UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Department of Network Engineering (ENTEL)

Wireless Networks Group (WNG)

Ph.D. Thesis

EFFICIENT SHARING MECHANISMS FOR VIRTUALIZED MULTI-TENANT HETEROGENEOUS NETWORKS

Ph.D. Candidate:
Matteo Vincenzi

Directors:
Dr. **Elena Lopez Aguilera**
Dr. **Eduard Garcia-Villegas**

May 2022

Abstract

The explosive data traffic demand characterizing mobile communications' evolution towards 5G has stressed the need for a considerable network upgrade, in order to enable the innovative services envisioned for the near future. Network operators are trapped in a vicious cyrcle represented on one side by the physical resource constraints (e.g., the cost of power consumption and the scarcity of spectrum resource) and, on the other side, by insufficient financial incentives for deploying next-generation networks. Indeed, typically, network owners are also the service providers, which charge the end users with relatively low and flat tariffs, independently of the service enjoyed beyond the network access. A fine-scale management of the network resources is regarded as one of the candidate solutions, both for optimizing costs and resource utilization, as well as for enabling new synergies among network owners and third-parties. In particular, network operators could open their networks to third parties by means of fine-scale sharing agreements over customized networks for enhanced service provision, in exchange for an adequate return of investment for upgrading their infrastructures.

The main objective of this thesis is to study the potential offered by fine-scale resource management and sharing mechanisms for enhancing service provision and for extending the traditional business model towards a sustainable road to 5G. More precisely, the state-of-the-art architectures and technologies for network programmability and scalability are studied, together with a novel paradigm for supporting service diversity and fine-scale sharing. In this direction, we review the limits of conventional networks, we extend existing standardization efforts and define an enhanced architecture for enabling next-generation networks' features, which are based on the concepts of network-wide centralization and programmability.

The potential of the proposed architecture is assessed in terms of flexible sharing and enhanced service provision, while the advantages of alternative business models are studied in terms of additional profits to the network operators. We start with the study of the data rate improvement achievable by means of spectrum and infrastructure sharing among operators and with the evaluation of the profit increase justified by a better service provided. In detail, we present a scheme based on coalitional game theory for assessing the capability of accomodating more service requests when a cooperative approach is adopted, and for studying the conditions for beneficial sharing among coalitions of operators. Results show that collaboration is always beneficial, also in case of unbalanced cost redistribution within coalitions when sufficient tariffs are paid by the end users. However, coalitions of equal-sized operators typically provide better profit opportunities and require lower tariffs.

The second kind of sharing interaction considered in this thesis is the one between operators and third-party service providers, in the form of fine-scale provision of customized portions of the network resources. In order to assess the potential of the proposed architecture within this framework, we define a policy-based admission control mechanism, whose performance is compared with reference strategies. The proposed mechanism is based on auction theory and computes the optimal admission policy at a reduced complexity for different traffic loads and allocation frequencies. Because next-generation services include delay-critical services, we compare the admission control performances of conventional approaches with the proposed one. Results prove that the proposed approach offers near real-time service provision and reduced complexity. Besides, it guarantees high revenues and low expenditures in exchange for negligible losses in terms of fairness towards service providers.

To conclude this thesis, we study the case where adaptable timescales are adopted for the policy-based admission control, in order to promptly guarantee 5G service requirements over traffic fluctuations. In particular, in order to reduce complexity, we consider the offline pre-computation of admission strategies with respect to reference network conditions, then we study the extension to unexplored conditions by means of computationally efficient methodologies. Performance is compared for different admission strategies by means of a proof of concept on real network traces, with particular attention for delay critical services. Results show that the proposed strategy provides a tradeoff in complexity and performance with respect to reference strategies, while reducing resource utilization and requirements on network awareness.

Acknowledgements

This thesis is for all those who had the patience to see it accomplished, but mostly for those who gave me strength and motivation when I had none.

A special thanks goes to Nikos and Mikel, initially colleagues, who then became dear friends and fellow fighters; to my family and to all the great people I met in Fano, Bologna, and Barcelona, who never left me alone and always found a way to make me being a better person. I cannot avoid mentioning Luca, Michele, Giacomo, Antonio, along with Valentina, Fabrizio, Jesus, i totes les formiguetes, germanes i germans de Barcelona.

Thanks also to the 5GAuRA project's colleagues, Akshay, Rony, Girma, Lanfranco, Kafi, Jian, Christos, Tareq, Xiaojun, Asif and Meysam, with whom I shared this journey and many good memories.

Last but not least, a sincere thanks to my doctoral thesis advisors Elena and Eduard, for giving me a huge opportunity when I needed it, for guiding me with infinite patience, and for helping me to be a better researcher.

Propterea fit uti magnum volgata per aevom, omnigenus coetus et motus experiundo, tandem conveniant ea quae coniecta repente, magnarum rerum fiunt exordia saepe.

Lucretius, De Rerum Natura, V, 427-430

We know the past but cannot control it. We control the future but cannot know it.

Claude Shannon

List of Publications

[**C1**] **M. Vincenzi**, A. Antonopoulos, E. Kartsakli, J. Vardakas, L. Alonso, and C. Verikoukis, "Cooperation incentives for multi-operator C-RAN energy efficient sharing," in *2017 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2017.

[**J1**] **M. Vincenzi**, A. Antonopoulos, E. Kartsakli, J. Vardakas, L. Alonso, and C. Verikoukis, "Multi-tenant slicing for spectrum management on the road to 5G," *IEEE Wireless Communications*, vol. 24, no. 5, pp. 118–125, 2017. (Area: Telecommunications; Quartile Q1; IF: 9,202).

[**J2**] **M. Vincenzi**, E. Lopez-Aguilera, and E. Garcia-Villegas, "Maximizing infrastructure providers' revenue through network slicing in 5g," *IEEE Access*, vol. 7, pp. 128283–128297, 2019. (Area: Telecommunications; Quartile Q1; IF: 3,745).

[**J3**] **M. Vincenzi**, E. Lopez-Aguilera, and E. Garcia-Villegas, "Timely admission control for network slicing in 5G with machine learning," *IEEE Access*, vol. 9, pp. 127595–127610, 2021. (Area: Telecommunications; Quartile Q2; IF: 3,367).

Contents

List of Figures

List of Tables

List of Acronyms

3GPP	3rd Generation Partnership Project
5G-PPP	5G Infrastructure Public Private Partnership
AA	Always-Admit
AF	Application Function
AMF	Access and Mobility Management Function
API	Application-Programming Interface
AT	Above Threshold
AUSF	Authentication Server Function
BB	Best Bid
BBU	Base-Band Unit
BBF	Broadband Forum
BS	Base Station
BSS	Business Support System
CA	Carrier Aggregation
CAPEX	Capital Expenditures
CHF	CHarging Function

CN	Core Network
CP	Control Plane
COTS	Commercial-Off-The-Shelf
CQI	Channel Quality Indicator
C-RAN	Cloud-RAN
CTMC	Continuous-Time Markov Chain
DC	Data Center
DN	Data Network
DSA	Dynamic Spectrum Access
DTMC	Discrete-Time Markov Chain
EM	Element Manager
eMBB	enhanced Mobile Broadband
EMS	Element Management System
eNB	eNodeB
EPC	Evolved Packet Core
E2E	End-to-End
ES	Exhaustive Search
ETSI	European Telecommunications Standards Institute
FCFS	First-Come-First-Served
FDD	Frequency-division duplexing
FFT	Fast Fourier Transform
GOPS	Giga Operations Per Second
H-CRAN	Heterogeneous C-RAN
HetNet	Heterogeneous Network

HSS	Home Subscriber Server
IETF	Internet Engineering Task Force
IFFT	Inverse Fast Fourier Transform
InP	Infrastructure Provider
IoT	Internet of Things
IP	Internet Protocol
ISG	Industry Specification Group
IT	Information Technology
LAA	Licensed-Assisted Access
LOS	Line-of-sight
LTE	Long-Term Evolution
MANO	Management and orchestration
MC	Macro Cell
MCS	Modulation-and-Coding Scheme
MIMO	Multiple-Input and Multiple-Output
ML	Machine Learning
MME	Mobility Management Entity
mMTC	massive Machine-Type Communications
mmWave	millimeter-Wave
MNO	Mobile Network Operator
MOP	Master Operator
MOP-NM	MOP network manager
MVNO	Mobile Virtual Network Operator
NaaS	Network-as-a-Service

NE	Network Element
NEF	Network Exposure Function
NF	Network Function
NFV	Network Function Virtualization
NFVI	NFV infrastructure
NN	Neural Network
NOE	Network Operation Engine
NOS	Network Operating System
NRF	Network Resource Function
NS	Network Services
NSACF	Network Slice Admission Control Function
NSI	Network Slice Instance
NSSF	Network Slice Selection Function
NTU	Non Transferable Utility
OFDM	Orthogonal Frequency-Division Multiplexing
ONF	Open Networking Foundation
ONOS	Open Network Operating System
OPEX	Operating Expenditures
OSS	Operations Support System
OPNFV	Open Platform for NFV
OTT	Over-The-Top
OVSDB	Open vSwitch Database
PCF	Policy Control Function
PCRF	Policy and Charging Rules Function

PDN	Packet Data Network
P-GW	Packet Data Network Gateway
PoC	Proof of Concept
POP	Participating Operator
POP-NM	POP network manager
PRB	Physical Resource Block
QoS	Quality of Service
RAN	Radio Access Network
REST	REpresentational State Transfer
RESTCONF	REST Configuration Protocol
RF	Radio Frequency
ROI	Return of Investment
RRH	Remote Radio Head
SBA	Service Based Architecture
SC	Small Cell
SCEF	Service Capability Exposure Function
SD	State Dependent
SD-WAN	Software-Defined Wide Area Network
SDN	Software Defined Networking
SDNi	Software Defined Network interface
SEES	Service Exposure and Enablement Support
S-GW	Serving Gateway
SI	State Independent
SLA	Service Level Agreement

SMF	Session Management Function
SNR	Signal-to-Noise Ratio
SNIR	Signal-to-Noise-and-Interference Ratio
SoA	State-of-the-Art
SON	Self-Organizing Network
SP	Service Provider
TCO	Total Cost of Ownership
TTI	Transmission Time Interval
TX	Transmission
UDM	Unified Data Management
UE	User Equipment
UP	User Plane
UPF	User Plane Function
uRLLC	ultra-Reliable and Low Latency Communications
VIM	virtualized infrastructure manager
VM	Virtual Machine
VNF	Virtual Network Function
XML	Extensible Markup Language

Chapter 1

Introduction

The global traffic is exploding both in terms of volume and services provided, due to the growing number of heterogeneous devices connected to the Internet (e.g., smartphones, tablets, personal computers, autonomous devices) and to the widespread adoption of diverse and bandwidth-greedy applications. In particular, experts foresee that the number of devices connected through the Internet protocol (IP) will expand worldwide to somewhere between 3.9 billion in 2018 and 5.3 billion by 2023 [1], with the number of mobile devices growing by 150%, of which about the 10% will be equipped with a 5G connection. Mobile data traffic by itself will determine a 100-fold capacity increase by the end of 2023, and the average downstream connection speed will grow from 2.0 Mbps in 2015 to nearly 43.9 Mbps by 2023 [1, 2], reaching 575 Mbps in case of 5G connections.

Huge challenges are set for 5G mobile networks in order to meet stringent requirements of next-generation services. Besides traditional communication devices, the Internet of Things (IoT) paradigm is expected to provide connection to almost everything, including surveillance and medical systems, home appliances, industrial devices and vehicles. Indeed, IoT devices are expected to be the most growing class of devices, with an annual rate of 30%, followed by smarthpones with the 7% [1]. In addition, service requirements defined by industrial players and governmental bodies for 5G are very strict [1, 3–6]: i) sub-millisecond latencies for delay-critical services, ii) a 100-fold capacity increase to serve the needs of new applications and network hyperdensification, and, iii) Quality of Service (QoS) and policy control for

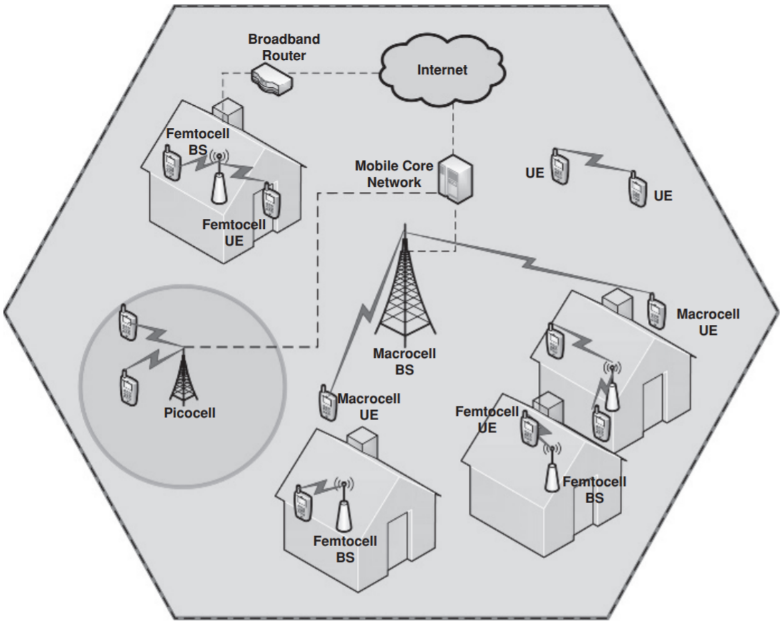


Figure 1.1: Heterogeneous cellular wireless network [7].

reliable communications. More precisely, 5G is considered the enabling technology for three main service types: i) enhanced Mobile Broadband (eMBB) services with high throughput and mobility demands, ii) ultra-Reliable and Low Latency Communications (uRLLC) setting strict requirements in terms of delays and reliability, and, iii) massive Machine-to-Machine Communications (mMTC) requiring low data rates for massive IoT-like deployments.

The transition between successive generations of mobile communication networks represents a challenging problem to be solved by mobile network operators (MNOs) in terms of: i) the infrastructure upgrade needed for enhancing the QoS and supporting new services, ii) the physical resource constraints, such as, the availability of frequency bands, the propagation features within the available bands, the transmitted power, the physical location of the antennas with respect to the covered area, and, iii) the business model to adopt in order to attract (new) end users while covering deployment costs and guaranteeing long-term revenues.

Traditionally, network densification has represented one of the main strategies for fully exploiting the scarcity of spectrum resources. In particular, heterogeneous networks (HetNets) have been adopted for maximizing the spatial utilization of licensed frequency bands. Indeed, an additional layer of base stations (BSs) with small transmit power, that is, the small cells (SCs) layer, is deployed on top of the conventional macro cell (MC) layer, composed by BSs providing coverage over a wide geographic area, as represented in Figure 1.1. The small transmit power and coverage characterizing SCs provide a very efficient way of using the spectrum resource, as a high frequency reuse factor can be adopted[1]. Besides, SCs can be exploited for filling coverage holes in the MC layer, or to provide additional capacity in high-traffic areas (e.g., stadiums or shopping malls). HetNets have been widely deployed and, in 2015, the 51% of total mobile data traffic was offloaded onto SC and Wi-Fi networks [2].

Thanks to enhanced interference management techniques and improved transmission technologies (e.g., coordinated multiantenna/multipoint transmission technologies [8]), the Shannon upper bound on spectral efficiency can be approached at the network density level needed for 5G deployments. In addition, although communications in the millimeter-wave (mmWave) frequencies are a well-known solution for 5G capacity boost [9], a further extension of the available bandwidth can be achieved through the harmonization of the licensed spectrum utilization among different MNOs, both in time and space domains. Indeed, the traditional approach for network design is based on the fixed fragmentation of the licensed spectrum for exclusive utilization by different MNOs, with networks being designed for supporting overestimated peak traffic volumes [10], which often results in a poor utilization of the spectrum resource [11]. Because spectrum refarming would be a prohibitively expensive and slow process, standardization bodies and manufacturers are currently very active in Dynamic Spectrum Management (DSM) for frequency sharing. The main efforts related to DSM are dedicated to:

- The extension of LTE-A carrier aggregation (CA) principle in the 5 GHz unlicensed spectrum via Licensed-Assisted Access (LAA) [12]
- The aggregation of different technologies, for instance, LTE and Wi-Fi technologies via LTE Wireless-LAN Aggregation (LWA) [13]
- The application of cognitive radio principles, for tiered dynamic spectrum access (DSA) [14]
- The implementation of flexible and scalable network sharing solutions [15]

[1] A frequency reuse of one is normally used for macro and small cell network layers in long-term evolution (LTE) networks [8].

