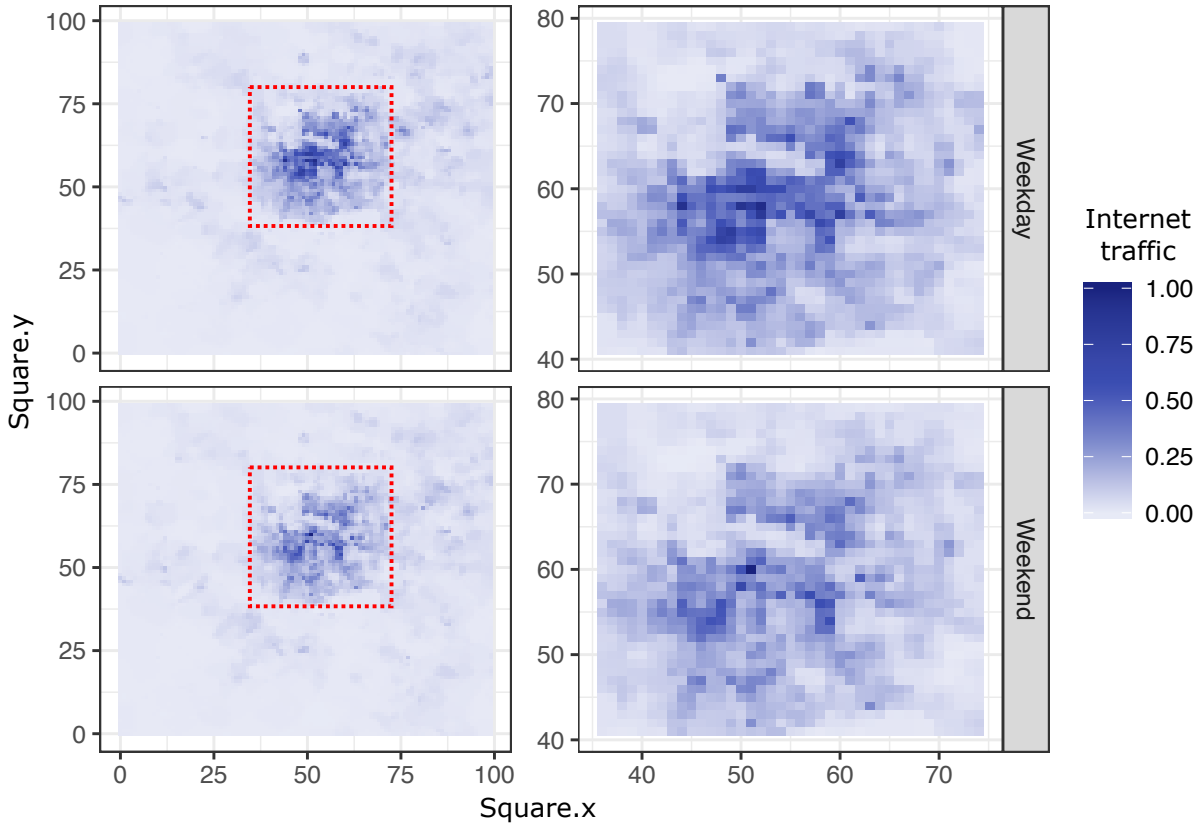

Figure 4.3: Internet traffic activity during two weeks

As the Internet traffic activity in this data set corresponding to rural areas is very low, we decided to focus on urban, suburban, and sub-rural areas around Milan. Exactly 1,521 squares were selected from the grid, corresponding to an area of 83,997,225 m^2 around central Milan. For each day, the data entries are aggregated in time slots of one hour. Finally, the CDRs for Internet traffic activity is normalized in order to scale it to values between 0 and 1 for each hour of the day, for all square areas. Figure 4.3 presents the normalized Internet traffic activity heatmap for the entire data set, separated by weekday and weekend days. Note that none of the data comes from holidays, and the selected squares are the ones that belong to the region with highest traffic activity, *i.e.,* around city center Milan.

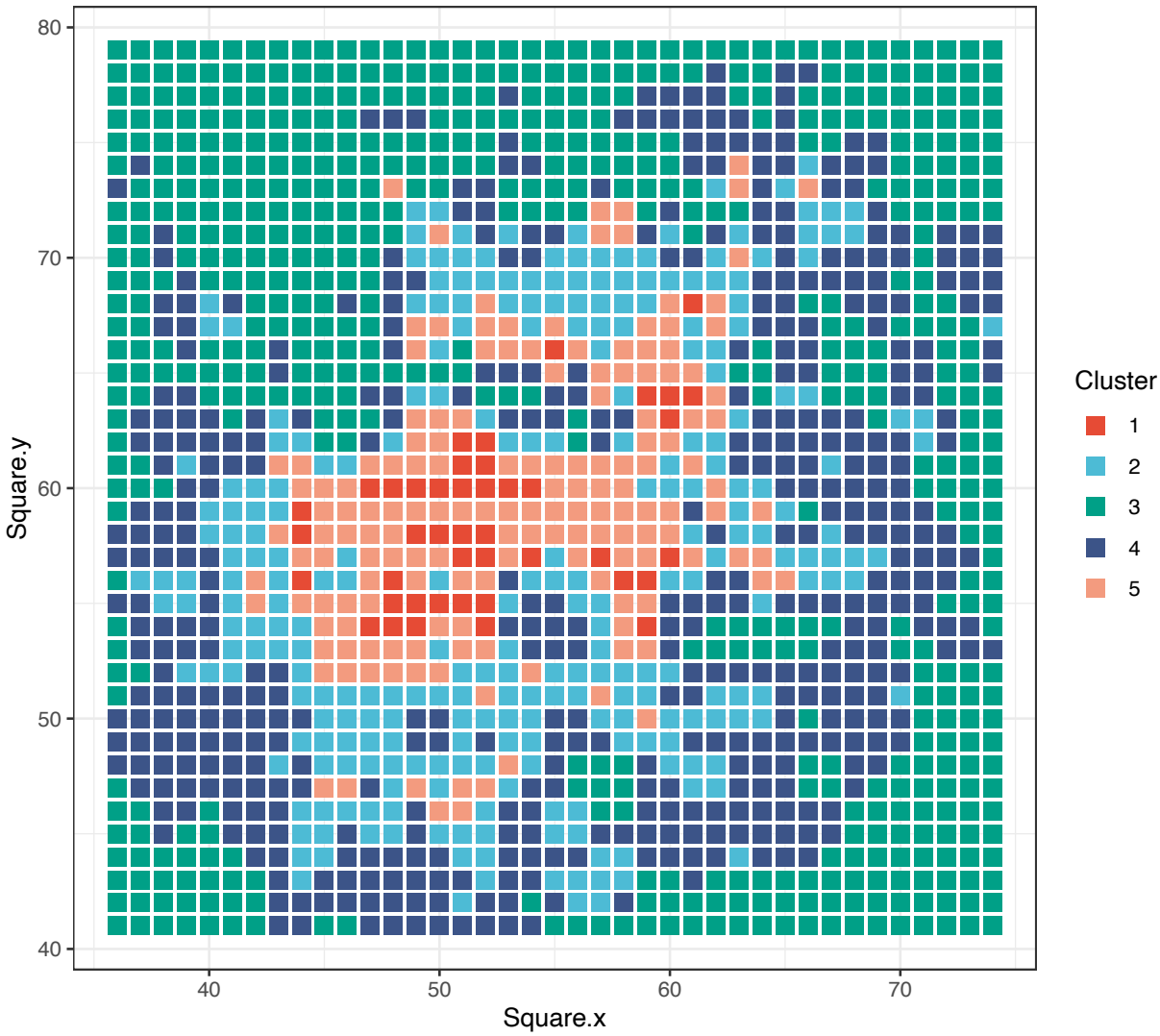

Figure 4.4: Milan city center clusters

After preprocessing and filtering the data, k-means [Macqueen, 1967], an unsupervised learning technique for data clustering, was applied using the euclidean distance. This part of the work was developed using R, a statistical programming language. This technique were used to distinguish the Internet traffic behaviour among regions of Milan in different hours of the day. In this case, the features employed were the sum of Internet traffic for each square area and their x and y positions. As a result, the area was divided into five clusters with different demand behaviour during a day, illustrated in Figure 4.4. The number of clusters was defined empirically based on the Elbow test [Thorndike, 1953], in which the results is presented in Figure 4.5. The Elbow test is a method of interpretation and validation of consistency within cluster analysis designed to help to find the