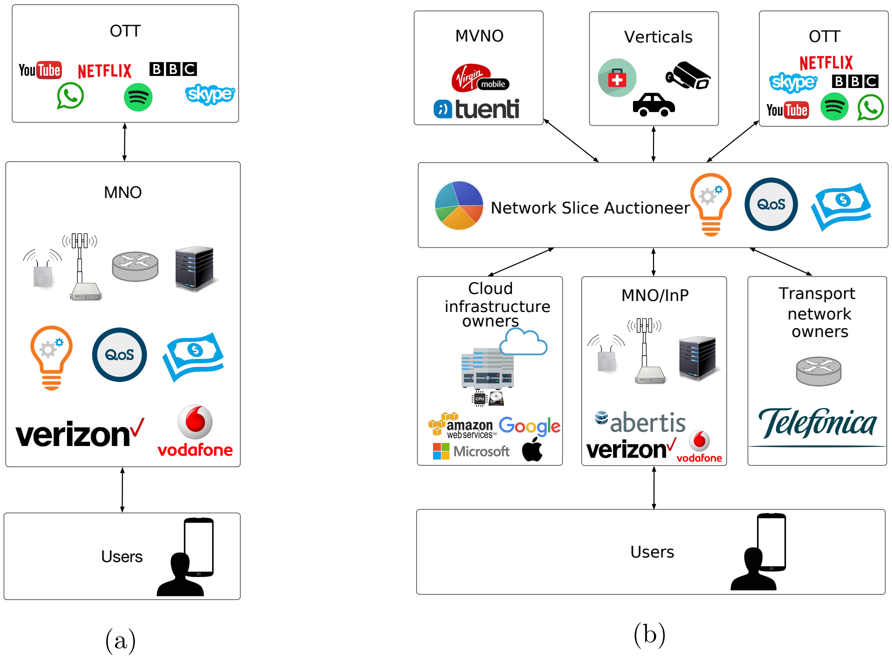


Figure 1.2: Business model with: a) Standalone deployment of E2E infrastructure; b) Heterogeneous ecosystem of market players.

Conventionally, long term sharing agreements are established between MNOs at coarse granularities (i.e., months/years) [10], while fine-scale sharing mechanisms are needed for optimizing resource usage and for supporting, at a reasonable cost, a massive number of devices and diverse services. In conclusion, a new paradigm for fine-scale resource sharing is needed as an alternative strategy for mitigating the underutilization and scarcity of physical resources, as well as for making 5G networks sustainable on the long-term. Indeed, BSs are the most expensive component of conventional radio access network (RAN), with operating expenditures (OPEX) representing the 60% of the Total Cost of Ownership (TCO) [16]. Therefore, the BSs densification in future network deployments has to be complemented by the optimization of network operations in order to both improve the energy-efficiency and reduce costs and CO_2 emissions.

In the direction of 5G, a revolutionary network upgrade is needed for providing customized QoS support to a huge variety of heterogeneous services, while keeping complexity low by designing a flexible architecture. However, from MNOs' perspective, the costs for such upgrades cannot be covered by the traditional business model. Indeed, the standalone scenario depicted in Figure 1.2a, where a few big MNOs deploy

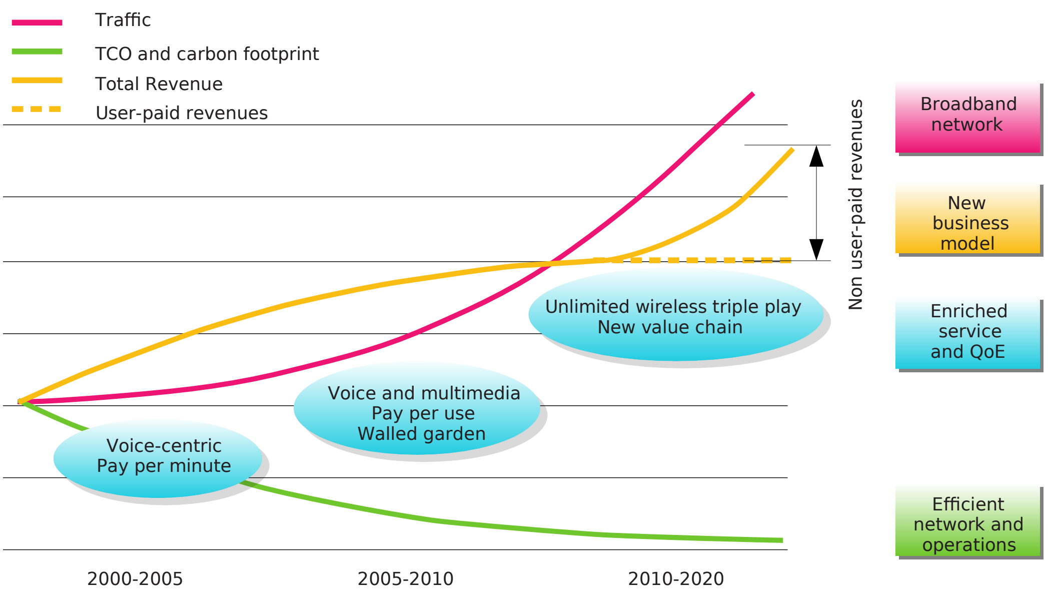


Figure 1.3: Mobile service provision: trends and challenges (based on [17]).

and upgrade their own infrastructure for end-to-end (E2E) service provisioning, with the main return of investment (ROI) represented by the relatively low-cost flat tariffs charged to the end users [17], leads to an increasing gap in revenues, as detailed in Figure 1.3.

New value chains have to be introduced by MNOs for addressing the increasing gap in revenues experienced over the last decade and, in this direction, sharing and multi-tenancy have been widely adopted over the years for improving networks profitability. Firstly, passive sharing appeared for reducing capital expenditures (CAPEX) by sharing passive elements (e.g., physical sites, antenna masts, cabling, cabinets, power supply, etc.) among MNOs [18, 19]. Afterwards, in order to achieve a more efficient utilization of the licensed spectrum and for reducing OPEX, MNOs began to perform active sharing [20], that is, to act as infrastructure providers (InPs) and to lease part of their infrastructure and spectrum to mobile virtual network operators (MVNOs): i) MNOs that look for coverage and/or capacity extension in a specific geographical area, or, ii) market players willing to act as MNOs without a stand-alone network deployment and/or a spectrum license.

An alternative business model, depicted in Fig. 1.2b, has emerged in the last decade [3, 4, 21], encouraging the cooperation among MNOs and other market players. In particular, alternative market opportunities stem from the on-demand provision to third-party service providers (SPs) of customized portions of the network assets with QoS guarantess. SPs can range from MVNOs to over-the-top (OTT) players (e.g., streaming services), or even vertical industries (e.g., e-health, surveillance, automotive). In order to achieve the full potential of this new business model, the network infrastructure needs to support scalability and programmability for meeting the heterogeneous constraints set by very different services simultaneously and in a dynamic manner.

Network slicing is a novel paradigm for network sharing and resource provision at fine-scale, which is expected to be one of the keystones of the 5G architectural revolution [22]. Indeed, it foresees the dynamic isolation of QoS-tailored portions of network resources into customized virtual networks (i.e., the network slices), leased to third-party SPs (i.e., the slice tenants) for service provision. In other words, the alternative business model introduced above can be extended to the fine-scale provision of network slices, giving rise to the 5G *slice market*. In this particular marketplace, network slices would be the traded commodity, whereas InPs would be the responsible for continuous technology upgrade, and SPs the middlemen bargaining over the network resources and providing the finished product to the end users. From a contractual point of view, QoS requirements associated with a specific service could be guaranteed by a service level agreements (SLA) between InP and SPs detailing the characteristics of the slice leased (e.g., nominal throughput, maximum delay, resource holding time, shared/exclusive access to slice resources, etc.).

The concept of slice market is expected to introduce a strong competition between different InPs and SPs, thus oxygenating the typically closed and monolithic ecosystem of telecommunication services and introducing the preconditions for fast innovation. Indeed, on the one hand, InPs could better manage and monetize the utilization of their resources and, on the other hand, any SP could possibly enter the market of mobile communications, independently of the ownership of a network infrastructure.

1.1 Objectives and Contributions

This thesis is motivated by the incompatibility of conventional networks and business model with 5G objectives, both in terms of flexibility for supporting next-generation services and additional revenues for covering the costs of the network upgrade. The aim of this thesis is to provide a thorough analysis of architectural, technological and methodological solutions for fine-scale slicing, the enabling framework for a sustainable transition towards 5G networks and services. On the one hand, we define the infrastructure upgrades needed for supporting next-generation services and business model, and, on the other hand, we assess the financial incentives achievable for the deployment of 5G networks. In particular, we study in detail the impact on the value chain of different sharing interactions between MNOs and the rest of the market ecosystem, that is, other MNOs or SPs. The global objectives and contributions of this thesis can be categorized as follows:

- Definition of enhancements for making next-generation networks' architecture flexible and sustainable. The new architecture is an extension of pre-5G standardization activities and integrates technologies that enable more agile sharing interactions, and that support programmable policies, custom QoS and effective service prioritization. The challenges associated with these novel features are discussed for different network segments, besides, the recent standardization efforts in this direction are also presented. The proposed architecture was published in IEEE Wireless Communications Magazine (**J1**).

- Study of the benefits, in terms of spectrum utilization efficiency, enabled by spectrum and infrastructure sharing. We consider the case where multiple MNOs pool their infrastructure and licensed spectrum for sharing underutilized resources on a fine-scale, thus improving coverage and offered data rates without the need of purchasing extra frequency bands, or deploying new network nodes. A cooperative approach is capable of accomodating more service requests, and providing higher data rates. Performance, in terms of data rate increase, is obtained through simulations and compared for the stand-alone and cooperative approaches. The results were published at the IEEE International Conference on Communications (**C1**), and in IEEE Wireless Communications Magazine (**J1**).

- Study of the financial incentives offered to MNOs by spectrum and infrastructure sharing. On the one hand, higher revenues, in terms of tariffs charged by MNOs, are enabled and justified by a better coverage and enhanced data rates. On the other hand, the increase in OPEX, associated with the pooling of the spectrum in a shared infrastructure, can be redistributed in different ways among MNOs.

A novel scheme based on coalitional game theory is developed, where user pricing, MNOs' characterization and coalitions' size are used as parameters for deciding whether to follow a cooperative or stand-alone approach. The results were published at the IEEE International Conference on Communications (**C1**), and in IEEE Wireless Communications Magazine (**J1**).

- Proposal of mechanisms for custom and fine-scale slice allocation to SPs, and study of the financial incentives offered to MNOs. According to the alternative business model, additional revenues are enabled for MNOs when acting as InPs and charging third-party SPs for the leasing of network slices. Because network resources are limited in relation to slice requests, mechanisms are defined for the admission of such requests, while maximizing InPs revenues and finding a tradeoff between resource utilization and fairness towards the SPs. Auction theory is used for modeling the SLAs bargaining among InPs and SPs, and admission control policies are defined and compared for on-demand and periodic slice allocation. As far as we know, this is the first effort in comparing on-demand and periodic slicing with respect to fairness towards SPs, resource utilization, InP's profit, and timeliness. The results were published in IEEE Access (**J2**).

- Definition of enhanced mechanisms for timely slice allocation at reduced complexity. Motivated by the strict and diverse requirements foreseen for 5G, network slicing represents a valid solution for QoS customization and resource sharing in a dynamic and scalable way. The exclusive allocation of slices to different SPs has a great potential in terms of QoS guarantees, mostly for services with very strict requirements. On the other hand, the high complexity typically associated with an exclusive allocation, could harm timeliness, customization and efficiency in the resource utilization. A policy-based admission control mechanism is defined for exclusive network slicing at fine and adaptable timescales, while reducing complexity by offline pre-computation of the admission strategies. In particular, optimal admission strategies are computed by means of exhaustive search for sample network state conditions, which are then employed for the training of a neural network in case of unexplored state conditions, as well as for clustering operations capable of providing network-wide solutions. Different policies and mechanisms for the computation of the admission strategies are compared, and results are provided in terms of efficiency in resource utilization, fairness to the SPs, InPs' revenue and complexity. As far as we know, this is the first study considering a variable timescale for improved customization in slice provision at a reduced increase in complexity. The results were published in IEEE Access (**J3**).

The research activities described in this thesis have been accomplished within the framework of EU Horizon 2020 Research and Innovation Programme "5GAuRA" (No. 675802). The candidate carried out the first part of his PhD at the Department of Signal Theory and Communications (TSC) with supervisors affiliated to the Telecommunications Technological Centre of Catalonia (CTTC) and to Iquadrat Informatica S.L., while the second part of the PhD has been accomplished at the Department of Network Engineering (ENTEL).

1.2 Thesis Outline

With the motivations and contributions of this thesis defined, in the remainder of this chapter we present the outline of this work.

In Chapter 2, we present the state-of-the-art (SoA) contributing to the enabling of next-generation services and networks. In particular, we first introduce novel architectures and technologies for network programmability and scalability. Aftwerwards, we review the main solutions for extending these same concepts to the RAN, together with strategies for E2E service provision at the edge of the network.

In Chapter 3, we present the roadmap for extending concepts and solutions presented in Chapter 2 to the particular case of 5G networks. More precisely, we first present the standardization efforts for the enabling of network sharing and multi-tenancy in RAN and core network (CN) achitectures of pre-5G networks (i.e., those generations of mobile networks preceding 5G). Afterwards, we study the main reasons for the unsuitability of conventional mobile networks for QoS customization and fine-scale sharing and present the network slicing paradigm as candidate solution. We also review the architectural advances standardized in pre-5G networks for flexible network sharing and QoS guarantees, thus, partially covering the lack in flexibility of conventional networks. Then, we define the necessary technological and architectural upgrades needed for extending the concepts of network programmability and slicing to 5G networks. The proposed approach includes a a new management entity for the E2E management of the slicing process, both for efficient resource utilization and QoS support, as well as for enabling a new framework for fine-scale network sharing. Finally, we compare the proposed architecture with recent standardization efforts and SoAs in network slicing for 5G.

Chapter 4 presents a novel scheme for studying the conditions for beneficial RAN sharing among coexisting MNOs. In particular, the focus is on how QoS and MNOs' profits can be improved thanks to a more efficient spectrum utilization. The problem is modeled as a coalitional game and investigated for two scenarios, when different combinations of market and spectrum share are associated to each MNO. The conditions are studied for deciding whether cooperation is advantageous with respect to a stand-alone approach, and for choosing them most profitable coalition.

Chapter 5 proposes a policy-based admission control mechanism for network slicing in 5G, for the case of competing SPs providing the same type of service. A benchmark for the performance is provided with respect to reference policies when different pools of resources, traffic loads, and slicing frequencies are considered. Two approches are compared, that is, the one where network slices are provided to the SPs in an on-demand manner, and that where slice allocation is performed periodically. The performance of the proposed solution is studied in terms of revenue rates to the InPs, resource utilization, admission rate, computational expenses and promptness of the admission control.

Chapter 6 provides a proof of concept on real network traces for exclusive slice allocation to SPs providing the same type of service. This work proposes a dynamic implementation of the methodology introduced in Chapter 5, with variable timescales suitable for 5G services. More in detail, approaches based on the offline and exhaustive search of the optimal admission strategies are combined with machine learning algorithms for computing near-optimal strategies in case of unexplored network conditions, and with clustering procedures for reducing complexity at a network level. The performance of the proposed methodology is studied in terms of the fairness guarantees and negotiation power to the SPs, revenues and expenditure reduction to the InPs, and computational-efficiency for selecting the admission strategies.

Chapter 7 summarizes the conclusions of this thesis and proposes future research directions.

Chapter 2

SoA in Network Programmability

5G, the next generation of mobile networks, is still far from its maturity in terms of deployment, however, requirements have been proposed by standardization bodies [3, 5, 6], and new technologies are being fine-tuned by the research community, while the resulting architectures and mechanisms are being integrated into 3rd Generation Partnership Project (3GPP) specifications. In this chapter, the principles standing at the base of network programmability and the candidate SoA architectures and technologies to be integrated into conventional mobile networks for an upgrade towards 5G is reviewed. First, solutions are analyzed for network programmability with support for QoS customization and dynamic sharing. Besides, the extensions needed for programmable and scalable RAN/CN are introduced, together with solutions for E2E service provision at the edge of the network.