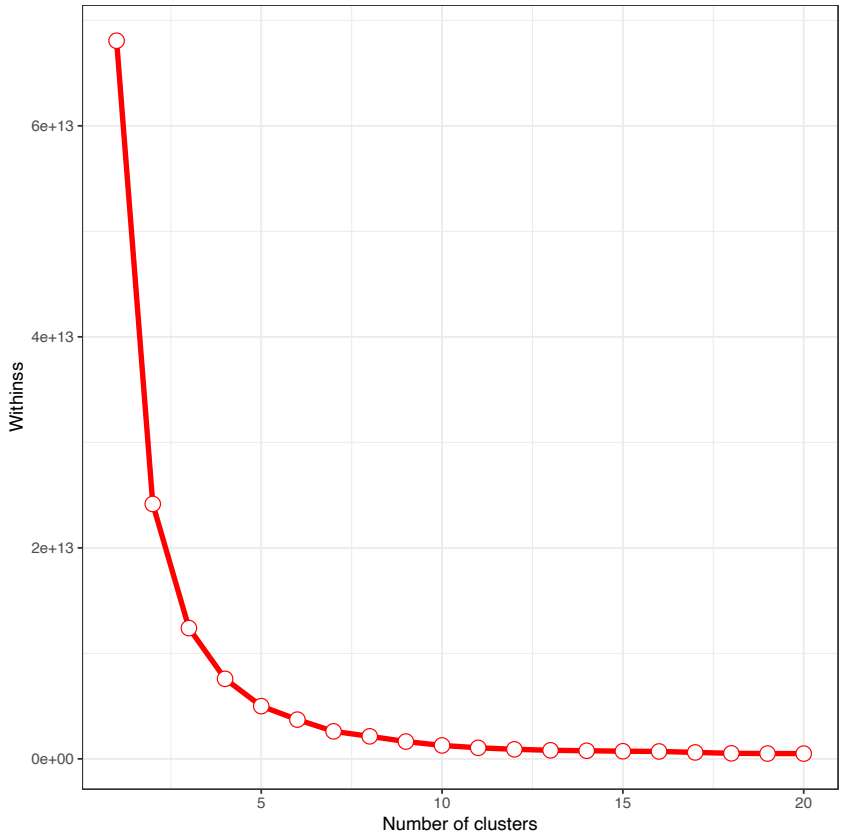

Figure 4.5: Elbow test results.

appropriate number of clusters in a dataset. From Figure 4.5 it is possible to notice that the line begins to flatten significantly right after the number of clusters is equal to five. Therefore, it is clear that the optimized number of clusters for clustering the data set in the k-means clustering algorithm is five.

The demand behavior of each cluster is depicted in the bar plots in Figure 4.6, which presents the normalized value of Internet traffic for each cluster on weekdays and weekends.

4.3.2 Scenario

To evaluate the proposed solution, a simulation of an F-RAN system according to RAN scenarios described by Third Generation Partnership Project (3GPP) [3GPP, 2010, Annex A] was performed. Each macrosite is composed of one macrocell with one MDC, and one cluster of smallcells with an MDC positioned in the centroid of this cluster of antennas, such as presented in 4.7. All parameters values contained in the 3GPP Technical

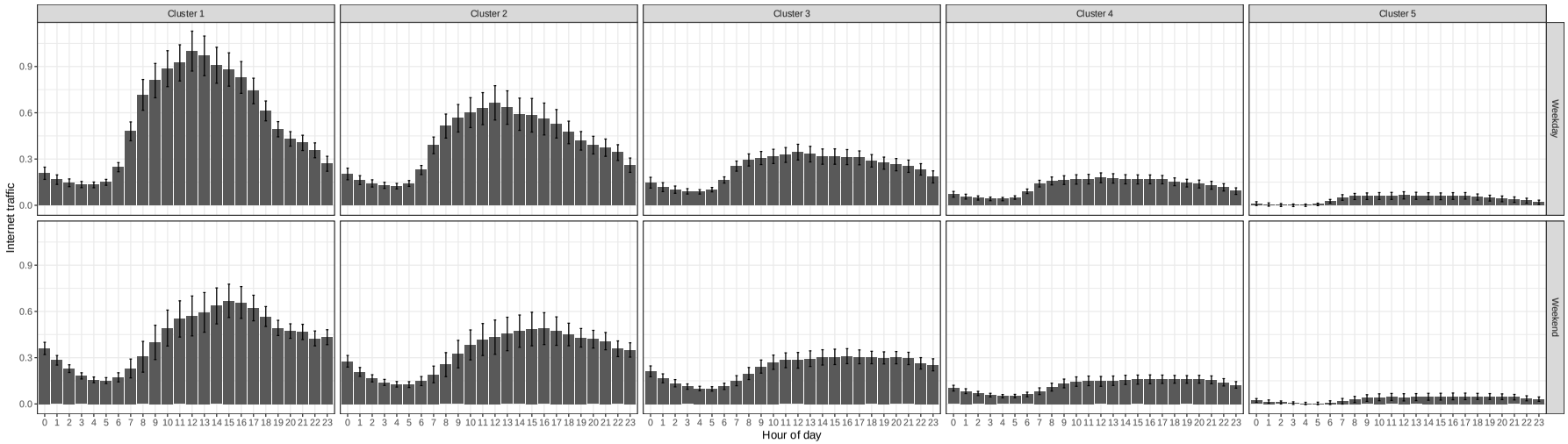

Figure 4.6: Internet traffic activity during a day.

Report [3GPP, 2010, Annex A] were reused and all the MDC parameters are summarized in Table 4.2.

Table 4.2: MDCs' specifications

MDC	**Processor**	i	g_{is}	p_{is}	f_{is}	k_{is}
Macrocell	D-1653	1	11	8	2.8 GHz	0.340
	D-1667	2	11	12	2.7 GHz	0.510
Smallcell	D-1513N	3	7	4	2.2 GHz	0.170
	D-1633N	4	7	6	2.5 GHz	0.255

The definition of the number of GPPs within MDCs is based on a DX2000 2U rack[Hiro Microdatacenters,], with Intel Xeon processors with maximum efficiency of $e_{is} = 16$ operations per cycle for each MDC $s \in \mathcal{S}$ and for all vBBU class $i \in \mathcal{I}$. The cost of allocation per vBBU k_{is} in USD follows the proportion of the price of an Amazon C5[Amazon Web Services, 2019] instance. RRHs were configured with a maximum transmission rate (uplink) of 50 Mbit/s [Watanabe and Machida, 2012]. The workload for each RRH at each time slot is calculated by multiplying the average number of bits transmitted in one second of operation to a random number between the interval delimited by the error rate of the normalized traffic activity value gathered from the Milano CDR data set. The delay budget is set to $\varphi = 0.00270$ s [China Mobile Research Institute, 2011], $w = 200$ operations per bit for the channel decoding process [Holma and Toskala, 2009], the distance for signal regeneration is set to $h = 50,000$ m, and the delay per hop is $\lambda = 0.00005$ s [Marotta et al., 2018].

The simulation of the Milan city center was divided into four smaller simulations with different characteristics: (*i*) downtown; (*ii*) urban; (*iii*) suburban; and (*iv*) sub-rural. Figure 4.8 summarises the simulation configuration for these four areas:

- Downtown: the simulation for downtown is composed of three macrosites following the demand behaviour from cluster 1 and four from cluster 2. The distance between macrocells is set to $d_{macro} = 1000$ m. In this case, the area of one hexagon is equivalent to the area of 14 squares.
- Urban: the simulation for the urban area is composed of four macrosites following the demand behaviour from cluster 2 and three from cluster 3. The distance between macrocells is set to $d_{macro} = 2000$ m and the area of one hexagon is equivalent to 62 squares.

- Suburban: the simulation for the suburban area is composed of three macrosites following the demand behaviour from cluster 2, two from cluster 3, and four from cluster 4. The distance between macrocells is set to $d_{macro} = 3000$ m and the area of one hexagon is equivalent to 141 squares.
- Sub-rural: the simulation for the sub-rural area is composed of two macrosites following the demand behaviour from cluster 4 and five from cluster 5. The distance between macrocells is set as $d_{macro} = 9000$ m and the area of one hexagon is equivalent to 1,271 squares.