2.1 Principles, Architectures and Technologies

The 5G Infrastructure Public Private Partnership (5G-PPP), launched by the EU Commission with the support of industry manufacturers, telecommunications operators, SPs, and small to medium-sized enterprises, released its plan and perspectives for future mobile networks. By means of multiple European projects, such as METIS, iJOIN, Mobile Cloud Networking, CROWD, and 5G-PICTURE [23], network function virtualization (NFV) and software defined networking (SDN) have been proposed as the key enabling technologies driving the mobile networks (r)evolution, and their integration as the candidate approach for E2E network programmability and scalability [24–27]. Indeed, as introduced in Chapter 1, three-fold are the novelties enabled by network programmability in terms of dynamic and adaptive resource assignment: i) the efficient utilization of scarce resources based on the actual needs (as an alternative to a static over-provisioning) [28], ii) the support for enhanced and customized services [22], and, iii) the fostering of alternative business models and revenue opportunities, thus accelerating innovation and rollouts for next-generation networks [3, 4, 21].

NFV and SDN are both solutions introduced for simplifying and enhancing the network management, with common aim of promoting innovation, creativity, openness and competitiveness [29, 30]. However, they rely on very different but complementary approaches. Indeed, while NFV optimizes the implementation of network services over the physical resources of a specific device, SDN focuses on the provisioning, management, (re)configuration, and control of (virtualized) physical resources over multiple devices. Therefore, SDN can be implemented according to the NFV framework in order to enhance its performance, facilitate its operation and simplify the compatibility with legacy deployments. However, the virtualization of network functions is not bound by SDN technologies, and vice versa [31, 32]. Below, a review of the SoA in network programmability is presented, with particular attention to the dynamic and customized resource isolation needed for a secure and profitable coexistance of competing SPs within the 5G slice market introduced in Chapter 1.

2.1.1 Network Function Virtualization

Service provision within the telecommunications industry has been traditionally deployed by the same owners of the network and, typically, by means of special-purpose proprietary equipments for each of the supply chain's functionalities. However, for the support of next-generation services, legacy appliances make network testing and upgrades increasingly difficult and costly. More in detail, novel services are being characterized by stricter and more diverse requirements, imposing the continuous purchase, upgrade, and operation of specialized appliances, which results in high CAPEX and OPEX [33]. As described in Chapter 1, increased users' subscription prices by themselves are not a viable solution for compensating the increased investments, indeed, at the scale needed for transforming legacy infrastructures towards 5G, this approach would most probably lead to customer churn. Therefore, alternative solutions have to be found for building next-generation networks while reducing expenditures and product cycles.

NFV has been proposed for addressing these challenges, as it leverages virtualization principles for flexibly deploying, managing and upgrading network devices [31, 32, 34, 35]. According to the definition in [36]:

> *Network virtualization is any form of partitioning or combining a set of network resources, and presenting/abstracting it to users such that each user, through its set of the partitioned or combined resources has a unique, separate view of the network.*

More into detail, NFV is based on the following principles [37–40]:

- **Decoupling software from hardware:** Network functionalities are separated from underlying hardware by substituting dedicated hardware with software instances deployed in commercial-off-the-shelf (COTS) devices. Consequently, network functions (NFs) are substituded by virtual network functions (VNFs) implemented in software as virtual machines (VMs) running on one or more general purpose devices. According to this approach, hardware and software can follow independent evolution paths, allowing for agile network deployment, maintenance and upgrade.

- **Flexible network function deployment:** The same physical resources can be shared by multiple VNFs with isolation guarantees for reliability and privacy concerns, besides, appliances can be reconfigured on-the-fly in order to host different VNFs over time. The set of VNFs chained together for implementing a given service can be hosted in, and migrated to, different data centers (DCs) deployed along the network. Hence, with respect to the server farm paradigm commonly used for cloud services, which is based on oversized and centralized DCs [41], multiple distributed and low capacity DCs are deployed in network locations with very diverse space and resource constraints (e.g., street cabinets). This approach enables new design features, for instance, placing DCs closer to end users is expected to reduce traffic within the CN and to be one of the enablers of uRLLCs. However, the choice on the particular set of DCs

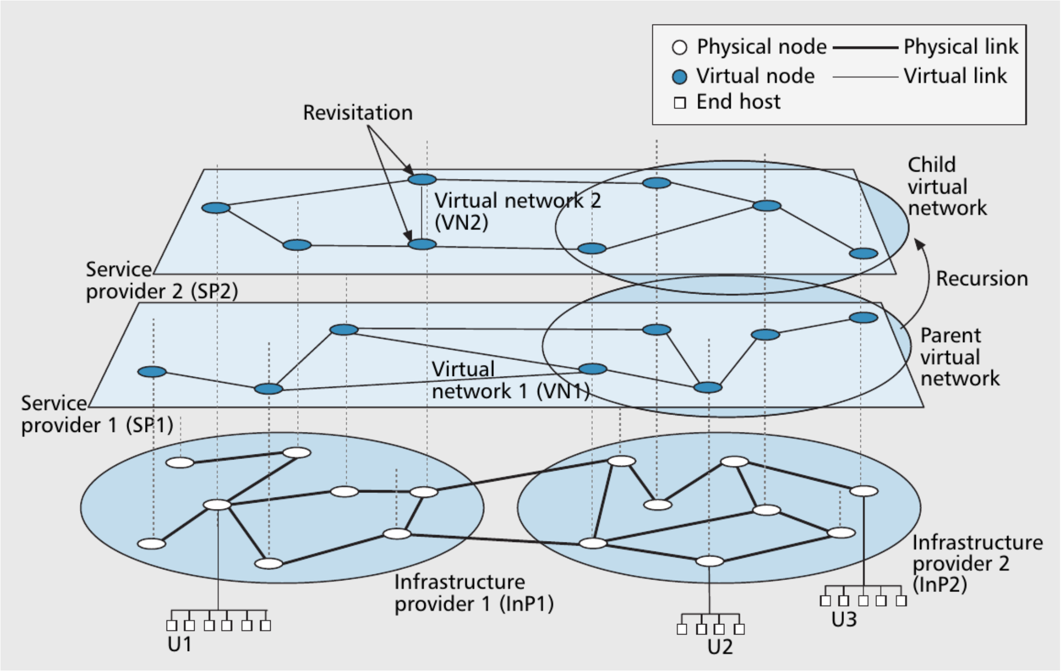


Figure 2.1: The concept of network virtualization [42].

to be used for implementing the VNFs of given NF instance is not a trivial problem, and sets new network optimization domains, which have to take into account not only QoS requirements, but also physical resources constraints. Indeed, for a given service, sufficient communication resources need to be guaranteed between the underlying VNFs, in order to respect the overall timing requirements, while balancing the traffic load among DCs, guaranteeing resiliance to network failures, and reducing power consumption.

- **Dynamic scaling:** The performance of a given service can be scaled on-demand with a fine granularity. For instance, more VNFs can be instantiated at a given time if complex operations, or massive service requests, have to be served.

In conclusion, NFV introduces the concept of virtualization into communication networks and, like the abstraction of information technology (IT) infrastructures into multiple VMs, it abstracts networks nodes and communication links into multiple virtual networks, each composed by distributed VNFs, as described in Figure 2.1. In other words, logical network functions can be dispatched by InPs to SPs as instances of plain software, which are deployed into appliances distributed between

DCs, network nodes and end user premises (i.e., high volume servers, switches and storage units). Consequently, InPs can focus in the deployment and upgrade of the physical infrastructure, while SPs can design and deploy network services with full abstraction with respect to hardware related concerns [42]. This novel paradigm contributes to building the foundations for alternative business models, and will be exploited in Chapter 4 for efficient resource utilization and sharing by InPs, and in Chapters 5 and 6 for the proposal of admission schemes for VNFs requests by SPs.

From a technical perspective, a discussion on the challenges for the introduction of the NFV framework in mobile networks is presented in [34, 43, 44], with particular focus on customer premises equipments (CPEs) and on LTE's CN, that is, the evolved packet core (EPC). Conversely, from a business perspective, the European Telecommunications Standards Institute (ETSI) has defined possible use cases and service models (i.e., roles and interaction among players involved in the service provision) for the application of the NFV framework to mobile networks [45]. More precisely, the service models typical of cloud computing could be adopted through a NFV-as-a-service paradigm, where single VNFs, or entities of the standard architecture described in Section 2.1.3, are provided on the market by InPs as services [46].

In order to better understand the potential of NFV in terms of the economy of scale, we briefly discuss the specific case of CPEs [34], however, similar observations can be made for other network components. CPEs are typically made up of multiple functional blocks (e.g., routing, switching, modem, firewall, radio, etc.), which, for the most part, are embedded in hardware. In case a modification is needed, in the best scenario, a firmware upgrade could be executed remotely or by the same customers. However, whenever substantial upgrades are needed and cannot be achieved by a new firmware release, either the intervention of a specialized technician or the substitution of the CPE could be required for every customer. In all cases, this approach would lead to additional expenses either for the SPs or for the end users. On the other hand, in a NFV-based approach, a cheaper solution could be achieved by flexibly modifying the needed VNFs at the customers' premises, or by re-allocating part of the CPE's VNFs to one of the InP's distributed DCs.

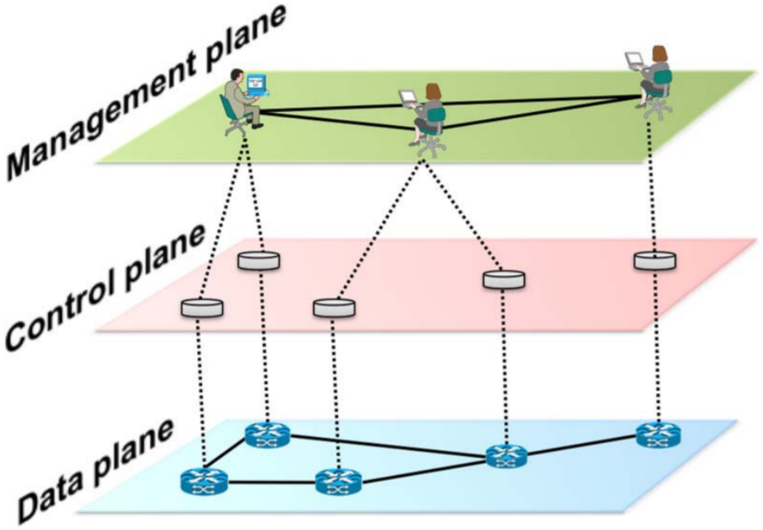


Figure 2.2: Network's layered view [47].

2.1.2 Software Defined Networking

Legacy networks are very prone to configuration errors and undesidered behaviours such as packet losses, establishment of undesidered paths, or service agreement violations [48]. Besides, as introduced in Section 2.1.1, special-purpose and proprietary systems are typically adopted by network owners, which lead to costly upgrades that impede innovation (e.g., the transition from IPv4 to IPv6 started 26 years ago and still it is not fully accomplished [49]). More in detail, in order to counteract network misconfigurations, many proprietary management devices, operating systems and applications (e.g., firewalls and deep packet inspection tools) are typically employed, and possibly developed by different vendors, which leads to high design and operational complexity [50].

An additional factor contributing to the increase of complexity and to the lack of flexibility in conventional networks is represented by the vertical integration tipically adopted, that is, to the joint deployment of control and operational planes within each network device, as shown in Figure 2.2 [47]. In particular, the high-level network policies defined by the management plane are enforced by devices' control plane into their data plane, which, in turn, executes them as opportune forwarding behaviour. In case of network upgrade, the control plane of each device involved with the modification has to be manually updated (i.e., by installing new firmware

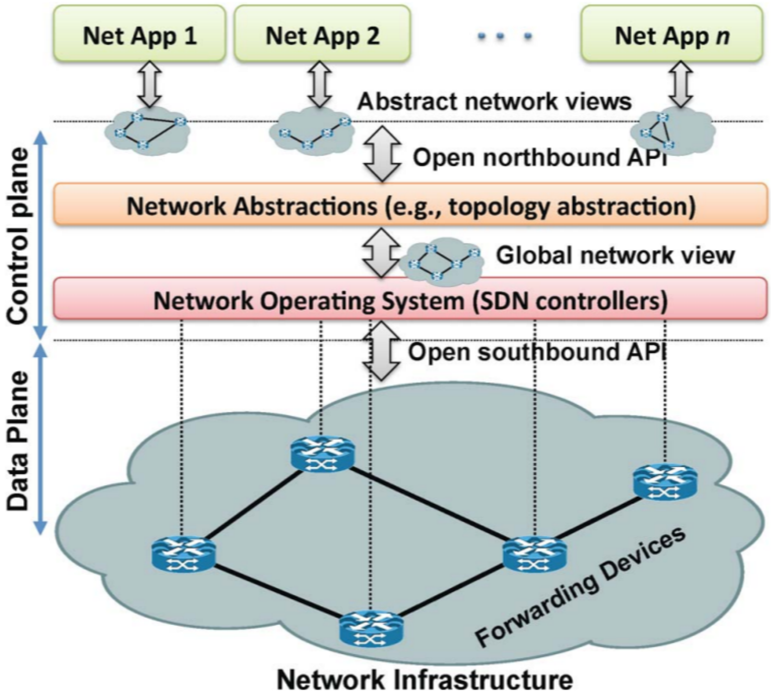


Figure 2.3: SDN architecture [47].

or, in the worst case, upgrading the hardware), which translates into costly and very long deployments [49]. A good example is represented by IP networks, which adopt distributed transport protocols embedded in switches and routers through low-level and vendor-specific control mechanisms [51]. Finally, the lack of flexibility in existing networks is a clear impairment also in the case of faults and load changes, as no or limited support can be implemented for dynamic reconfiguration, and because of the local network state awareness provided by co-located control and data planes.

SDN is an alternative architecture for transport networks introduced in 2010 by Stanford University [52], while studying a novel interface between control and data plane (i.e., OpenFlow [52–54]), which could guarantee: i) network abstraction, ii) flexibility, iii) reconfigurability, and iv) adaptability [47,55]. Below, we describe SDN key principles, which give rise to the layered architecture described in Figure 2.3 [47]:

- **Breaking the vertical integration by control and data plane decoupling:** The control logic is extracted from the network devices, which become generic packet forwarding elements. To this aim, the open southbound API (e.g., OpenFlow [52–54]) enables a driver equivalent abstraction, which separates

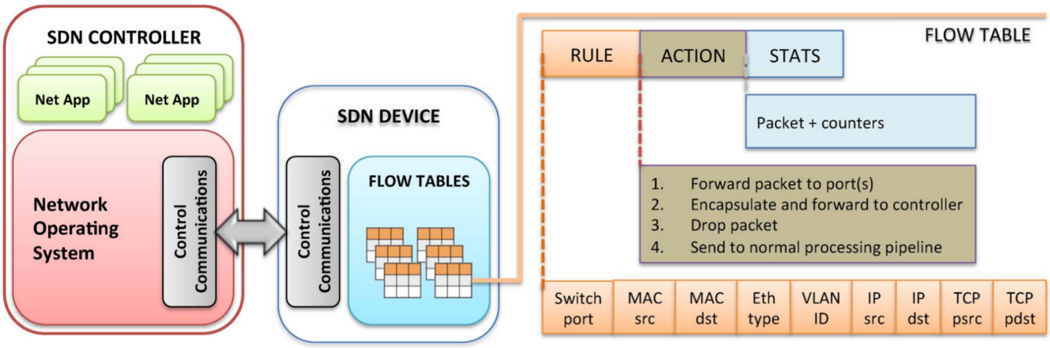


Figure 2.4: OpenFlow enabled device [47].

devices' logics from the underlying hardware, as shown in Figure 2.3.

- **Making forwarding protocols flow-based (instead of destination-based):** A flow is a sequence of packets undergoing identical service policies from source to destination, and identified by forwarding devices via a set of matching fields, as illustrated in Figure 2.4 [53]. This approach enables unique flexibility, as the forwarding behavior of a given network device (e.g., load balancer, traffic shaper, etc.) can be defined as a set of actions (e.g., dropping, or forwarding instructions) over packets with the same flow identifier. Therefore, different network components (e.g., routers, switches, firewalls) can be implemented by integrating different packet-handling rules over a generic forwarding device, in the form of flow tables' pipelines [52]. For instance, priorities and QoS customization can be expressed over different rules by following the flow tables' sequence number and the row order within each table. Besides, as shown in Figure 2.4, network awareness could be gathered by computing statistics on the matched packets, which could be used for notification tools (e.g., state change alarms) [56].

- **Centralization of the network's control logic:** The control plane is centralized and aggregated from each device into a common SDN controller, also known as network operating systems (NOS) [50]. The NOS provides network abstraction at a global level, as well as resources for flexible network programming, indeed, it is typically runned as a software platform on COTS technology. Similarly to an operating system for IT appliances, it simplifies the network reconfiguration and upgrade, by direct update and enforcement of the data plane policies through the southbound API [57].

- **Enabling network programming through high-level applications:** Network applications are deployed in software on top of the NOS, which are interpreted and enforced by the NOS over the underlying data plane infrastructure through the open northbound API [58]. As shown in Figure 2.5, the NOS is capable of configuring each network device's flow tables for implementing the forwarding behaviour required by the applications. Besides, through the NOS, information on the global status of the data plane, as well as essential services (e.g., device and link discovery, or topology astraction), can be accessed by network applications developers. In conclusion, the NOS is a platform for high-level and unified network programming, in contrast with the proprietary and device-specific control tools tipically used in legacy networks (e.g., Cisco IOS and Juniper JunOS). Finally, the NOS is also responsible for implementing security mechanisms capable of guaranteeing hierarchical isolation among the rules created by different priority applications (i.e., the rules generated by a given application should only be able to modify those created by lower priority applications [59, 60]).

It is important to remark that a logically centralized control plane does not necessary require a physically centralized system. Indeed, at least in big network deployments, a single entity for managing the whole network would result in poor performance and scalability, and it would represent a single point of failure, thus providing poor reliability guarantees [61, 62]. Consequently, a combination of physically distributed NOS is tipically adopted [61, 63, 64], each managing a different portion of the network. To this aim, north and southbound APIs are completed by east/westbound APIs, as illustrated in Figure 2.6, for the interconnection and coordination of multiple controllers (e.g., for exchanging controlled devices, topological information or network status, as well as for monitoring and notification services).

With respect to the role of SDN in the realization of fully programmable networks, the abstraction achievable through high-level network management applications is expected to address the limitations of the low-level instruction sets typically used in conventional networks (i.e., complex and device-specific configurations, incomplete or conflicting forwarding rules, fault sensitivity, etc.) [60]. Indeed, the easier (re)configuration of the forwarding devices would allow the network programmers to focus on the development of novel and advanced functionalities [60]. More in detail, code modularity and portability could be available for network application

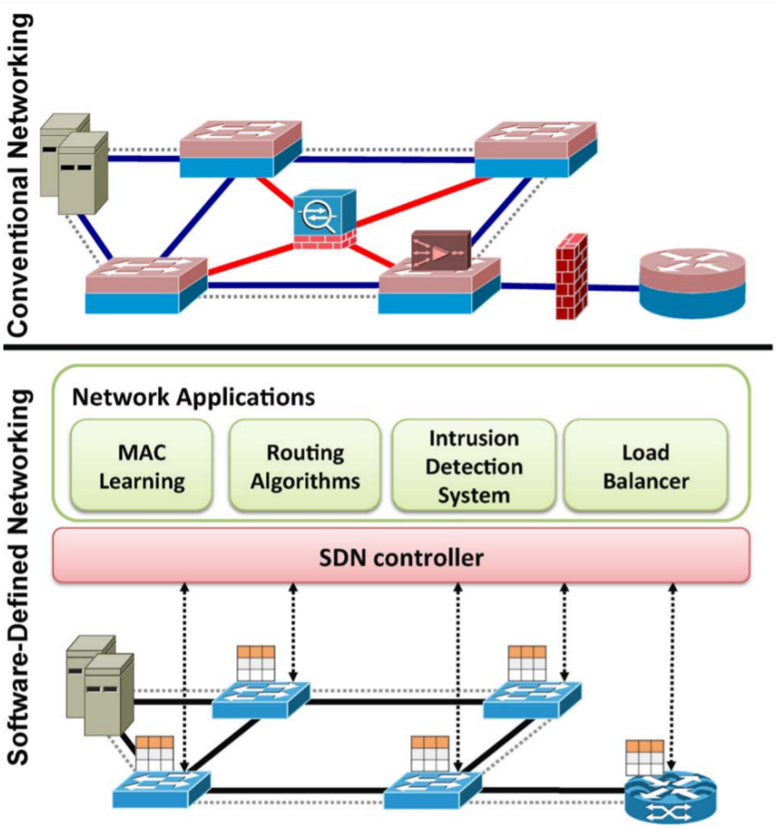


Figure 2.5: Conventional versus SDN networks [47].

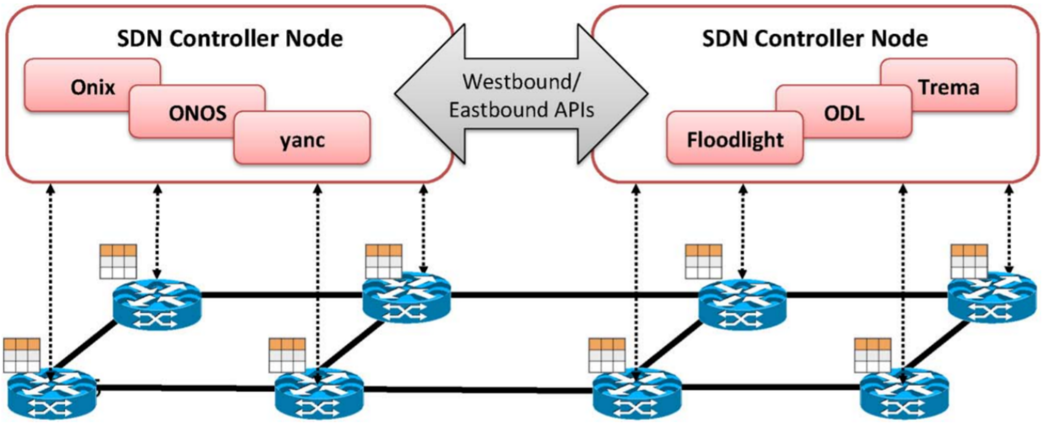


Figure 2.6: Distributed control and east/westbound APIs [47].

programmers, and integrated development environments (e.g., NetIDE [65]) are expected to arise with the aim of speeding up applications' lifecycle, and reducing deployment errors [66]. Besides, software libraries could be deployed for supporting heterogeneous scenarios with customized network applications (e.g., home and enterprise networks, DCs, or Internet exchange points). Finally, high-level network programs could be exploited for defining virtual network topologies, which could be used for providing a simplified representation of the entire network, or for implementing network sharing by splitting the physical infrastructure into multiple virtual networks assigned to different tenants [67].

The key concepts that characterize SDN, that is, network programmability and the centralization of the control logic, will be embedded in the architecture proposed in Chapter 3 for E2E network programmability in 5G. The same concepts will be adopted in Chapter 4 for the study of efficient network sharing schemes, which provide enhanced QoS to the users subscribers and financial incentives to the MNOs.

2.1.3 Standardization and Rollouts

The ETSI NFV industry specification group (ISG) focuses on the standardization of architectural and operational solutions for the adoption of the NFV framework in communication networks, with focus on the management and orchestration of VNFs. More precisely, the ETSI NFV ISG has defined the virtualization requirements [68], as well as the management and orchestration (MANO) architecture for the NFV framework [37], specifying its functional components, interfaces, and related application-programming interfaces (APIs). Besides, the ETSI NFV ISG has studied performance, reliability and security related to the NFV framework [69], and it has provided instructions for real-life and multi-party proof of concept (PoC) implementations [70].

Below we present the ETSI MANO NFV architecture, which is composed by three main components as depicted in Figure 2.7 [32, 34, 37, 40]:

- **NFV infrastructure (NFVI):** The NFVI is the combination of the hardware and software resources necessary for deploying VNFs (i.e., computation, storage and DCs' connectivity). A virtualization layer (e.g., hypervisor-based [39, 71]) extracts the required resources from the underlying hardware and makes them available for VNFs deployment in the form of VMs with custom computation, storage and network units [39]. When virtualization is not implemented by design on the available hardware, a virtualization layer can be installed on top of an operating system, or VNFs can be deployed as applications/containers [39, 72].

- **Netwok Services (NS):** The NS are the ordered composition of multiple logical blocks, that is, the VNFs, each with specific functionalities (e.g., gateways, firewalls, traffic shapers, etc.) and with precise interfaces for interconnection. According to a virtualized approach, VNFs can be implemented as the chaining of multiple VMs, possibly running in different hardware devices. VNFs are administered by element management systems (EMSs), which take care of VNFs' creation, configuration, monitoring, performance and security. EMSs provide SPs with essential information and tools for deploying and managing high-level services, with full integration with conventional network management systems, such as the operations support system (OSS) and the business support system (BSS).
- **NFV MANO:** The NFV MANO is a vertical network management tool, that guarantees the functionalities needed for VNFs provisioning, monitoring and adaptation, both in terms of resource lifecycle, through virtualized infrastructure managers (VIMs), as well as at a service level, through VNF managers and the high-level NFV orchestrator. Therefore, the NFV MANO handles all virtualization-related tasks necessary for supporting the NFV framework, and has a full control over the underlying infrastructure and virtualization framework. Besides, by being interconnected with standard network management systems (i.e., OSS and BSS) it allows the coordination with legacy equipments and, therefore, it provides MNOs and InPs with a unified framework for the management of their networks with full transparency with respect to the underlying technology.

The Linux Foundation launched the open platform for NFV (OPNFV) project [40,73] for accelerating the adoption of the NFV framework, which produced multiple novel components for OpenStack, an open source and standard cloud computing platform. Indeed, although many other cloud computing platforms and controllers exist in the market (e.g., open source CloudStack and Eucalyptus, or commercial solutions such as Microsoft Azure, Google Cloud, Amazon Web Services), OpenStack has become the de facto technology for the VIMs due to its wide adoption by the IT industry

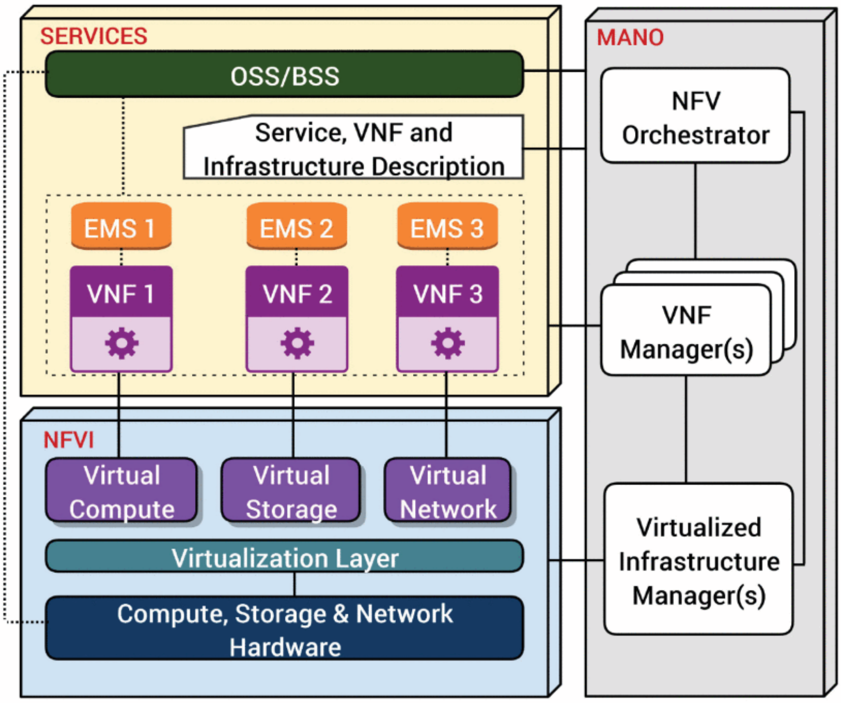


Figure 2.7: Network function virtualization architecture [35].

for managing large-scale cloud deployments. In particular, OpenStack provides a solution for Infrastructure-as-Service (IaaS), according to which it is possible the composition of different services (e.g., for orchestration, networking, computation, storage, etc.), each with an open API for easy integration. In the past releases, thanks to the OPNFV efforts, OpenStack adapted some of its services for supporting multiple features required by NFV, which have been successively included in the ETSI NFV ISG standardization efforts [74].

As introduced in Section 2.1.2, another possible approach for the innovative deployment of communication networks consists in principles such as the logical centralization of the control plane, separated from the data plane, and in the programmability of flow-based network functionalities. Although these design principles find their roots in a series of paradigm shifts proposed in the past fourty years [50, 75, 76], they gained the attention of the literature and of telecommunication industry only during the last decade. Indeed, over time, the motivations introduced in Chapter 1 pushed vendors and InPs towards a more competitive and profitable implementation of next-generation networks, which began with the proposal of OpenFlow as southbound API for SDN [40, 52].

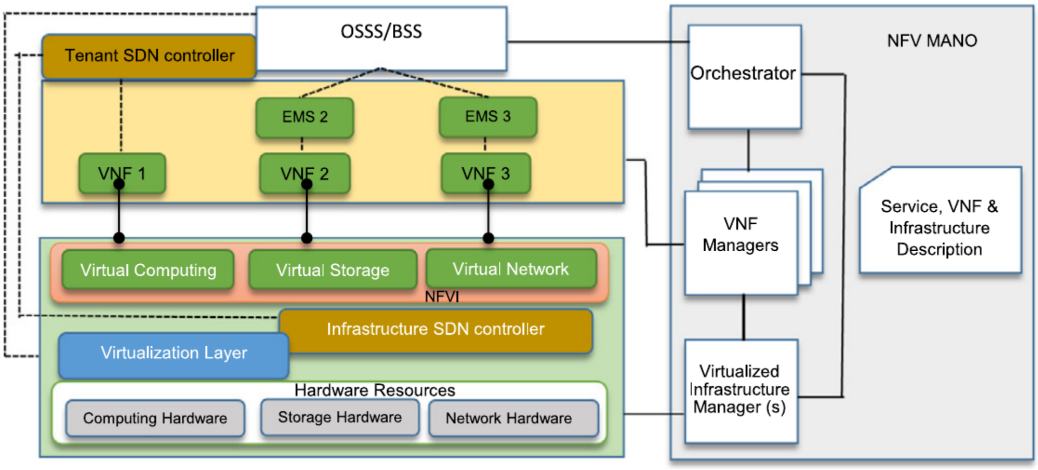


Figure 2.8: Integrated SDN and NFV MANO architecture [39, 78].

To date, multiple rollouts of SDN-based networks have been carried out. For instance, Google has adopted SDN-based solutions for the interconnection of its DCs with the objectives of increasing efficiency and reducing costs [63]. Besides, many are the ongoing efforts promoted by carriers, device manufacturers, InPs and SPs for the adoption of SDN through open standardization. For instance, Google, Facebook, Yahoo, Microsoft, Verizon, and Deutsche Telekom formed the Open Networking Foundation (ONF) [77], while Cisco, Ericsson, IBM, Juniper Networks, Microsoft, NEC, Red Hat and VMware launched the OpenDaylight initiative, hosted by the Linux Foundation [64]. Finally, ETSI proposed guidelines for the deployment of SDN networks within the NFV framework according to the integrated architecture represented in Fig. 2.8 [78], while the Broadband Forum (BBF) released recommendations for the support of SDN in broadband networks [79], including scenarios where only part of the network is equipped with SDN functionalities.

With respect to the forwarding devices, many commercial vendors included support for the OpenFlow API (e.g., HP [80], or NEC [81]). Besides, many software-based implementations have been proposed for Layer 2 and Layer 3 forwarding devices both with open (e.g., Open vSwitch [82] and ONF Stratum [83]) and proprietary implementations (e.g., Pica8 [84]). As the number of SDN-based devices increases, it is of great importance to develop multicontroller and multidomain standard interfaces. Therefore, configuration and communication compatibility has to be guaranteed among different vendors' devices [52, 54, 77, 85], in clear contrast with the proprietary and closed solutions typically adopted in conventional networks.