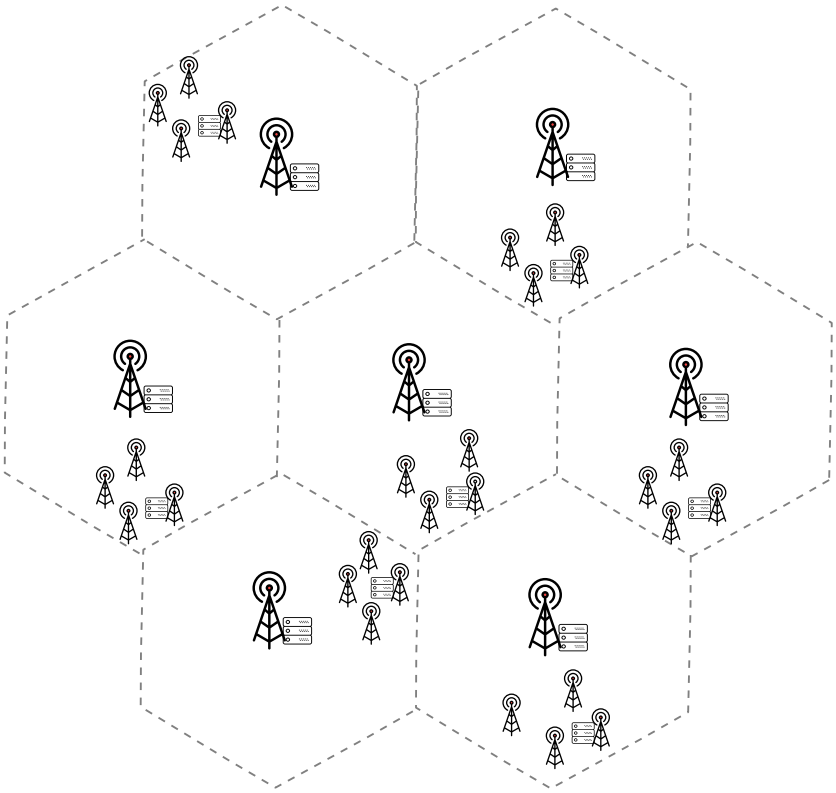

Figure 4.7: Scenario representation.

The source code for all the experimentation is available on GitHub[1]. Next, the results of the simulations are analyzed.

[1] `https://github.com/jonathanalmd/f-ran-optimal-assignment`, last accessed on 10-18-2020.

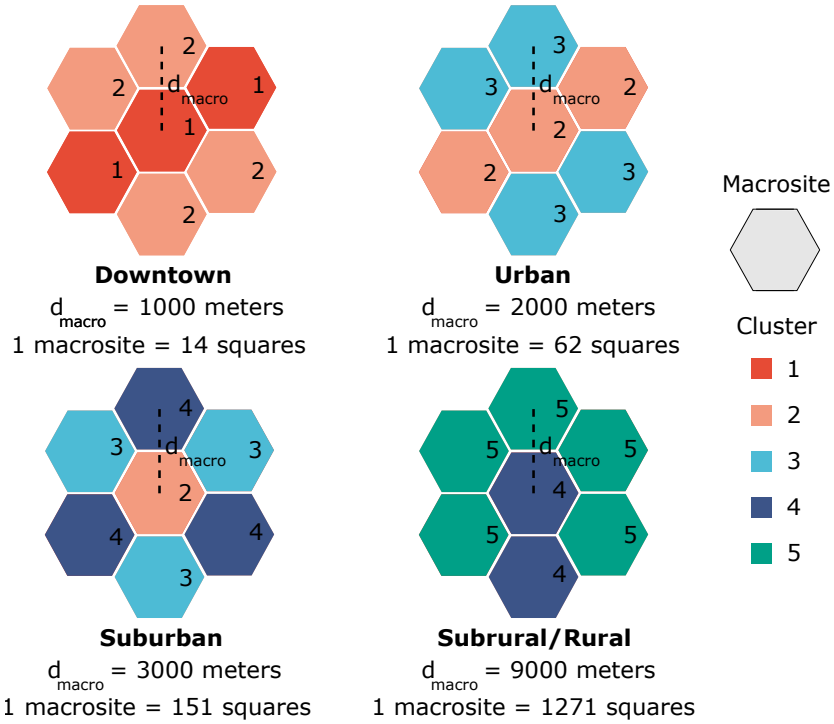

Figure 4.8: Milan simulation scenarios.

4.4 Results and Analysis

To characterize the relation between processing power consumption and distance between MDC and RRH, first, the Equation 4.2 is analysed considering different classes of vBBUs. Figure 4.9 illustrates the influence of distance on the minimal number of vBBUs per class that are allocated, considering an RRH with fixed upstream data rate of 50 Mbit/s, characterizing an important trade-off in F-RANs.

To evaluate the presented solution, the analysis is focused in the potential gains considering cost minimisation with optimised vBBU allocation. The optimal assignment of RRHs to MDCs is achieved by applying BILP. From the optimization result, the optimal allocation of vBBUs in terms of cost minimisation with the decision assignment binary variable $a_{ism}(t)$ is obtained. Furthermore, this result is combined with the possibility of increasing operator's income by leasing idle resources. In this case, the maximum income for each time slot is obtained by multiplying $U_{cores}(t)$ (Equation 4.4) by a constant, which represents the cost per core defined by the operator. To present the results, this constant is set equal to 0.02125 USD, half of the price per core set by Amazon for a *c5.large* dedicated instance.

Figure 4.10 summarises the results obtained, separated by urban region categories,

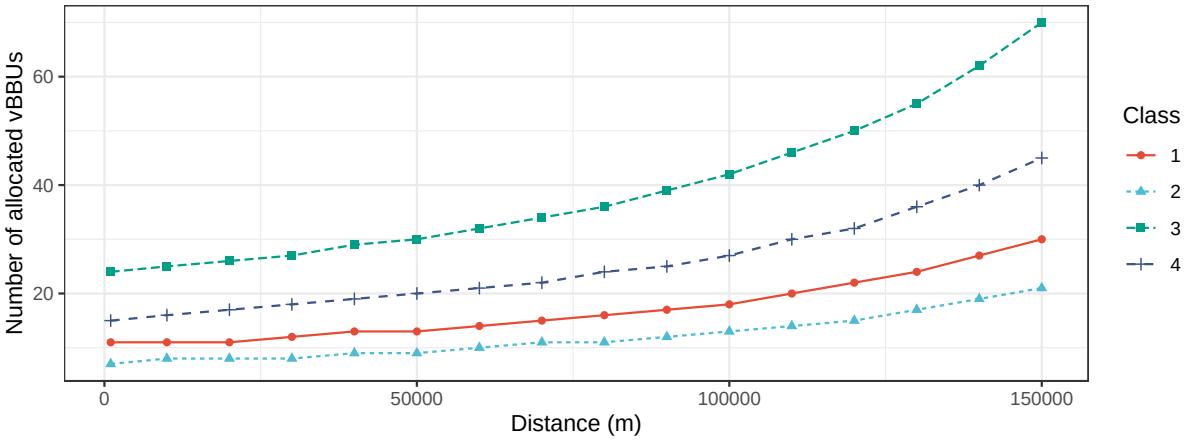

Figure 4.9: Trade-off between distance and processing power.

weekday versus weekend, and the day is divided into night (12 AM to 6 AM), morning (6 AM to 12 PM), afternoon (12 PM to 6 PM), and evening (6 PM to 0 AM). This figure presents the proportion of maximum income per cost of allocation, indicating the potential of gains for network operators. The potential for gain is more relevant when the proportion is more than one, which means the operator's maximum income is higher than the budget used to serve its demand, *i.e.,* the operator can gain this proportion of income for each cash unit used to serve its demand.

From the results, the maximum potential for gain from leasing idle processing power occurs during the night, and is slightly higher on weekdays. From the previous results for demand behaviour in different regions presented in Figure 4.6, it is possible to see that those correspond to the periods of lowest Internet traffic activity. For the remaining periods of the day the most relevant results are from weekday mornings and afternoons for the urban area and from weekday mornings to evenings for downtown. These regions and periods are the ones with highest peaks of activity and potentially the periods that ASPs are most interested in provisioning their services at the edge. It is possible to notice that there is an high opportunity in leasing edge processing resources, *i.e.,* vBBUs of MDCs from smallcells. Note that this opportunity is considerably more evident in the downtown area. This behaviour occurs because the optimal solution, in high demand scenarios, decides to centralize the processing to more powerful GPPs, which are located within MDCs from macrocells. On weekends, this processing centralization is only evident during the period of afternoon to evening for the downtown area and only during the evening for the urban area. For weekend mornings downtown, the opportunities of leasing

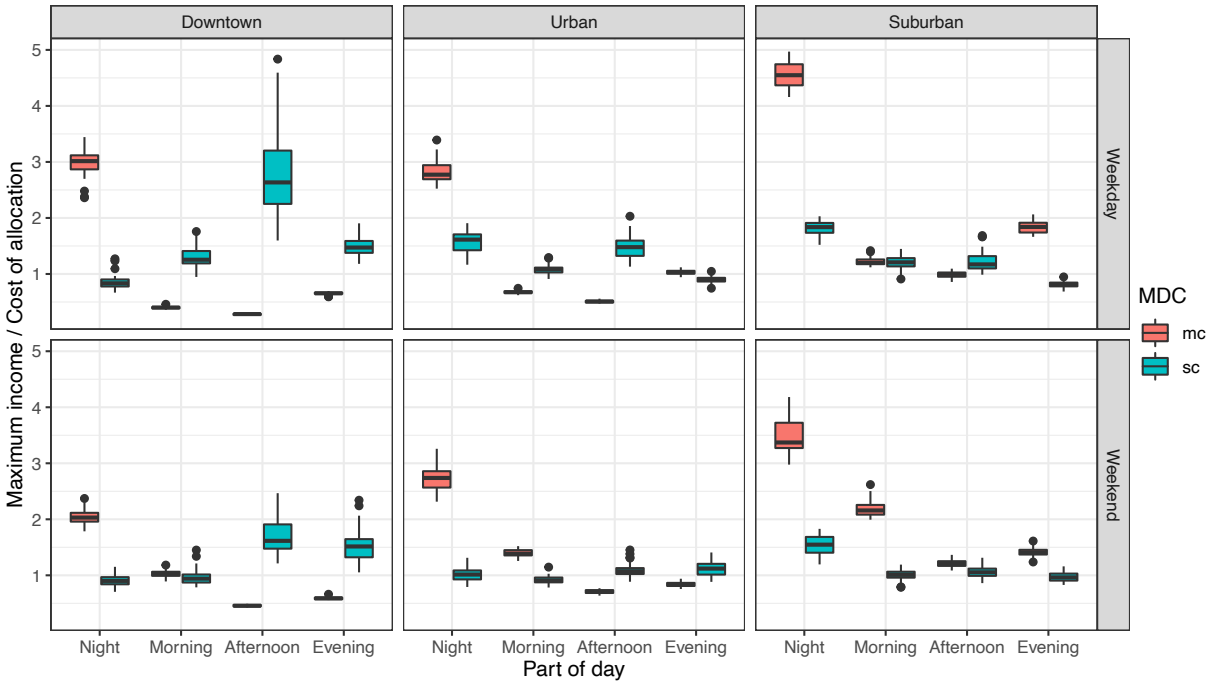

Figure 4.10: Ratio of maximum income to cost of allocation for each time of the day, on weekdays and weekends for MDCs from macrocells and smallcells.

processing power of MDCs from both macrocells and smallcells are statistically the same. Nonetheless, considering macrocells' MDCs, there is a higher potential of gains when compared to weekdays.

Notice that even considering the allocation cost for an operator to serve its own demand is the cost of an Amazon EC2 C5 instance, there is still a huge opportunity in leasing idle resources for half this price, in most of situations. Considering a sample size of 50 and the application of a t-test with 95% confidence interval, the proportion of maximum income per cost of allocation is statistically less than one only for macrocells' MDCs only in downtown weekdays during morning to evening, urban weekdays during morning and afternoon, and downtown/urban weekends during afternoon and evening. For smallcells' MDCs, this occurs only during downtown weekday nights. Nonetheless, as aforementioned, the most significant opportunity to be exploited relies on leasing edge processing to ASPs. Considering the same scenarios, the confidence interval for the proportion is considerably greater than one, as depicted in Figure 4.10. In particular, this value can be greater than three for downtown weekdays, fluctuating between 2.7 and 3.2 during afternoons. These results show that there is a better opportunity for operators to increase their income through edge processing leasing in these mentioned scenarios,

especially in downtown.

From the perspective of the entire day (weekdays and weekends), considering a 95% confidence interval, the proportion are statistically greater than one for MDCs from macrocells in downtown, urban, and suburban areas. In particular, in suburban areas this proportion fluctuates around 2.1. Considering MDCs from smallcells, in which the confidence interval in downtown, urban, and suburban areas are around 1.1 and 1.5. Note that the opportunity for leasing processing from smallcells' MDCs is better in downtown, which is the region with higher values, fluctuating between 1.4 and 1.5. Whereas, suburban areas have more potential of gains when considering MDCs from macrocells, in which leasing more powerful processing power can be characterized as a key opportunity to this region. A balance between these two, urban regions have potential of gains both for macro and smallcells processing. This interval is impacted by the high potential of gains during night periods. However, ASPs may not be so interested in providing their services during night. In this context, operators should take into account the period of the day when deciding the price of allocation per core. Moreover, considering that ASPs may have more interest in provisioning services in downtown/urban areas, operators should consider the region when setting the price of allocation. Therefore, operators must evaluate the ASPs' interests in order to set this price according to the different situations.

Note that the results from the optimization for subrural/rural areas was not included to not compromise the visualization of the results for urban areas. This kind of region has a demand behaviour that diverges a lot from urban areas. It is not fair to consider an MDC for subrural/rural region with the same configuration of an MDC used in urban areas, it is expected that the processing power will be underused during the day. In this case, a neutral hosting may be ideal for most network operators, with a shared infrastructure where the Internet activities from different operators are summed. Nonetheless, even considering a homogeneous configuration for MDCs, subrural and rural regions have the potential to be exploited regarding processing power leasing. Even though it is a subrural/rural region, it is still closer to the city center when compared to Amazon EC2 instances that, for a large proportion of cities, is located hundreds or even thousands of kilometers away. In this context, for example, an operator may exploit this opportunity by employing most of their MDCs' processing power from subrural/rural regions to this purpose.

Comparing the potential gains between the city of Milan and the province of Trento,