OpenFlow, managed by ONF [86], has become the de defacto standard for southbound APIs, yet, it is worth mentioning a few protocols and plugins that have been proposed as alternative or complementary solutions. NETCONF [87] is a configuration protocol based on Extensible Markup Language (XML) developed and standardized by the Internet Engineering Task Force (IETF), which allows the the installation, manipulation, and deletion of network devices' configurations. However, contrarily to OpenFlow, it cannot add new forwarding functionalities. The Open vSwitch Database (OVSDB) protocol [88] has been designed to provide advanced management functionalities in Open vSwitches, allowing the creation of multiple virtual switches instances within the same device, setting QoS policies on the interfaces, and managing queues. CISCO OpFlex [89] aims at enhancing the scalability by offloading part of the control capabilities back to the forwarding devices. Finally, the OpenDaylight project proposed Software Defined Network interfaces (SDNis), that is, east/westbound APIs developed as applications through the northbound API [90].

Standard northbound APIs are also fundamental for guaranteeing the portability and interoperability of network applications across heterogeneous control systems. Many efforts have been produced in this context, such as NOSIX [91], that introduced the abstraction of southbound APIs as device drivers for the application layer. In alternative, the PANE controller [92] and the ONF approach presented for Open Network Operating System (ONOS) [40,93] foresee the employment of the northbound APIs for the dynamic and granular control of network resouces [54,86]. Interesting are the approaches that develop the northbound APIs by exploiting SDN programming languages, such as, Frenetic [94]/Pyretic [60, 67], Procera [95], P4 [96]. Contrarily to the convergence described for the southbound interface, a unified northbound interface has not been identified yet. Indeed, as very diverse requirements can correspond to different applications, vertically oriented northbound APIs are likely to arise as market standards [47]. As a confirmation, existing SDN controllers often come with their own northbound APIs, which are mainly Java based, or deployed according to REpresentational State Transfer (REST) and REST Configuration Protocol (RESTCONF) approaches [40].

Many SDN controllers have been proposed in the past decades, each with different approaches in terms of distribution of the control logic, south/northbound APIs, device compabitibility and integration with the NFV framework [97, 98]. Some of the existing NOS are proprietary, such as, the NEC's Network Operation Engine

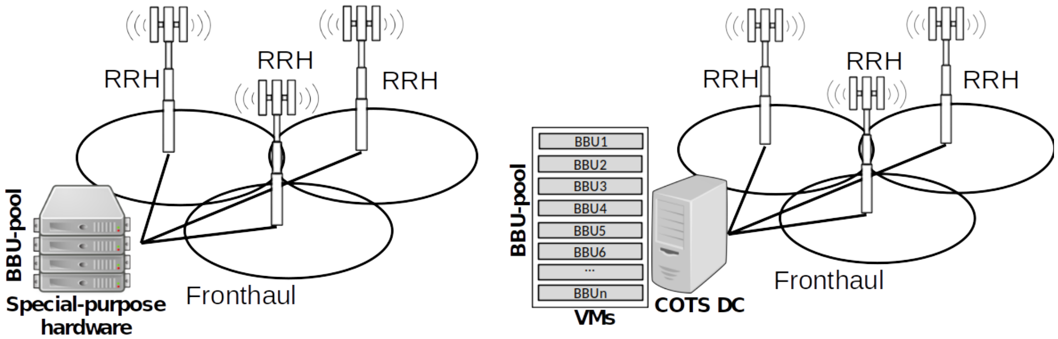
(NOE) [99], Juniper's Contrail Controller [100], and Nicira's NOX [50], which has been successively intergrated with VMware's Software-Defined Wide Area Network (SD-WAN) technology [101]. However, many other controllers have been proposed in the literature, such as, Onix [61] and Meridian [102], while some others have been introduced by the open community, such as, the Linux Foundation's OpenDayLight [64], the Big Switch Networks' FloodLight [103], and the ONF's ONOS [40, 93, 104] controllers.

Among the proposed solutions for SDN, some adopt a centralized control plane (e.g., Floodlight [103]), because of the high throughput enabled in this configuration by the high parallelism of cloud-based multicore computer architectures. Indeed, many SDN controllers target very specific niche markets, such as enterprise networks, cloud providers and telecommunication companies, with strict requirements in terms of throughtput (e.g., Meridian [102], Juniper's Contrail Controller [100], and VMware's SD-WAN controller [101]). On the other hand, Onix firstly introduced a distributed control plane with east/westbound APIs [61], followed by ONOS [40, 93] and OpenDayLight [64, 105]. This design approach often results in the support of use cases with more diverse requirements.

2.2 Programmability Extensions to Mobile Networks

As for DCs and transport networks, deploying virtualized and programmable mobile infrastructures is a fundamental requirement for the efficient provision of enhanced services in next-generation networks. In parallel to the advances in the NFV framework and in the SDN architecture, many solutions have been proposed for their integration in mobile networks. In this section, we present different building blocks for the enabling of programmability in mobile networks from different perspectives. On the other hand, we refer to the Chapter 3 for a holistic approach to network programmability in 5G networks.

The concept of centralization has been firstly introduced by China Mobile in 2010 [16], with the proposal of the Centralized-RAN architecture described in Figure 2.9a, where BSs are split into: i) a remote radio head (RRH) responsible for analog radio frequency functions, ii) a base-band unit (BBU) for digital base-band processing, and, iii) a fronthaul link for the connection of the two components. Therefore, the



(a) Centralized-RAN

(b) Cloud RAN

Figure 2.9: Centralized and Cloud-RAN.

main difference with SDN is the physical centralization of part of the data plane's functional blocks, rather than a logical centralization of the control plane. However, this approach still leads to well documented advantages that have been studied in large-scale trials [16]. Indeed, by co-locating the BBUs of multiple RRHs CAPEX and OPEX can be shared for multiple BSs, with a 41% power consumption reduction demonstrated in [106] thanks to shared cooling costs.

Once the intelligence of multiple network elements (NEs) is centralized, many extensions towards efficient network deployments and management are unlocked, as the global view on the network state can be exploited for enabling optimized and cheaper coordination among neighbouring BSs. For instance, mostly in dense HetNet deployments, enhanced interference management and coordinated multi-antenna/multi-point transmission technologies can be achieved with lower traffic loads on the backhaul, as the management entities of different NEs are co-located. Besides, the basic principles underpinning network programmability (cf. Section 2.1) can be explored for an optimal utilization of physical resources, as well as for simplifying the development of novel services and business opportunities.

Following an SDN-based approach, multiple NEs' control plane could be logically centralized, thus providing efficient, scalable and platform-agnostic mechanisms for, but not limited to, seamless mobility, load balancing, radio resource allocation to RRHs/users, QoS and access control policies [58, 107]. In particular, as depicted in Figure 2.10, the fronthaul network connecting BBU-pools to RRHs could be reliably implemented with SDN switches managed by a common controller, capable of defining which RRHs to support with a specific BBU-pool (i.e., defining routes, data rates,

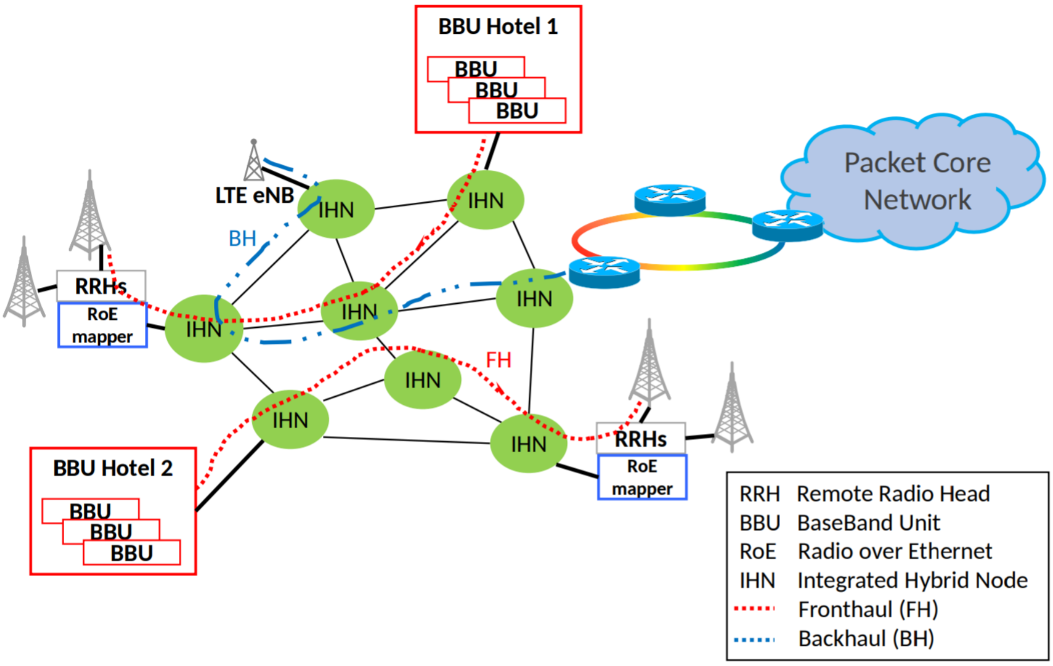


Figure 2.10: Fronthaul SDN implementation [111].

and delays among RRHs and associated BBU-pools), as well as optimizing inter-BBU communications (e.g., when coordination is needed among RRHs supported by different BBU-pools) [39, 108, 109]. Finally, the network abstraction enabled by SDN would allow the adoption of simplified topologies, as well as the implementation of multiple virtual mobile networks within the same physical infrastructure, which could be managed and customized through standardized APIs [39, 40]. We remark that an SDN-based strategy is crucial mostly in the case of dense HetNet deployments, as it could guarantee a simplified and scalable approach for counteracting, or at least mitigating, RANs' bottlenecks, while reducing OPEX and overheads [110].

Another important technological step enabled by centralized architectures in mobile networks is the employment of data-centric approaches for the development of network (data and control plane) functionalities. More in detail, the integration of the NFV framework in next-generation networks enables the concept of Cloud-RAN (C-RAN), according to which special-purpose and vendor-specific BBU-pools are substituted with COTS DCs, as represented in Figure 2.9b. [112,113]. Therefore, with respect to the Centralized-RAN architecture, base-band processing functionalities are implemented as VNFs, enabling additional efficiency and scalability by a suitable placement of the VNFs' instances within the network's DCs. For instance, the VNFs used for the coordination of multiple RRHs could be placed within the same DC (e.g., user association, resource allocation, coordinated transmission, or interference management).

In a C-RAN architecture, more flexible associations can also be implemented between BBUs and RRHs. More precisely, a specific RRH's BBU could be deployed over multiple DCs, besides, specific BBU's VNFs could be adapted as a function of the traffic load and even shared by multiple RRHs [114–116]. For instance, the overall network power consumption could be reduced by aggregating the BBU resources needed for serving a low-traffic region, thus minimizing the total number of DCs used [117,118]. We remark that adaptive strategies for the utilization of DC resources is particularly relevant in the case of C-RAN based HetNets (H-CRANs), as the higher OPEX required by denser deployments can be reduced by optimizing the utilization of virtual BBU-pools' resources [115,119]. In Chapter 4, we will demonstrate how the efficient utilization of BBU and spectrum resources can lead to beneficial sharing schemes among MNOs.

The deployment of data-centric technologies within telecommunication infrastructures leads to the edge computing paradigm, depicted in Figure 2.11, which foresees the deployment of small-scale cloud infrastructures for third-party service provision, also known as cloudlets, in the proximity of the end users (i.e., within the RAN or CN) [39,120]. Therefore, SPs can deploy data-centric services and delay-critical applications (e.g., localization services, enhanced video processing, augmented reality, IoT, e-health/telemedicine applications, etc.) at the edge of the network, with strict guarantees in terms of E2E latencies [39]. Besides, network operators can implement caching strategies for alleviating network congestion on the transport network by offloading popular contents closer to the end users. On the other hand, large-scale and centralized DCs will probably remain the reference technology for delay-tolerant, computing-intensive tasks, and for storing less popular content.

Within the edge computing paradigm, research community, manufacturers, mobile operators and standardization bodies proposed several alternatives for the convergence of IT and telecommunication services, among the others, mobile edge computing (MEC) [39,121] and fog computing [39,120,122]. One possible way of cathegorizing different edge computing solutions takes into consideration the distance of the cloud resources with respect to the end users, as summarized in Figure 2.12. More in detail,

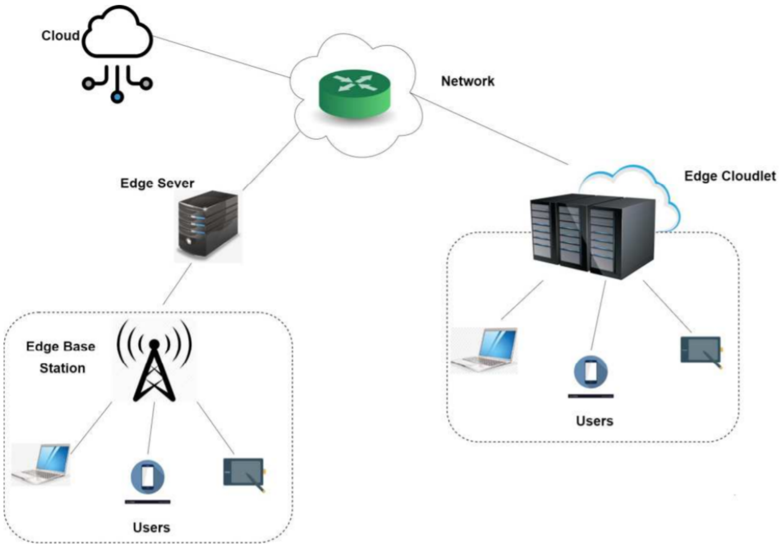


Figure 2.11: Edge Computing paradigm [120].

while fog computing is a general approach for bringing cloud computing functionalities to the lower layers in multiple domains' and industries' networks, MEC focuses on the deployment of cloudlets at the edge of the RAN in telecommunications networks. Particularly interesting is the Fog-RAN architecture introduced in [123], which flexibly integrates the edge computing paradigms with the C-RAN infrastructure, by allowing the sharing of cloudlets resources by both RAN functionalities and content-oriented SPs. An important feature enabled by the integration of network and cloud functionalities is the possibility for SPs to exploit radio state information provided by MNOs for the provision of innovative context/location-aware applications and services (e.g., augmented reality, IoT, etc.) [39, 124].

Independently from the exact placement of the cloudlets within the network, the edge computing and NFV paradigms can be easily integrated by deploying at the cloudlets' premises also the VNFs needed for the network functionalities [39,120,122]. According to this approach, a cooperative and distributed strategy can also be implemented for boosting communication and computation capacity by exploiting multiple cloudlets, as well as for reducing OPEX [39, 120, 124, 125]. Indeed, VNFs workflows can be reconfigured and scaled when necessary, in order to provide adequate services in specific regions. For instance, in case of overloaded regions, extra cloud resources could be provided to both NFs and resource-intensive applications through cloudlets in the proximities [123]. Besides, in case of services with strict time requirements, both VNFs and cloud resources necessary for service provision could be moved closer to the place where they are made use of [123].

In conclusion, most of the existing approaches to edge computing share the following objectives:

- The reduction of transmission costs and improvement of QoS by avoiding bottlenecks at the core network and at the Internet backbone (i.e., increasing capacity, reducing service response time and limiting network congestion).
- Providing third-party application developers with real-time network state awareness and E2E QoS provisioning capabilities.
- The extension of the value chain by attracting new vertical segments in the market ecosystem, thanks to the rapid deployment and testing enabled for innovative applications and services.

In Chapters 5 and 6, the concept of E2E QoS provision at the edge of the network, and the financial incentives stemming from alternative business models, will be exploited for deriving novel admission schemes. The timeliness and effectiveness of these schemes will be proved for the support of delay-critical applications, and for the optimization of InP's profits and SPs' competitiveness.

2.3 Summary

In this chapter, the basic principles, architectures and candidate technologies for network programmability are reviewed, with focus on next-generation services, sustainable network upgrades, and fast innovation. First, the adoption of a virtualization framework and the centralization of the control logic are identified as the building blocks for enabling programmable policies, custom QoS and dynamic resource sharing. Standardization efforts and rollouts are then presented, together with the extensions to mobile networks. In particular, the advantages in jointly deploying connectivity and third-party service functionalities at the edge of the network (possibly within the same data-centric infrastructure) are discussed in terms of network scalability and enhanced service provision.