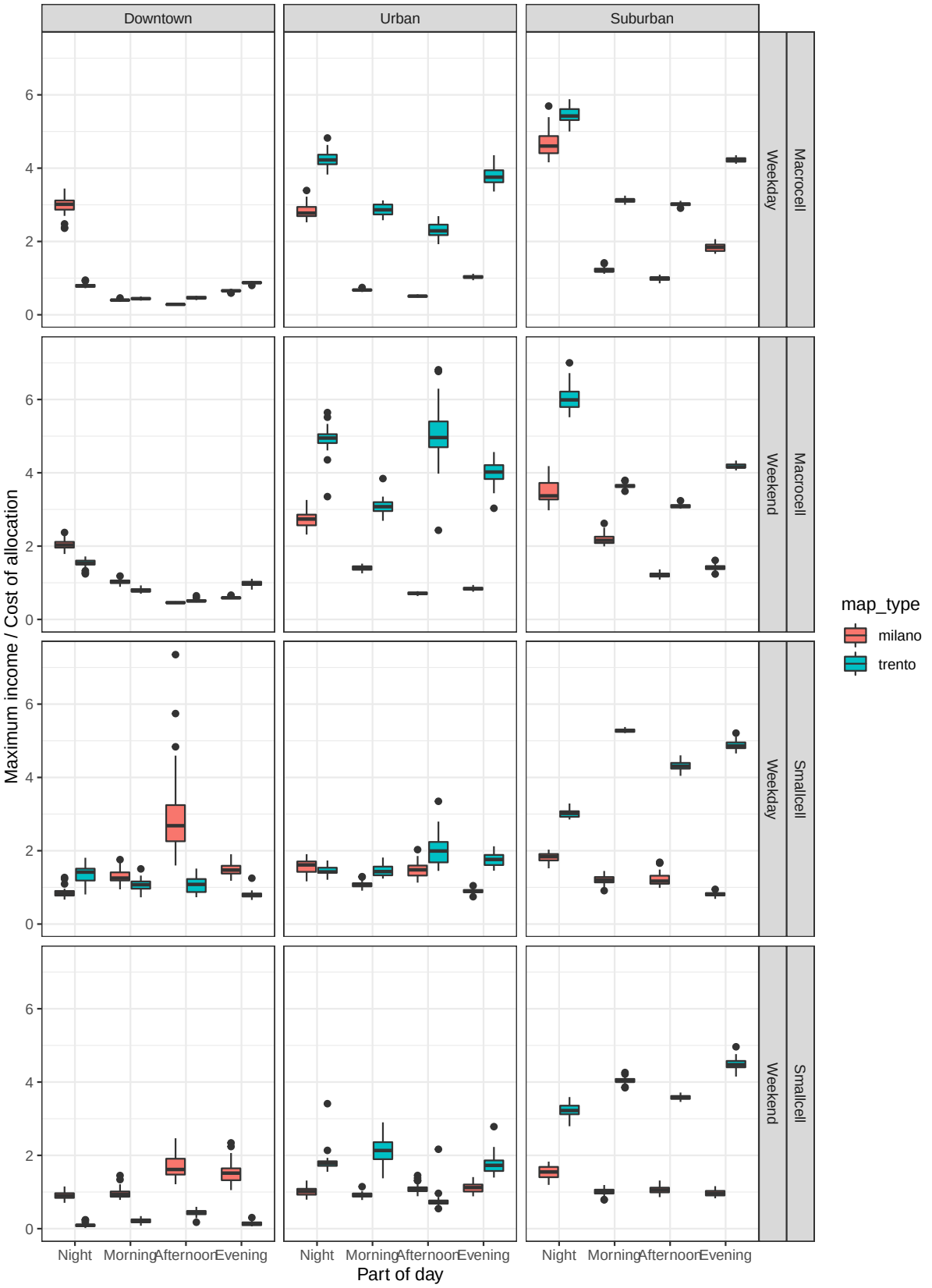

Figure 4.11: Ratio of maximum income to cost of allocation for each time of the day, on weekdays and weekends for MDCs from macrocells and smallcells considering the city of Milan and the province of Trento.

the major differences occur in urban and suburban scenarios, as depicted in Figure 4.11. Given that the traffic behavior in the Trento downtown area is similar to the behavior in Milan downtown area, it was expected that the opportunity for income gains would be similar to what was observed in Milan. For urban e suburban areas, the internet traffic in Trento is much less intense when compared with Milan. Consequently, considering the same configuration of MDCs, the opportunity of income gains in Trento is naturally higher. Nonetheless, it should be taken into account the fact that ASPs probably are more interested in providing their services in larger cities, where the demand for processing is higher. Therefore, in theory, even though the proportion of maximum income and cost of allocation can be higher for the province of Trento, it is necessary to consider that the demand from ASPs will probably be lower than for the city of Milan. To evaluate such aspect, it is necessary to perform a further investigation in the characteristics of service usage in both cities, which is beyond the scope of this thesis proposal.

4.5 Importance and Benefits of Decomposing and Integrating Time Granularities

It is important to decompose time granularities since it can be used to bring clear benefits to F-RANs' performance. The benefits are related to the usage of fine-grained data to improve the solution of thicker granularities. Based on the simulated scenarios presented in the previous sections, it is possible to evidence the importance of decomposing the timescales by applying the optimal solution and analyzing the number of migrations per MDC per day, such as presented in Figure 4.12.

The number of migrations increases as the time interval for decision-making decreases. Considering a sample size of 35 and the application of a t-test with a 95% confidence interval, the average number of migrations per MDC per day for time intervals of 10, 30, and 60 minutes are 168.4 $\pm$ 4.8, 65.7 $\pm$ 1.7, and 36.3 $\pm$ 1.0, respectively. The number of migrations for a time interval of 10 minutes and 30 minutes is approximately 4.6 and 1.8 times higher than for a time interval of one hour. Analyzing the spatial aspect, the number of migrations is higher in urban areas, in which Internet traffic is intense and fluctuates more.

It is crucial to decide a proper time interval for decision-making since each migration has an associated cost depending on the migration time and transmission cost

[Liu et al., 2018]. Moreover, a larger range can enable the application of more sophisticated ML techniques as well as more frequent updates in the model. Therefore, this use case shows that a larger time granularity is, in fact, better suited for vBBU allocation in F-RANs.

Considering the integration of time granularities, there is an opportunity in incorporating data from a smaller timescale for decision-making in a larger timescale. For instance, training a model for traffic prediction for each hour using data in the time granularity of minutes can improve the performance of the predictor (*i.e.,* reduce the error), as depicted in Figure 4.13. This figure presents the Root Mean Square Error (RMSE) for three different models: M1, M2, and Default. The RMSE is defined as:

$$RMSE = \sqrt{(\frac{1}{n})\sum_{i=1}^{n}(y_i - x_i)^2} \tag{4.6}$$

The RMSE is small, it can indicate that the noise is small and, consequently, the model is good at predicting the observed data. Conversely, the trained model is missing important features if the RMSE is large, *i.e.,* the noise is large. M1 is the model trained using data in the time granularity of hours, *i.e.,* using the Internet traffic for each hour of the day. M2 is the model trained using data in the time granularity of minutes, *i.e.,* using the Internet traffic for each 10 minutes of the day. M1 and M2 were trained using data from two weeks (14 days) and tested using five weeks (35 days). Lastly, the default model only computes the average Internet traffic during a day and allocates this value during the entire day, which is used to evaluate if it is worth using a traffic predictor rather than a simple allocation rule.

Analyzing the boxplot presented in Figure 4.13 it is possible to notice that the RMSE for M2 is lower when compared with M1. Considering a 95% confidence interval and a sample of 35 observations, the RMSE for M1, M2, and Default are 0.0575±0.0057, 0.0384±0.0065, and 0.1214±0.01333, respectively. These results indicate a performance improvement (RMSE reduced about to 50%) through time granularity integration by only incorporating fine grained data to a decision-making solution from a thicker granularity, *i.e.,* using data at the timescale of minutes to predict in hours.

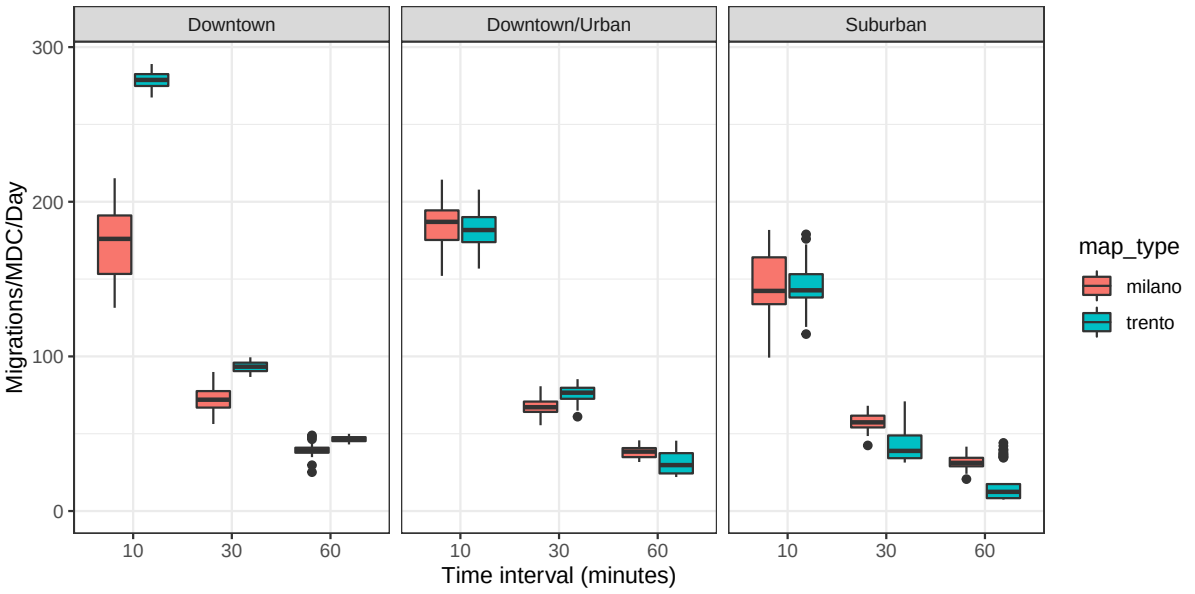

Figure 4.12: Number of migrations per MDC per day in Milan and Trento (Italy), considering the trade-off between processing power and distance between MDC and RRH evaluated in different time granularities under three time intervals: (*i*) 10 minutes; (*ii*) 30 minutes; and (*iii*) 60 minutes.

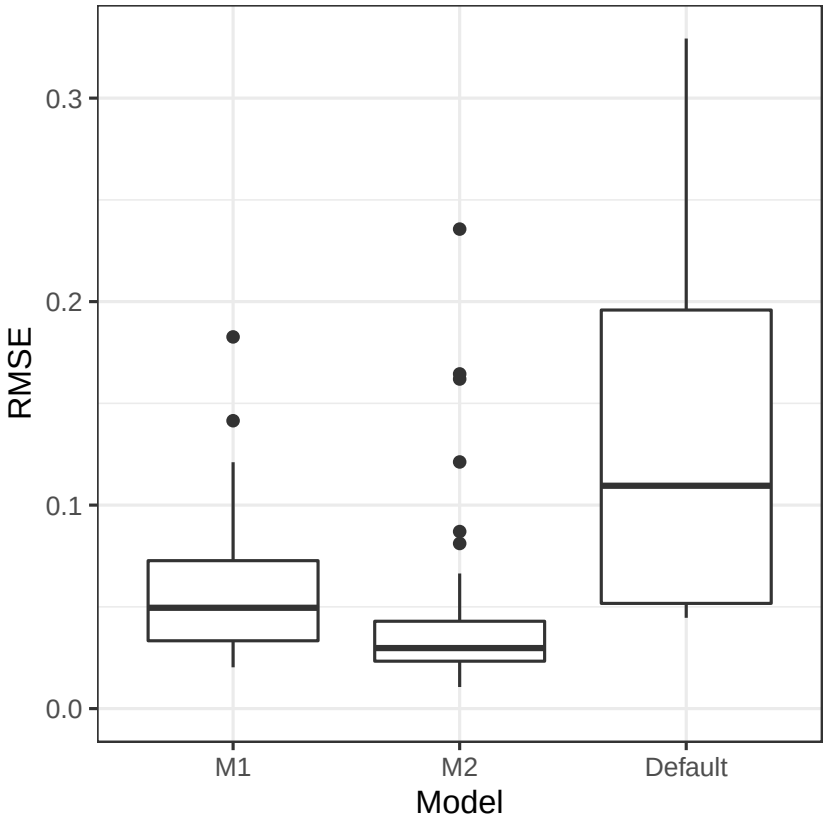

Figure 4.13: RMSE for each trained model.

Chapter 5

Concluding Remarks and Future Work

5.1 Conclusions

In the first part of this thesis proposal (Chapters 2 and 3), the benefits and challenges of implementing an AI-driven F-RAN were discussed. To drive the discussion, AI techniques that are applied to an F-RAN on three timescales are considered: hours, minutes/seconds, and milliseconds. At the time granularity of hours, AI techniques such as neural networks, k-means, and genetic algorithms can bring benefits to F-RANs in terms of low energy consumption, cost reduction, and processing power minimization in the cloud. Meanwhile, at a time granularity of minutes/seconds, reinforcement learning-based techniques and adaptive online learning algorithms enable F-RANs to reduce energy consumption of RRHs, enhance caching hit rate, and promote better decisions for local service deployment. At the time granularity of milliseconds, the application of AI classifiers and Q-learning can bring spectral efficiency gains, throughput improvements, and CQI signaling overhead reduction. Finally, to enable integration between AI solutions from the same granularity and from different granularities, a multiagent architecture for F-RANs is proposed.

In the second part of this thesis (Chapter 4), opportunities for network operators to reduce their expenditures through optimal allocation of vBBUs in F-RANs were investigated, tackling a particular goal from the time granularity of hours. Besides, the possibility of generating additional revenue by leasing idle resources to ASPs is discussed, highlighting which situations this opportunity can be better exploited. In particular, the

challenge of improving vBBU allocation in terms of optimal assignment of RRHs to MDCs for cost minimisation is addressed, considering the trade-off between processing power and distance between MDC and RRH. In this context, optimization model to perform vBBU allocation by optimally deciding the RRH to MDC assignments considering such trade-off is proposed. The solution was evaluated by applying a real CDR data set, simulating regions from Milan area. To identify the demand behaviour from different regions considering spatial and temporal characteristics, k-means clustering was applied. The results of this work highlighted the opportunity for network operators to exploit their infrastructure better and increase their gains.

Given characterized challenges and the proposed solutions, further investigations and experiments have been conducted aiming to verify the following the fundamental question and hypothesis by answering the research questions presented in Chapter 1.

Fundamental Question: How to design an intelligent decision-making system able to address the challenges of F-RANs considering the distance between MDC and RRH as a key factor?

Hypothesis: AI techniques must be applied in order to improve decision-making considering the distance between MDC and RRH under different time constraints in F-RANs.

Even though there is still more experiments and investigation to perform, the presented results underlie the proposed hypothesis. During this work, we were able to to answer the research questions (RQ 1 to 3) associated with the hypothesis. The answers to each question are detailed as follows.

RQ 1 - What are the main decisions to be taken in an AI-driven F-RAN considering the different time granularities?

As presented in Chapter 2, throughout a systematic investigation of the application of AI techniques in F-RANs and its benefits, it was possible to classify around three time granularities: hours, minutes/seconds, and milliseconds. Concomitantly, by investigating the AI-driven solutions for F-RANs, it was possible to identify key decision-making possibilities of each time granularity.

RQ 2 - How to integrate decision-making possibilities from different time granularities in an AI-driven F-RAN?

In Chapter 3, a solution to enable integration between AI solutions from the same granularity and from different granularities, a multiagent architecture for F-RANs was proposed.

RQ 3 - How to formalize decision-making in F-RAN considering vBBUs allocation, the distance between MDC and RRH, and time constraints?

The problem of optimal allocation of vBBUs considering the distance between MDC and RRH can be formulated as an optimization problem, as presented in Chapter 4. In this sense, this proposal presented an optimization model to perform vBBU allocation by deciding the RRH to MDC assignments considering trade-off between processing power and distance. The objective function is subjected to constraints regarding horizontal and vertical allocation. Furthermore, the solution was evaluated by applying a real CDR data set, simulating regions from Milan area.

Based on the investigations conducted, it is possible to identify some open challenges. For instance, it seems to be possible to incorporate ASP transactions to the optimization model in order to minimize the operators' expenditures while directly considering the potential of income gains by leasing idle resources. Moreover, there is still the need to perform further investigation in regarding ML techniques that are suitable to the problem of vBBU allocation. Considering the literature review, it is possible to address some adequate AI techniques to be applied. Nonetheless, for this specific problem (vBBU allocation considering the distance between MDC and RRH) it is necessary to perform extensive comparative experiments in order to obtain the best feasible solution, *i.e.,* closer to the optimal solution.

5.2 Future Work

As future work, we aim to incorporate ASP transactions to the optimization model to minimise the operators' expenditures while directly considering the potential of income gains by leasing idle resources. Moreover, we will investigate the use to of ML techniques to make vBBU allocation decisions and compare them with the optimal solution. Then, we aim to integrate different agents with distinct capabilities from the same time granularity, which is the first step to implement the proposed multi-agent architecture.

Further, we aim to deal with the challenge of integrating decision-making across different granularities. The main challenge to be tackled is the integration between different

time granularities while taking into account the time constraints and trade-offs among decision-making possibilities. In this context, our first step is to implement the agents from the granularity of hours and integrate them into one multi-agent system, followed by the agents from fine-grained granularities. Once we implement the multi-agent systems from each granularity, we aim to integrate them into one multi-agent architecture for AI-driven F-RANs. In this case, the goal is to achieve integration among different time granularities and evaluate the gains.