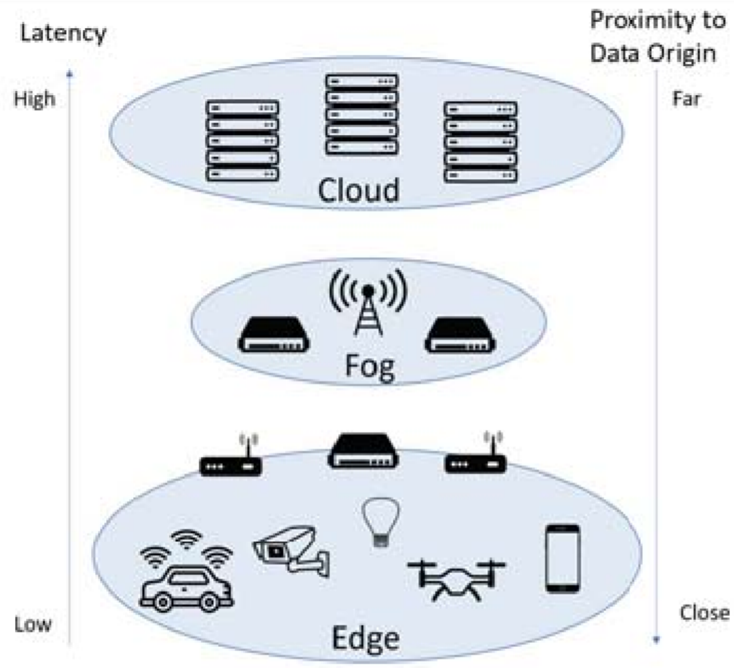


Figure 2.12: Cloud computing in mobile networks: existing strategies [122].

This chapter lays the basis for a clear understanding of the architectural and technological transition towards 5G, by introducing the necessary upgrades needed for achieving E2E network programmability and alternative revenue opportunities, as discussed in the following chapters. In Chapter 3, the solutions described in Sections 2.1.1, and 2.1.2, will be integrated in a novel architecture capable of guaranteeing fine-scale sharing and dynamic QoS, jointly over different segments of the network (i.e., RAN, CN, transport network and cloud infrastructure), within a unified framework for its monitoring and management. More in detail, the extensions for mobile networks presented in Section 2.2 will be enhanced in Chapter 3 and used in Chapter 4 for the study of beneficial sharing schemes among MNOs, exploiting efficient BBU and spectrum resource utilization. Finally, the same concepts will be adopted in Chapters 5 and 6 for flexible and timely admission schemes, capable of supporting diverse service types by third-parties, especially in case of limited resources and strict time constraints.

Chapter 3

Enhanced Architecture for E2E Fine-Scale Sharing in 5G

Overview

The ever increasing traffic demands foreseen for the evolution to 5G has stressed the need for increasing network capacity. As the network densification has almost reached its limits, mobile network operators are motivated to share their network infrastructure and the available resources through dynamic spectrum management. Although some efforts have been made in this direction by concluding sharing agreements at a coarse granularity (i.e., months or years), the 5G developments require fine timescale agreements, mainly enabled by network slicing. In this chapter, taking into account the radical changes proposed for next-generation networks, a thorough discussion is provided on the challenges that network slicing brings at different network segments, while introducing a new entity capable of managing slicing in the end-to-end.

Contributions

[J1] M. Vincenzi, A. Antonopoulos, E. Kartsakli, J. Vardakas, L. Alonso, and C. Verikoukis, "Multi-tenant slicing for spectrum management on the road to 5G," *IEEE Wireless Communications*, vol. 24, no. 5, pp. 118–125, 2017. (Area: Telecommunications; Quartile Q1; IF: 9,202).

3.1 Related Works

As introduced in Chapter 1, over the years MNOs seeked new business models for chasing new incomes, for reducing the costs of network ownership, and for optimizing the resource utilization. Below, we provide a review of the main technological and architectural upgrades introduced through different generations of pre-5G networks. Then, we introduce the slicing paradigm as candidate solution for the support for fine-scale sharing and QoS customization in 5G.

3.1.1 Active Sharing and Programmability in pre-5G Networks

Active RAN and CN sharing have been introduced in 3GPP Rel. 6 [126] for reducing OPEX by jointly scheduling active elements' resources (e.g., eNodeB antennas). Afterwards, 3GPP Rel. 12 [127] added a more complex sharing scheme, where MNOs act as InPs and sell their spare capacity to MVNOs looking for coverage and capacity extension. In this direction, the reference architecture is the one presented in 3GPP Rel. 14 [20] and illustrated in Fig. 3.1, where the master operators (MOPs) (i.e., the InPs and MNOs) share their RAN and/or CN with participating operators (POPs) (i.e., the MVNOs). MNOs can act as MOP or POP depending on whether they offer or seek coverage/capacity extension. Besides, each NE, such as eNodeBs (eNBs), home subscriber server (HSS), serving/packet data network (PDN) gateway (S/P-GW), mobility management entity (MME), and policy and charging rules function (PCRF), is associated with an element manager (EM), possibly co-located within the NEs premises. The overall network is then handled by a network manager (NM), which, through type 2 interface (Itf-N), provides end user functions for the management of single NEs or specific subnetworks, defined according to NEs' vendor, technology or employment in RAN/CN functionalities. Finally, the MOP network manager (MOP-NM) can open RAN/CN management functionalities to the POP network manager (POP-NM) through type 5 interface, with multi-vendor and multi-technology support. Both EMs and MOP-NM shall adopt self-organizing network (SON) functions for the automation of the sharing mechanisms.

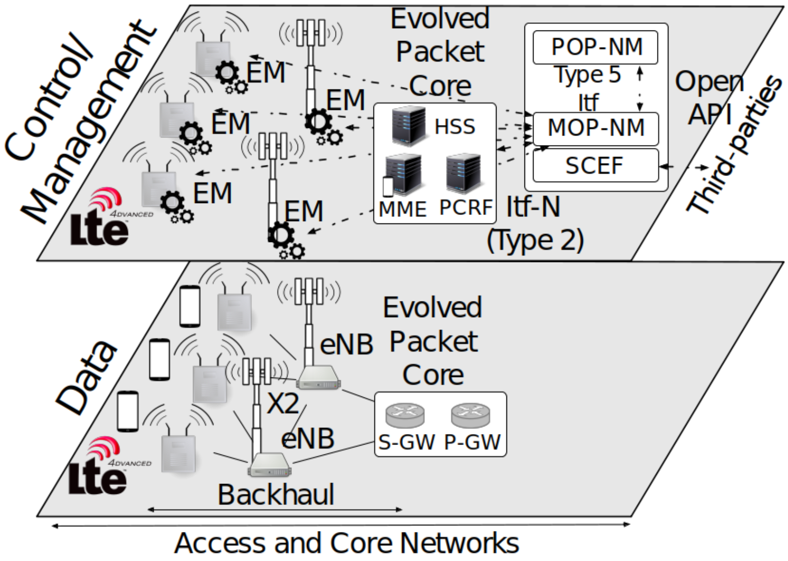


Figure 3.1: Standard Network Sharing Architecture.

Sharing among MOP and POPs is regulated by agreements on legal, financial, technical, and operational aspects, defining shared resources, rights and duties of each operator. These agreements normally stipulate long-term commitments, which represent a limitation, in terms of flexibility, for pre-5G network sharing mechanism. With respect to RAN sharing, an on-demand automated capacity brokering study has been proposed in 3GPP Rel. 13 [127] for scenarios like periodical capacity excess during night hours, or short-term extra capacity needs during special events (e.g., sports, concerts, fairs). Besides, according to 3GPP Rel. 14 [20], MOP shall optimize network resources while respecting the agreed shares of each POP, and shall be able to perform adequate pricing, by recording the resource usage of each POP compared to the planned one, differentiating between downlink and uplink, and among different QoS profile criteria.

3GPP foresees support for network programmability by securely opening network services and capabilities to third parties, under SLAs and with abstraction from underlying network interfaces and protocols. We remind that, according to the architecture proposed by 3GPP Rel. 14 [20] and illustrated in Figure 3.1, each NE is associated with a possibly co-located EM, and MOPs can manage the whole network through their MOP-NM, which provides management functions over different sub-networks. In LTE-A Rel. 14 [128], the MOP can open RAN/CN management functionalities to third parties through the Service Capability Exposure Function

(SCEF). More in detail, the interfaces among SCEF and the RAN/CN entities are defined within the trust domain of the MNO, while open APIs could be enabled towards the third-party OTTs/verticals. Examples of exposed services include network access authorization, traffic prioritization, charging policies and network statistics, among others. The PCRF is in charge of taking decisions on QoS-tailored service requests, following the standardized SCEF signaling flow for sessions set up.

3.1.2 Network Slicing towards 5G

Mobile communication systems preceding 5G lack in flexibility, providing no support for custom QoS provision and elastic network sharing, which in turn is expected to be the key enabler for a sustainable road to 5G. Since Rel. 14 [129], the possibility for SPs to dynamically provide customized services is explored by 3GPP, mainly differentiated in three well-known use cases:

1. LTE-like telecommunication services for MNOs, MVNOs, and OTTs
2. High capacity video/audio streaming for OTTs (i.e., eMBB)
3. Massive machine-type communications and delay-critical services for verticals and OTTs (i.e., mMTC and uRLLC)

Leveraging the solutions for network programmability introduced in Chapter 2, network slicing has been proposed for providing both fine-scale sharing mechanisms among InPs/MNOs and dynamic QoS to third parties (i.e., MVNOs, OTTs and verticals) [22] and has already attracted the attention of the main standardization bodies [39]. Through this paradigm, well-known cloud service models such as Software-as-a-Service (SaaS) and IaaS can be extended to the network level, thus enabling the Network-as-a-Service (NaaS) concept [22, 130, 131]. In particular, if SaaS and IaaS provide third parties with software licenses (e.g., for mailing and backup applications) and hardware resources (e.g., storage and computation resources), respectively, on a subscription basis and with full abstraction from the underlying systems [132], NaaS provides SPs with parallel sets of customized resources, that is, the network slices, which are dynamically isolated from the pool of network resources.

A layered representation of a possible slicing implementation in 5G networks is

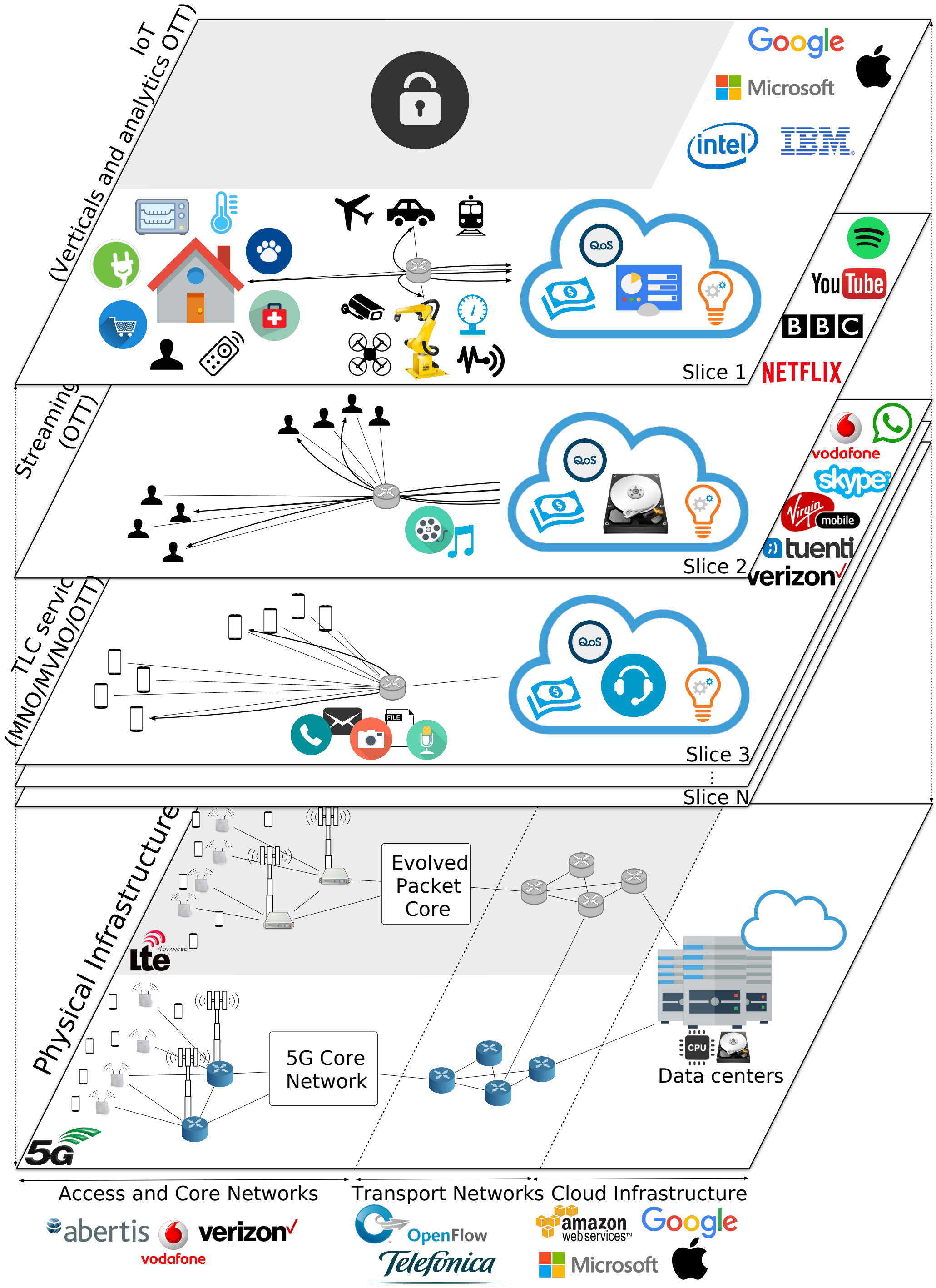


Figure 3.2: Flexible network slicing in 5G.

depicted in Figure 3.2, according to the use cases provided in Rel. 14 [129]. At the bottom, the physical infrastructure, which is split among isolated slices and is abstracted in each slice as:

1. Cloud computing resources, depicting the pool of DCs for service development
2. A virtual switch, symbolizing the E2E pool of communication resources (access, core, and transport network) used for service delivery

The LTE-A portion of the network is highlighted in light grey color, with respect to the overall 5G network, and its lack in flexibility and scalability is represented by locking the support for slicing over its infrastructure. Indeed, legacy networks are generally composed of special-purpose hardware, capable of implementing only specific functions. Consequently, computing, storing and communication resources cannot be flexibly customized, with no or limited support for QoS management. On the other hand, through slicing, MNOs can extend coverage and capacity in real-time, avoiding the traditional long-term agreements described in Chapter 1, which may not address the actual resource requirements of the network [10]. Besides, slices can include heterogeneous resources from the RAN, the transport network and the cloud infrastructure.

In order to enable this holistic vision of network programmability, E2E network management has to be performed, that is, jointly for different segments of the network (i.e., RAN, CN and, when needed, transport network and cloud infrastructure). In particular, for respecting both SPs requirements and 3GPP protocols constraints, the slicing of cloud DCs and SDN-based packet data networks (PDNs) has to be planned jointly to the orchestration of RAN resources [133, 134]. Indeed, sufficient VNFs have to be instantiated in centralized and/or edge DCs for third-party service provision, while performing RAN operations according to standards' constraints, such as the hybrid automatic repeat request (HARQ) timing. Besides, the appropriate connectivity among VNFs composing a specific service has to be guaranteed by properly dimensioning the transport PDN, in terms of bandwidth, topology, traffic, device CPU, and forwarding tables [47].

In conclusion, E2E slicing mechanisms need to be defined by integrating algorithms for RAN scheduling [36] and SDN flows' optimization (in transport and backbone

PDNs) [47] with centralized/distributed cloud resources orchestration (for RAN/CN NFVs and content-oriented services) [119]. On the other hand, in order to achieve sufficient incentives for the network upgrade towards next-generation networks, InPs need to: i) minimize the OPEX thanks to fine scale resource allocation [133], and ii) maximize the number of profit opportunities, that is, the admission of diverse service requests (i.e., with different QoS requirements), constrained by the resource availability [133, 135].