References

[3GPP, 2010] 3GPP (2010). 3rd Generation Partnership Project (3GPP) TR 36.814 - Further advancements for E-UTRA physical layer aspects. *Technical Report*, 9(3):1–104. 42, 44

[3GPP, 2020] 3GPP (2020). 3rd Generation Partnership Project (3GPP) TS 29.520 version 16.4.0 Release 16 - 5g; 5g system; network data analytics services; stage 3. Technical Report 4. 2

[Alqerm and Shihada, 2018] Alqerm, I. and Shihada, B. (2018). Sophisticated Online Learning Scheme for Green Resource Allocation in 5G Heterogeneous Cloud Radio Access Networks. *IEEE Transactions on Mobile Computing*, 17(10):2423–2437. 2, 16, 17, 22, 23

[Amazon Web Services, 2019] Amazon Web Services (2019). Amazon EC2 C5 instances specifications and pricing. `https://aws.amazon.com/ec2/instance-types/c5/`, Last accessed on 2019-07-31. 27, 44

[Aqeeli et al., 2018] Aqeeli, E., Moubayed, A., and Shami, A. (2018). Power-aware optimized RRH to BBU allocation in C-RAN. *IEEE Transactions on Wireless Communications*, 17(2):1311–1322. 4

[Aryal and Altmann, 2018] Aryal, R. G. and Altmann, J. (2018). Dynamic application deployment in federations of clouds and edge resources using a multiobjective optimization AI algorithm. *International Conference on Fog and Mobile Edge Computing*, pages 147–154. 5, 10, 12, 21

[Balevi and Gitlin, 2018] Balevi, E. and Gitlin, R. D. (2018). A clustering algorithm that maximizes throughput in 5g heterogeneous f-ran networks. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–6. 23

[Bhaumik et al., 2012] Bhaumik, S., Chandrabose, S. P., Jataprolu, M. K., Kumar, G., Muralidhar, A., Polakos, P., Srinivasan, V., and Woo, T. (2012). CloudIQ: a framework for processing base stations in a data center. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, pages 125–136. 33

[Chamberlain and Teucher, 2020] Chamberlain, S. and Teucher, A. (2020). *geojsonio: Convert Data from and to 'GeoJSON' or 'TopoJSON'*. R package version 0.9.2. 38

[Chen et al., 2018a] Chen, L., Xu, J., Ren, S., and Zhou, P. (2018a). Spatio Temporal Edge Service Placement: A Bandit Learning Approach. *IEEE Transactions on Wireless Communications*, 17:8388–8401. 2, 6, 13, 14, 22, 23

[Chen et al., 2017] Chen, M., Saad, W., Yin, C., and Debbah, M. (2017). Echo state networks for proactive caching in cloud-based radio access networks with mobile users. *IEEE Transactions on Wireless Communications*, 16(6):3520–3535. 23

[Chen et al., 2018b] Chen, X., Zhang, H., Wu, C., Mao, S., Ji, Y., and Bennis, M. (2018b). Performance optimization in mobile-edge computing via deep reinforcement learning. *IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pages 1–6. 23

[Chien et al., 2019] Chien, W., Lai, C., and Chao, H. (2019). Dynamic Resource Prediction and Allocation in C-RAN With Edge Artificial Intelligence. *IEEE Transactions on Industrial Informatics*, 15(7):4306–4314. 2, 4, 5, 6, 10, 11, 21, 24

[China Mobile Research Institute, 2011] China Mobile Research Institute (2011). C-RAN the road towards green RAN. Version 2.5. 35, 44

[Dongqing Liu et al., 2017] Dongqing Liu, Khoukhi, L., and Hafid, A. (2017). Decentralized data offloading for mobile cloud computing based on game theory. In *2017 Second International Conference on Fog and Mobile Edge Computing (FMEC)*, pages 20–24. 5

[Du and Nakao, 2018] Du, P. and Nakao, A. (2018). Deep learning-based application specific ran slicing for mobile networks. In *2018 IEEE 7th International Conference on Cloud Networking (CloudNet)*, pages 1–3. 23

[Gao et al., 2019] Gao, Z., Zhang, J., Yan, S., Xiao, Y., Simeonidou, D., and Ji, Y. (2019). Deep reinforcement learning for bbu placement and routing in c-ran. In *2019 Optical Fiber Communications Conference and Exhibition (OFC)*, pages 1–3. 24

[Girgis et al., 2019] Girgis, A. M., Ercetin, O., Nafie, M., and ElBatt, T. (2019). Fundamental limits of memory-latency tradeoff in fog radio access networks under arbitrary demands. *IEEE Transactions on Wireless Communications*, 18(8):3871–3886. 24

[Habibi et al., 2019] Habibi, M. A., Nasimi, M., Han, B., and Schotten, H. D. (2019). A comprehensive survey of RAN architectures toward 5G mobile communication system. *IEEE Access*, 7:70371–70421. 2

[Hiro Microdatacenters,] Hiro Microdatacenters. Micro Data Center specifications. `http://hiro-microdatacenters.nl/`, Last accessed on 2019-10-13. 44

[Holma and Toskala, 2009] Holma, H. and Toskala, A. (2009). *LTE for UMTS-OFDMA and SC-FDMA based Radio Access*. John Wiley & Sons. 44

[Huang et al., 2017] Huang, X., Sun, Y., Zhang, C., and Peng, M. (2017). A hierarchical approach for terminal awareness in fog radio access networks. In *2017 IEEE/CIC International Conference on Communications in China (ICCC)*, pages 1–6. 23

[Hussain et al., 2020] Hussain, B. et al. (2020). Artificial Intelligence-powered Mobile Edge Computing-based Anomaly Detection in Cellular Networks. *IEEE Transactions on Industrial Informatics*, 16(8):1–1.

[Imtiaz et al., 2018] Imtiaz, S., Ghauch, H., Koudouridis, G. P., and Gross, J. (2018). Random forests resource allocation for 5G systems: Performance and robustness study. *2018 IEEE Wireless Communications and Networking Conference Workshops*, pages 326–331. 2, 16, 18, 22, 23

[Jaffry et al., 2020] Jaffry, S., Shah, S. T., and Hasan, S. F. (2020). Data-driven Semi-supervised Anomaly Detection using Real-World Call Data Record. In *IEEE Wireless Communications Networks Conference*, pages 3–8.

[Jiang et al., 2019] Jiang, F., Yuan, Z., Sun, C., and Wang, J. (2019). Deep q-learning-based content caching with update strategy for fog radio access networks. *IEEE Access*, 7:97505–97514. 24

[Jiang et al., 2019] Jiang, Y., Ma, M., Bennis, M., Zheng, F., and You, X. (2019). User Preference Learning Based Edge Caching for Fog Radio Access Network. *IEEE Transactions on Communications*, 67:1268–1283. 6, 13, 15, 22, 24

[Larsen et al., 2019] Larsen, L. M. P., Checko, A., and Christiansen, H. L. (2019). A survey of the functional splits proposed for 5G mobile crosshaul networks. *IEEE Communications Surveys Tutorials*, 21(1):146–172. 16

[Liu et al., 2018] Liu, J., Yang, Q., Simon, G., and Cui, W. (2018). Migration-based dynamic and practical virtual streaming agent placement for mobile adaptive live streaming. *IEEE Transactions on Network and Service Management*, 15(2):503–515. 10, 52

[Lu et al., 2019] Lu, L., Jiang, Y., Bennis, M., Ding, Z., Zheng, F., and You, X. (2019). Distributed edge caching via reinforcement learning in fog radio access networks. In *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, pages 1–6. 24

[Macqueen, 1967] Macqueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297. 41

[Mao and Yan, 2019] Mao, Z. and Yan, S. (2019). Deep learning based channel estimation in fog radio access networks. *China Communications*, 16(11):16–28. 24

[Marotta et al., 2018] Marotta, M. A., Ahmadi, H., Rochol, J., DaSilva, L. A., and Both, C. B. (2018). Characterizing the relation between processing power and distance between BBU and RRH in a cloud RAN. *IEEE Wireless Communications Letters*, 7(3):472–475. 4, 27, 35, 36, 44

[Moon et al., 2019] Moon, J., Simeone, O., Park, S., and Lee, I. (2019). Online reinforcement learning of x-haul content delivery mode in fog radio access networks. *IEEE Signal Processing Letters*, 26(10):1451–1455. 24

[Musumeci et al., 2016] Musumeci, F., Bellanzon, C., Carapellese, N., Tornatore, M., Pattavina, A., and Gosselin, S. (2016). Optimal BBU placement for 5G C-RAN deployment over WDM aggregation networks. *Journal of Lightwave Technology*, 34(8):1963–1970. 35