In 3GPP Rel. 15 [6], the RAN sharing requirements for 5G have been defined such that a maximum and minimum allocation can be statically reserved to each POP, over a specified period of time and/or region. On the other hand, if unplanned additional capacity is needed by a POP, available spare capacity shall be dynamically allocated. Besides, in the same release, 3GPP introduces an additional entity for the exposition of management functionalities to third-paties, that is, the Service Exposure and Enablement Support (SEES), besides, REST compliant APIs [136] are enabled towards third-party OTTs/verticals, such as OpenAPI [137]. In Rel. 16 and 18 [6, 138], 3GPP introduced the concept of dedicated network slices for efficient resource utilization and enhanced third-party user experience, besides, the requirements for slicing in 5G have been defined as:

- The on-demand slice creation, allocation, modification, and deletion with isolation guarantees
- The provision of suitable APIs to third parties for slice monitoring and management
- The elastic adaptation, within minimum and maximum limits, of the slice capacity
- The support for slice prioritization
- Multi-slice/multi-service support for a given user

In other words, a mechanism for fine-scale and flexible slicing orchestration is needed for slice customization according to requirements on functionality (e.g., priority, charging, security), performance (e.g., latency, data rates) or set of served user equipments (UEs) (e.g., Public Safety users, corporate customers). An interesting approach is presented in [139], where the *5G slice broker* is defined as an entity co-located with the MOP-NM and the SCEF/SEES. This broker provides management capabilities to third parties (through the Itf-N interface) and handles SLA negotiations through SCEF/SEES. The necessary interface enhancements for automated slicing management are also presented in [139], as well as a two-layer resource allocation strategy, such that the pool of resources is first split into different slices, followed by intra-slice resource optimization, according to the specific policies of slice tenants. Because RAN/CN network slicing has been mainly addressed in the literature, in the next section, we propose possible enhancements for enabling E2E slicing mechanisms in 5G for full QoS support and fine-scale sharing, while we refer to the Section 3.3 for the latest standardization efforts in this direction.

3.2 Proposed Enhancements for E2E Slicing in 5G

In this section, architectural enhancements are proposed for enabling a holistic approach to network programmability and sharing in 5G networks. In particular, the building blocks introduced in Section 2.2 are composed for enabling fine-scale multi-tenant slicing, flexible full-network sharing, and E2E QoS guarantees. First, a brief review is provided on the technological innovations required at the data plane for making network slicing a reality, then the enhanced control/management entities for flexible E2E network slicing are defined, and, finally, the Network Slice Auctioneer is introduced for E2E slicing bargaining and QoS support. The proposed architecture is compared with conventional approaches for highlighting the importance of network flexibility at all layers in order to enable E2E slicing support. In Fig. 3.3, data and control/management planes of legacy and proposed 5G architecture are illustrated (in grey light and white, respectively), where the network infrastructure is divided in three segments: RAN/CN, transport network, and cloud infrastructure.

3.2.1 Data Plane Virtualization

With regard to the data plane (at the bottom of Fig. 3.3), the 5G infrastructure should evolve by employing the most promising SoA technologies for network virtualization. More precisely, according to the H-CRAN architecture introduced in Section 2.2, eNBs are replaced with software defined RRHs in charge of analog radio

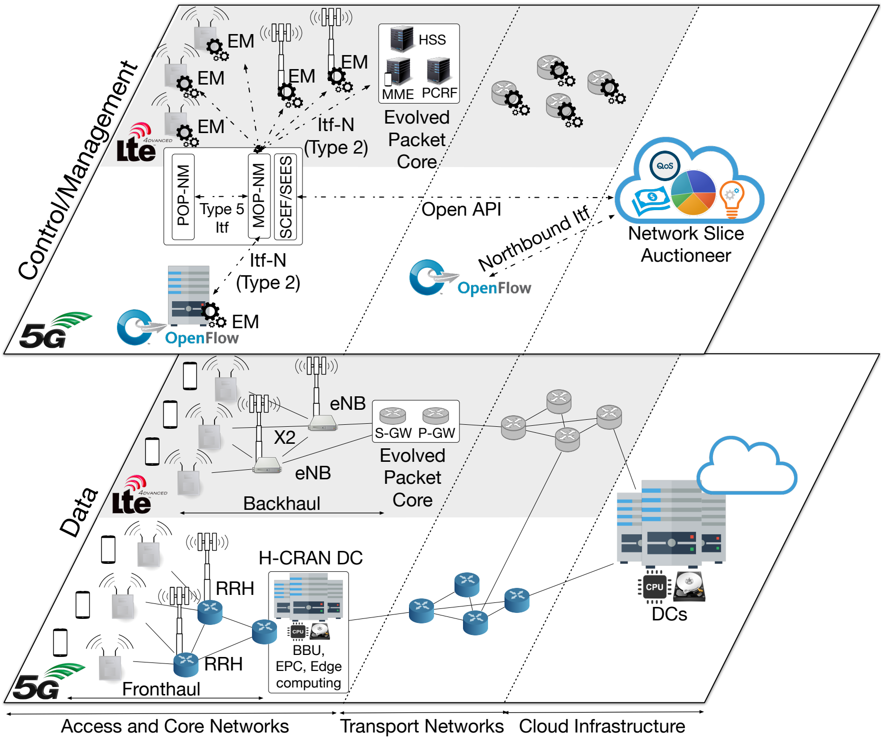


Figure 3.3: Enhanced network sharing architecture for E2E network slicing

frequency functions, while the BBUs in charge of digital baseband functionalities are centralized and deployed as virtual instances in a COTS small-scale DC [119]. The H-CRAN centralized architecture enables fast and enhanced network optimization (e.g., coordinated transmission functionalities, interference management, and energy efficiency) with considerable CAPEX/OPEX reduction [140]. The local DCs are connected among themselves, and to the set of available RRHs, through a software defined wired/wireless fronthaul (cf. Section 2.1.2), which substitutes or integrates the legacy backhaul.

The DCs deployed within the RAN/CN are exploited for implementing RAN/CN functionalities as well as for supporting the *edge computing* paradigm introduced in Section 2.2. The aforementioned elastic utilization of H-CRAN resources is enabled by the NFV paradigm presented in Section 2.1.1, which improves scalability by virtualizing and decomposing network services into a set of interoperating subfunctions, that is, the VNFs, which can be migrated and instantiated in different COTS platforms. Like fronthaul and backhaul, the legacy transport PDNs, generally created out of special-purpose and vendor-specific hardware, are substituted with programmable SDNs, introduced in Section 2.1.2, which interconnect different geographical areas and offer access to the cloud DCs. These aforementioned upgrades improve network flexibility and enable enhanced QoS provision, with significant impact on the value chain.

3.2.2 Control and Management Programmability

The top part of Fig. 3.3 shows the significant enhancement of the control/management plane achievable by adopting the programmability principles defined in Section 2.1.3. Thanks to the H-CRAN architecture, multiple standalone RAN/CN EMs can be centralized and possibly co-located with the MOP-NM, to which they are interconnected through software defined logical interfaces. In addition, the virtualization paradigm enables the flexible orchestration of the control/management entities in the form of VNFs, in such a way that prompt control/management operations can be performed by appropriately migrating the correspondent VNFs. For instance, VNFs with strict time requirements can be instantiated close to where they are required, and multiple VNFs with high interconnectivity demands can be co-located in the same DC.

In the proposed architecture, MOP-NM, SCEF/SEES, and slice broker are co-located, as in [139], since this approach offers enormous architectural advantages. More specifically, the slice broker can easily negotiate SLA requests and expose network control capabilities to third-party providers through the SCEF/SEES interfaces, as well as it can gain direct access to the RAN/CN monitoring and configuration through the MNO-MN. Moreover, encouraged by the network virtualization technologies proposed for next-generation infrastructures, the MOP-NM, the SCEF/SEES, and the 5G slice broker should be integrated in software for faster negotiation and management of network slices. The automated allocation of network resources through the described slicing architecture enables the appropriate programmability degree needed for flexible network adaptation to different services with diverse requirements. Moreover, on-demand slice orchestration is expected to take place at fine timescales, in such a way that resource usage is optimized with small granularity while competing third parties can get sufficient NaaS opportunities.

Besides the great benefits for third parties, the integration of MOP-NM management functions with 5G slicing orchestration also enables a new paradigm for flexible multi-tenancy among MNOs/InPs and MVNOs. Indeed, the pre-5G long-term contractual mechanisms for RAN/CN sharing could be substituted by automated real-time slicing, where MVNOs without network infrastructure can reduce CAPEX/OPEX by avoiding the deployment of the POP-NM and negotiate slices as general third-party service providers. Likewise, according to 3GPP Rel. 14 [20], multiple MNOs seeking coverage/capacity extension can pool their networks into a joint-venture InP, exploiting slicing orchestration for flexible DSM and infrastructure sharing. MNOs' shares could be translated into a minimum reserved slice allocation, while on-demand additional capacity can be provided through dynamic slice adjustment. This new paradigm represents a great opportunity in terms of:

1. Cost reduction thanks to infrastructure simplification
2. Flexibility improvement thanks to fine-scale slicing

while letting operators mantain the control over the underlying network. This evolution in mobile networks is in line with the concept of economy of scale enabled by the business model introduced in Chapter 1, where a few big market players are specialized in the management of the infrastructure and spectrum (i.e., the InPs)[1] and rent capacity to the rest of the stakeholders, which, in turn, focus on service provisioning (i.e., third parties).

As far as fronthaul, backhaul, and transport networks are concerned, the SDN architecture adds scalability and programmability thanks to the available open standard controllers and interfaces, which enable efficient VNF management as well as network-aware applications. Indeed, the SDN controller could be integrated with the MOP-NM for flexible interconnection among VNFs, such as for the dynamic mapping of virtual RRHs and BBUs over the fronthaul. Furthermore, similarly to SCEF/SEES, the SDN controller provides third parties with dynamic network management support, by allowing them to flexibly program the control plane through an open northbound interface. Therefore, third parties can define network slices through the SDN controller, by isolating the required bandwidth on the network links and by properly configuring the forwarding tables in switches and routers [54]. Besides, the proposed architecture introduces further enhancements in terms of CAPEX/OPEX reduction. In particular, according to Fig. 3.3, multiple controllers

[1] For instance, the Spanish operator Telefónica has created a subsidiary company for the management of its infrastructure on a global scale, aiming at improving the ROI through third-party service provision [141].

placed at the forwarding devices' locations are substituted with a central controller for scalable network configuration. Nevertheless, the network programmability features provided by the independent adoption of a slice broker for the RAN/CN, and an SDN controller for the fronthaul, backhaul, and transport networks, are not sufficient to ensure full E2E QoS support.

3.2.3 E2E Network Slice Auctioneer

In the 5G market ecosystem introduced in Chapter 1, third parties set more stringent requirements over underlying networks and demand a more active role in E2E network customization. Albeit network management and QoS prioritization mechanisms are partly supported by conventional networks, or foreseen for next-generation networks, they are traditionally deployed according to proprietary policies and mostly limited to specific network infrastructure (i.e., RAN/CN, transport network or cloud infrastructure), which leave third parties with limited or no control over full network optimization strategies.

According to the architectural enhancements discussed in this section, third parties can negotiate network slices by direct communication with the 5G slicing broker for RAN/CN slicing and with the SDN controller for the transport network slicing. On the other hand, a unique framework for the automated orchestration of E2E network slices might be sought by third parties, especially by those that are interested in a high-level monitoring and control over the network under the agreed SLAs, and willing to leave low-level optimization (e.g., resource allocation) to network owners. To this end, we propose the introduction of a novel entity into the management plane, that is, the E2E *Network Slice Auctioneer*.

As described in Figure 3.3, we envisage the deployment of the Network Slice Auctioneer as a third-party application running in a cloud infrastructure, which behaves as an intermediary between the InPs (i.e., the owners of RAN/CN, transport networks and edge/centralized cloud infrastructure) and the third parties (e.g., MVNOs or OTT/vertical SPs), and provides the following services:

- Receiving third parties service requirements
- Bargaining SLAs with InPs (i.e., MNOs, transport network owners and cloud

infrastructure providers) on behalf of the third parties, which compete for the allocation of appropriate E2E network slices

- Monitoring the allocated slices through the open APIs provided by the network programmability mechanisms defined in Sections 2.1.2 and 2.1.3
- Dynamic adaptation of the network slices to varying service requirements, fluctuating network conditions (e.g., wireless channel state or traffic load) and resource availability, with the level of abstraction desired by third parties
- Billing according to each InP's SLAs, while charging third parties with for the intermediation
- Managing the E2E slice lifecycle (e.g., releasing the corresponding resources when slices are no longer required by third parties)

Similarly to the controllers and orchestrators studied in Sections 2.1.2 and 2.1.3, the E2E Network Slice Auctioneer is a tool that exploits the features of network programmability for simplifying and optimizing network design and management, as well as for fostering innovation. However, the main difference lays in its unified approach to E2E network abstraction and optimization. Indeed, through its integrated interfaces with multiple controllers, domains and technologies, third parties can access dedicated portions of both network and cloud infrastructures, designed for providing E2E QoS guarantees. Besides, state information on RAN/CN, transport network and cloud infrastructure can be monitored by third parties through a unique platform.

As for SDN controllers (cf. Section 2.1.2), depending on the scalability and performance requirements of third parties, different strategies can be adopted for the implementation of the E2E Network Slice Auctioneer. Indeed, for services with strict time constraints or fast-fluctuating requirements, it could be implemented in the edge computing infrastructure and act as intermediary between the 5G slice broker [139] and the edge cloud InP, thus, providing a slicing-based extension of the Fog-RAN architecture presented in Section 2.2. On the other hand, when service requirements undergo slow variations and in case of, but not limited to, delay-tolerant and computing-intensive services, a centralized approach could be adopted by deploying the E2E Network Slice Auctioneer in the central cloud infrastructure, and adding the intermediation with the transport networks' InPs. Therefore, those telecommunication InPs that integrate cloud computing capabilities in their RAN/CN could also deploy a proprietary E2E

Network Slice Auctioneer in their edge computing infrastructure. Alternatively, according to the new business model introduced in Chapter 1, new value chains could be explored by SPs that decide to specialize in the deployment of (centralized/distributed) E2E Network Slice Auctioneer services. Finally, cross-sectorial industry organizatons could provide guidelines for the open standardization of this novel entity.

In Table 3.1, the main management entities described in this chapter are summarized, describing their responsibilities and challenges within the flexible network slicing paradigm described in Section 3.1.2. Most of the challenges are related with the joint fulfillment of RAN/CN constraints and third-party service requirements, when network virtualization technologies are employed. Indeed, although sub-millisecond service latencies have been already demonstrated by industrial testbeds, there are still many ongoing efforts on this topic.

Table 3.1: Orchestrators and challenges for E2E slicing in 5G

Orchestrator	**Responsibilities**	**Multi-tenant slicing challenges**
SON MOP-NM	• Self-configuration • Self-optimization • Self-healing	• Automated setup of shared RAN/CN (e.g., cell identity/discovery, base station configuration) • Joint dynamic optimization of network resources (e.g., coordinated transmission, interference management, virtual BBU optimization in C-RAN DCs) • Automated network backup through redundant infrastructure
SCEF/ SEES	• SLA intermediary • Network functionalities exposure	Security support through: • Slice isolation for third parties protection • Functionality access authorization for MNOs/InPs safeguard

5G slice broker	• Flexible RAN/CN slicing according to SLAs • Two level slicing: high priority to 3GPP functionalities, low to third parties • VNFs allocation and mobility management	Guarantee of 3GPP time constraints (e.g., HARQ) and service level sub-millisecond E2E delays through: • Cooperative distributed computing among multiple H-CRAN DCs • Dynamic VNFs migration among DCs • Dynamic management of the RRH/BBU split of functionalities, depending on the fronthaul technology and the real-time support for COTS platforms
SDN controller	Dynamic slicing of fronthaul, backhaul, and transport PDNs	VNFs interconnection according to: • Slice topology • Service requirements
E2E Network Slice Auctioneer	Dynamic real-time E2E resource bargaining, while: • Optimizing performance • Reducing costs	Joint dynamic planning and negotiation of network VNFs, for instance: • Boost of delay tolerant RAN/CN functionalities by adding extra VNFs at the cloud infrastructure • Delay-critical services can be moved to the H-CRAN DC • VNFs can be possibly shared among more NEs for cost minimization

For instance, [142] proposed H-CRAN architecture and protocol modifications for reducing latencies, while 3GPP Rel. 16 [143] introduced the shortening of the hardware processing time and the provision of reduced transmission time intervals.