[Nakayama et al., 2017] Nakayama, Y., Hisano, D., Kubo, T., Shimizu, T., Nakamura, H., Terada, J., and Otaka, A. (2017). Low-latency routing for fronthaul network: A monte carlo machine learning approach. In *2017 IEEE International Conference on Communications (ICC)*, pages 1–6. 23

[Nassar and Yilmaz, 2019] Nassar, A. and Yilmaz, Y. (2019). Resource allocation in fog ran for heterogeneous iot environments based on reinforcement learning. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pages 1–6. 24

[OpenStreetMap contributors, 2017] OpenStreetMap contributors (2017). Planet dump retrieved from https://planet.osm.org . `https://www.openstreetmap.org`. 38

[Peng et al., 2016] Peng, M., Yan, S., Zhang, K., and Wang, C. (2016). Fog-computing-based radio access networks: issues and challenges. *IEEE Network*, 30(4):46–53. 1, 6

[Peng and Zhang, 2016] Peng, M. and Zhang, K. (2016). Recent Advances in Fog Radio Access Networks: Performance Analysis and Radio Resource Allocation. *IEEE Access*, 4:5003–5009. 2

[Rahman et al., 2018] Rahman, G. M. S., Peng, M., Zhang, K., and Chen, S. (2018). Radio resource allocation for achieving ultra-low latency in fog radio access networks. *IEEE Access*, 6:17442–17454. 23

[Shahriari et al., 2017] Shahriari, B., Moh, M., and Moh, T. (2017). Generic Online Learning for Partial Visible Dynamic Environment with Delayed Feedback: Online Learning for 5G C-RAN Load-Balancer. In *2017 International Conference on High Performance Computing Simulation (HPCS)*, pages 176–185. 23

[Singh et al., 2020] Singh, S. K., Singh, R., and Kumbhani, B. (2020). The evolution of radio access network towards open-ran: Challenges and opportunities. In *2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pages 1–6. 1

[Soliman and Leon-Garcia, 2016] Soliman, H. M. and Leon-Garcia, A. (2016). Fully distributed scheduling in cloud-RAN systems. *2016 IEEE Wireless Communications and Networking Conference*, pages 1–6. 4, 23

[Sun et al., 2019a] Sun, G., Boateng, G. O., Ayepah-Mensah, D., and Liu, G. (2019a). Relational reinforcement learning based autonomous cell activation in cloud-rans. *IEEE Access*, 7:63588–63604. 24

[Sun et al., 2019b] Sun, Y., Peng, M., and Mao, S. (2019b). Deep reinforcement learning-based mode selection and resource management for green fog radio access networks. *IEEE Internet of Things Journal*, 6(2):1960–1971. 24

[Sun et al., 2018] Sun, Y., Peng, M., and Poor, H. V. (2018). A distributed approach to improving spectral efficiency in uplink device-to-device-enabled cloud radio access networks. *IEEE Transactions on Communications*, 66(12):6511–6526. 23

[Tang et al., 2017] Tang, J. et al. (2017). System cost minimization in cloud RAN with limited fronthaul capacity. *IEEE Transactions on Wireless Communications*, 16(5):3371–3384. 4

[Telecom Milano,] Telecom Milano. Milano Grid Telecommunications Dataset. `https://dandelion.eu/datagems/SpazioDati/telecom-sms-call-internet-mi/description/`, Last accessed on 2019-10-13. 38

[Thorndike, 1953] Thorndike, R. L. (1953). Who belongs in the family. *Psychometrika*, pages 267–276. 41

[Tinini et al., 2017] Tinini, R. I., Reis, L. C. M., Batista, D. M., Figueiredo, G. B., Tornatore, M., and Mukherjee, B. (2017). Optimal placement of virtualized bbu processing in hybrid cloud-fog ran over twdm-pon. In *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pages 1–6. 23

[Trinh et al., 2019] Trinh, H. D. et al. (2019). Detecting Mobile Traffic Anomalies through Physical Control Channel Fingerprinting: A Deep Semi-Supervised Approach. *IEEE Access*, 7:152187–152201.

[Wang et al., 2019] Wang, K., Yu, X., Lin, W., Deng, Z., and Liu, X. (2019). Computing aware scheduling in mobile edge computing system. *Wireless Networks*. 2, 16, 17, 22

[Watanabe and Machida, 2012] Watanabe, K. and Machida, M. (2012). Outdoor LTE infrastructure equipment (eNodeB). *FUJITSU Scientific Technical Journal*, 48(1):27–32. 44

[Xia et al., 2019] Xia, W., Quek, T. Q. S., Zhang, J., Jin, S., and Zhu, H. (2019). Programmable hierarchical C-RAN: from task scheduling to resource allocation. *IEEE Transactions on Wireless Communications*, 18(3):2003–2016. 5

[Xu et al., 2017] Xu, Z., Wang, Y., Tang, J., Wang, J., and Gursoy, M. C. (2017). A deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs. *IEEE International Conference on Communications (ICC)*, pages 1–6. 6, 13, 15, 22

[Xu et al., 2017] Xu, Z., Wang, Y., Tang, J., Wang, J., and Gursoy, M. C. (2017). A deep reinforcement learning based framework for power-efficient resource allocation in cloud rans. In *2017 IEEE International Conference on Communications (ICC)*, pages 1–6. 23

[Yan et al., 2018] Yan, Z., Peng, M., and Daneshmand, M. (2018). Cost-aware resource allocation for optimization of energy efficiency in fog radio access networks. *IEEE Journal on Selected Areas in Communications*, 36(11):2581–2590. 23

[Yousefpour et al., 2019] Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., Kong, J., and Jue, J. P. (2019). All one needs to know about fog computing and related edge computing paradigms: A complete survey. *Journal of Systems Architecture*, 98:289 – 330. 1, 2

[Yu et al., 2016] Yu, Y., Jindal, V., Yen, I., and Bastani, F. (2016). Integrating Clustering and Learning for Improved Workload Prediction in the Cloud. In *2016 IEEE 9th International Conference on Cloud Computing (CLOUD)*, pages 876–879. 2, 6, 10, 11, 21

[Zhang et al., 2017] Zhang, F., Zheng, J., Zhang, Y., and Chu, L. (2017). An efficient and balanced bbu computing resource allocation algorithm for cloud radio access networks. In *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, pages 1–5. 23

[Zhou et al., 2018] Zhou, Y., Peng, M., Yan, S., and Sun, Y. (2018). Deep reinforcement learning based coded caching scheme in fog radio access networks. In *2018 IEEE/CIC International Conference on Communications in China (ICCC Workshops)*, pages 309–313. 23

Appendix A

Publications

This appendix presents the articles published/submitted since the beginning of the masters until this work to its conclusion. These articles are papers resulted from the investigations about F-RAN and ML for decision-making, presenting the main solution and results detailed in this thesis proposal.

- Title: Optimal Allocation of vBBUs Considering Distance Between MDC and RRH in F-RANs
  - Authors: **Jonathan M. De Almeida**, Luiz A. DaSilva, Cristiano B. Both, Célia G. Ralha, Marcelo A. Marotta
  - Conference: 54th IEEE International Conference on Communications (ICC)
  - Submitted in October 2019 / Published in June 2020
- Title: Integrating Decision-making in AI-driven F-RANs Considering Different Time Granularities
  - Authors: **Jonathan M. De Almeida**, Luiz A. DaSilva, Cristiano B. Both, Célia G. Ralha, Marcelo A. Marotta
  - Journal: IEEE Vehicular Technology Magazine
  - Impact Factor: 7.921
  - Submitted in June 2020 / Revisions in February 2021
    - Also submitted to IEEE Network Magazine (Special Issue) in January 2020 / Major Revisions in March 2020 / Rejected in May 2020

- Title: Data-driven Anomaly Detection with Traffic Pattern Categorization in Mobile Cellular Networks
  - Authors: **Jonathan M. De Almeida**, Camila F.T. Pontes, Luiz A. DaSilva, Cristiano B. Both, João C. Gondim, Célia G. Ralha, Marcelo A. Marotta
  - Journal: IEEE Transactions on Network and Service Management
  - Impact Factor: 4.682
  - Submitted in November 2020