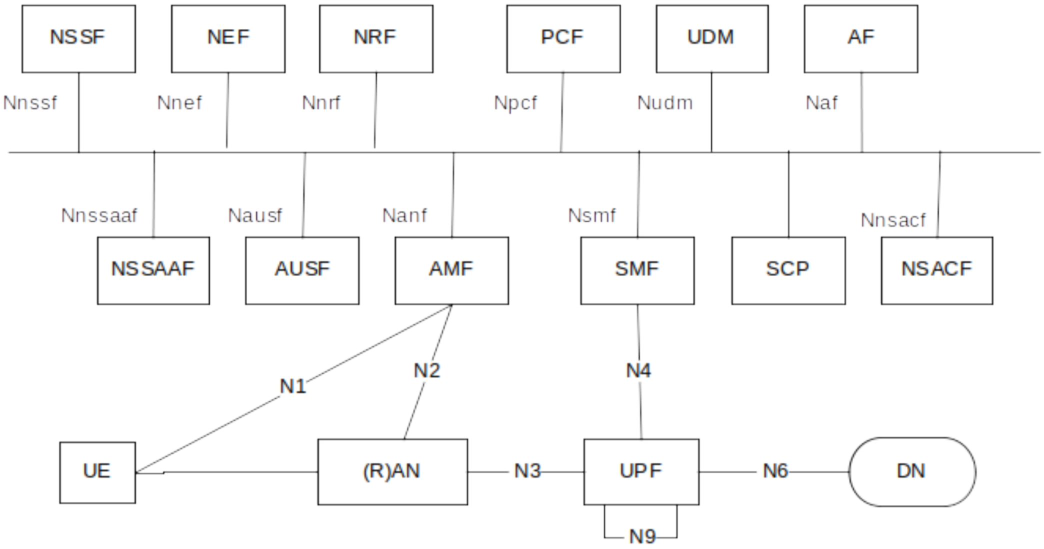


Figure 3.4: 5G System architecture (3GPP Rel. 17) [138].

3.3 Recent Advances in Network Slicing for 5G

In this section, in order to provide a complete picture on E2E network slicing, we present the most recent advances in network slicing for 5G, with focus on standardization for E2E programmability and slice orchestration, which we briefly compare with the approach proposed in this chapter. We refer to [39, 40, 144] for a thorough discussion on recent standardization, technologies and methodologies for E2E network slicing.

Inspired by SDN, the separation of control plane (CP) and user plane (UP) have been standardized by 3GPP Rel. 17 [138], in order to enable scalability, flexibility and independent deployment (e.g., according to a centralized or distributed strategy). In addition, the minimization of dependencies between RAN and CN is promoted and support for NFV and network slicing is enabled [138]. In particular, in RAN, multiple logical nodes can be mapped over a single physical NE [145], while in CN NFs can be deployed according to a distributed and scalable strategy [138]. Besides, the modularity and reutilization in NF design and interconnection are promoted, while, depending on service requirements on the maximum latency, UP functions can be flexibly deployed in a central location or at the edge of the network [138].

In Figure 3.4, the standard service based architecture (SBA) provided by 3GPP Rel.

17 is depicted, with clearly separated UP (i.e., UE, RAN, user plane function and data nework) and CP (i.e., the rest of the NFs) NFs, and exhibited service-based interfaces. A brief description of the main NFs is provided below:

- **User Plane Function (UPF):** it is a centralized entity that routes traffic between NFs and applications. Its location and configuration can be flexibly adjusted to the type of service required
- **Data Network (DN):** it represents operator services, third-party services, or the Internet access, which, in 4G, were typically accessed through the P-GW
- **Access and Mobility Management Function (AMF):** it is in charge of access control and mobility management, similarly to the MME in 4G. It also helps in interconnecting other NFs
- **Session Management Function (SMF):** it is employed for creating and managing sessions according to pre-defined policies (e.g., IP address selection and allocation, configuration of UP traffic rules, or support to roaming), similarly to the MME in 4G
- **Policy Control Function (PCF):** it provides policy rules to govern the nework behaviour, by integrating mobility management, network slicing, and roaming. Together with the CHarging Function (CHF), it substitutes 4G's PCRF
- **Unified Data Management (UDM):** it is used for storing subscribers' data and profiles, similarly to HSS in 4G
- **Authentication Server Function (AUSF):** it is employed for authentication purposes, similarly to HSS in 4G
- **Network Resource Function (NRF):** it is a novel entity that allows NFs functionality discovery and intercommunication via APIs, besides, it keeps track of all NF instances' profile
- **Network Exposure Function (NEF):** it is a centralized entity in charge of the exposition of services and capabilities offered by CN's NFs to third-parties (see Figure 3.5), similarly to SCEF/SEES in previous releases, to MANO in ETSI NFV architecture 2.1.3, and to the NOS in an SDN architecture 2.1.2

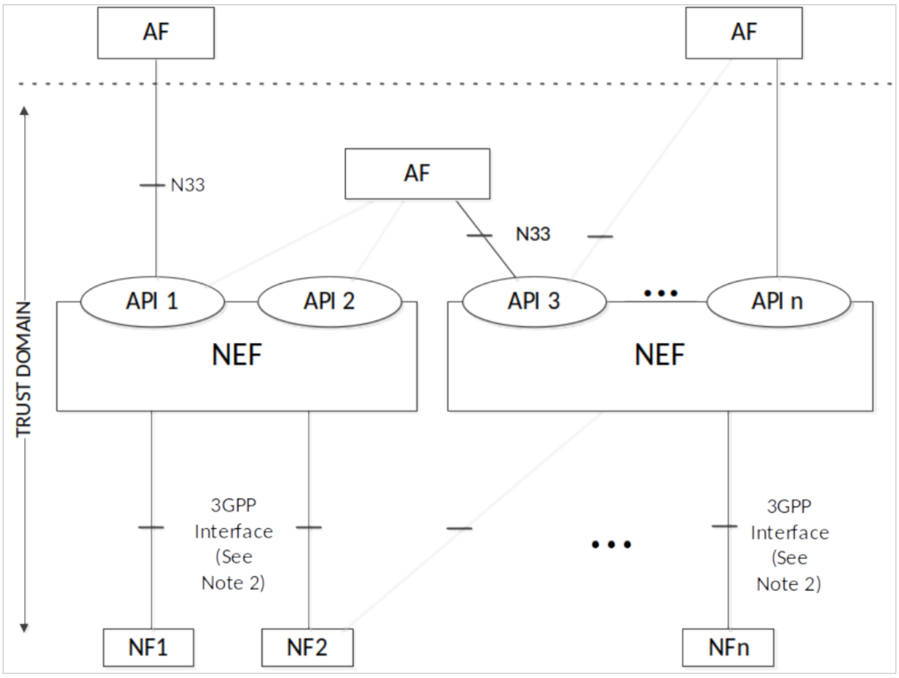


Figure 3.5: NEF architectrue and AFs' trust domain (3GPP Rel. 17) [138].

- **Application Function (AF):** it represents applications deployed on top of the 5G infrastructure, which can control the traffic routing, access the NEF by means of APIs, interact with PCF. In case of AFs trusted by the operator, they can directly interact with the NFs within the trust domain

- **Network Slice Selection Function (NSSF):** it provides assistance in the allocation of AMFs to specific users based on service requirements, and in the selection of suitable network slice instances (NSIs), that is, a set of NF instances and required resources (e.g., compute, storage and networking resources) forming a network slice

- **Network Slice Admission Control Function (NSACF):** it supports the monitoring and control of the number of UEs and data sessions per network slice, together with event-based notifications and reports on slices' status (e.g., to AFs). In case of reaching the maximum number of UEs, it enforces admission control policies

In conclusion, latest 3GPP releases support many concepts introduced in Chapter 2 for network programmability, such that, the decoupling of data and control planes, the logical centralization of the control functions, the flexible deployment of NFs as software instances, and according to different levels of distribution within the nework, the exposition of NFs to third-parties. Besides, third-parties are also provided with context awareness and notification services on the network state. Finally, 3GPP provides support for slice paradigm and for enhanced applications (e.g., delay-critical applications).

While the overall architecture is well defined, many open issues remain unsolved with respect to many operation and optimization aspects. Open challenges are mainly related with enhancements in flexibility, scalability and portability by adopting full network virtualization, as well as with the efficient E2E resource orchestration with isolation guarantees [39, 40, 144]. Consequently, many efforts by standardization bodies, associations, alliances, as well as projects and PoCs by both industry and academia are still ongoing. We refer to [39, 40, 144] for a detailed and comprehensive review. We remark that, similarly to the E2E Network Slice Auctioneer introduced in this chapter, many of the existing solutions converge in the vision of a unified orchestrator capable of managing multiple domains and technologies by coordinating multiple controllers along the service chain (i.e., for RAN/CN, transport networks and cloud infrastructures), thus enhabling E2E network abstraction, programmability and multi-tenancy.

3.4 Summary

In this chapter, some light has been shed on the challenges that multi-tenancy and network slicing bring in the next generation of mobile networks, introducing the SoA technologies and the new entities required for flexible network management. An enhanced 5G architecture for flexible network sharing and QoS guarantees has been proposed and compared with the legacy one for highlighting the importance of network programmability at all layers in order to enable E2E slicing support. The main enhancement consists in the adoption of centralized architecture, and virtualization technologies for implementing control/management entities as co-located software instances. In this way, a more flexible negotiation of SLAs is achieved, guaranteeing both flexible network sharing among InPs/MNOs, and a more efficient exposure of network control capabilities to third-party SPs. A comparison of the proposed architecture with recent standardization efforts and SoA is also provided.

Chapter 4

Efficient Sharing among MNOs

Overview

Facing increasing energy and traffic demands associated with 5G wireless networks, MNOs are motivated to gradually convert conventional RAN infrastructures into more flexible and efficient architectures, that is, dense HetNets with centralized and programmable architectures (i.e., H-CRAN). Beside the promising benefits enabled by programmable networks in terms of stand-alone network optimization, fine-scale sharing of resources and network elements is also encouraged among operators. In particular, a better utilization of the frequency resources can be achieved by MNOs, leading to enhanced coverage and data rates. On the other hand, as BS are the most expensive component of conventional RAN, it is fundamental to jointly optimize the energy-efficiency of the BBU-pool. In this chapter, a novel scheme based on coalitional game theory is introduced for identifying the potential margin for performance and profit gains provided by flexible network sharing and joint DSM among MNOs. The proposed scheme is capable of determining whether a spectrum and infrastructure sharing strategy is preferable to the stand-alone case, and what would be the pricing scheme to be adopted by MNOs, together with the financial benefits. Results in terms of QoS and financial gains are provided for a sample scenario with three operators, with different market and spectrum shares. More precisely, this work shows: i) cooperation among sub-coalitions of MNOs is always beneficial, yielding both higher revenues and enhanced QoS for the end users, and ii) the cooperation of all operators can be preferred to smaller coalitions for specific user pricing in different scenarios.

Contributions

[C1] M. Vincenzi, A. Antonopoulos, E. Kartsakli, J. Vardakas, L. Alonso, and C. Verikoukis, "Cooperation incentives for multi-operator C-RAN energy efficient sharing," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6.

[J1] M. Vincenzi, A. Antonopoulos, E. Kartsakli, J. Vardakas, L. Alonso, and C. Verikoukis, "Multi-tenant slicing for spectrum management on the road to 5G," *IEEE Wireless Communications*, vol. 24, no. 5, pp. 118–125, 2017. (Area: Telecommunications; Quartile Q1; IF: 9,202).

4.1 Related Works

As explained in Chapter 1, BSs are the most expensive component of conventional RAN, besides, according to [146], base-band processing is the most power-consuming element of SCs, due to the high processing complexity required compared to the low power transmitted. Thus, in order for 5G networks to be sustainable, it is fundamental to optimize the energy-efficiency of the SC layer.

Many works addressed power consumption minimization for conventional RAN, mainly leveraging BS switching-off concepts [147, 148]. The main drawback of switching-off in conventional RAN are the possible coverage holes generated, because of the typically disjoint BSs' service areas.

An alternative approach is the one represented by the C-RAN centralized and programmable architecture introduced in Section 2.2 and proposed for an efficient usage of RAN resources. Indeed, the BBU-pool, that is, the pool of the processing resources used for VNF instances with BBU functionalities, can be jointly and dynamically allocated to different RRHs, based on current network state.

Many papers have addressed the NP-hard problem of efficiently mapping BBUs and RRHs while seeking objectives, such as, the minimization of the power consumption. Among the possible strategies, [115] models the problem as a bin-packing problem,

[116] proposes graph coloring, [149] formulates a Knapsack problem, while [150] uses a listing algorithm.

Dealing with the volume of traffic forecasted for 5G networks, we discussed in Chapter 1 on how network densification and the offloading of mobile data traffic to SCs is one of the candidate strategies for tackling the scarcity of spectrum resources. However, it is a common opinion in the literature that the UEs offloading to the SC layer by itself will not be sufficient, and that bandwidth extension in the mmWave frequencies will be necessary [9]. On the other hand, many works provide strategies for improving the spectrum utilization by means of opportunistic (e.g., Cognitive Radio) [151, 152] or cooperative spectrum sharing [153], that is, where users without guaranteed access to the licensed spectrum (i.e., the secondary users) detect and dynamically utilize spectrum opportunities, withouth affecting the communications of the users with assigned portions of the licensed spectrum (i.e., the primary users). The main difference between opportunistic and cooperative spectrum sharing is that, in the first case, secondary users access the spectrum with minimum coordination with primary users, while, in the second case, secondary users also help in delivering primary users' traffic in exchange for a shared portion of the spectrum.

Focusing in approaches for improving resource utilization efficiency, many papers have addressed the problem for the case of conventional RAN. In particular, [154] studied the benefits of jointly deploying a new shared network, while [147] investigated the advantages offered by base stations switching-off.

Works concerned with C-RAN are mainly focused on the BBU optimization in the single operator case [115,149,150]. However, in a cooperative approach, fine-scale joint optimization of the BBU resources could better exploit the statistical multiplexing gain over shared networks [117, 118]. Indeed, as traffic typically undergoes dynamic fluctuations in time and space domain, consisting of the of many users' independent flows, which can be modelled as a random process. Because different BS are exposed to different traffic loads in time, depending on their locations, they will typically experience load peaks at different times. Therefore, the overall traffic load over a network can be regarded as a random fluctuation in both time and space, whose cumulative peak at a specific time instant is smaller than the sum of each BS's load peaks over time. Therefore, since independent deployment of BBU resources for single BSs requires a design dimensioned according to the peak traffic load of each base

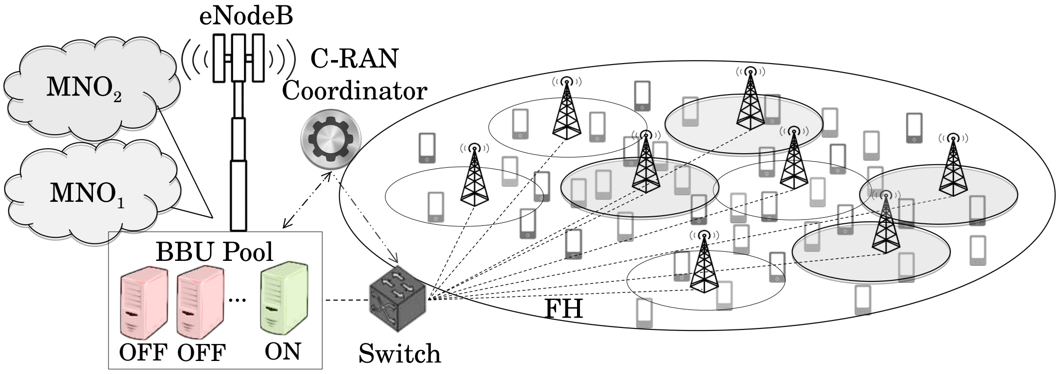


Figure 4.1: Scenario for spectrum and infrastructure sharing among MNOs with H-CRAN architecture.

station, the joint deployment and management of BBU resources for multiple BSs allows for potential cost savings. Hence, supported by the flexible architecture and mechanisms introduced in Chapter 2 for network sharing at a fine-scale, joint BBU-pool optimization among multiple cooperating MNOs enables power savings and QoS improvements. [116] provides a framework for flexible BBU resources orchestration for the case of two WiMAX operators sharing the BBU-pool. However, the possible enhancements in QoS (i.e., coverage and data rate) offered by spectrum sharing are not investigated.

4.2 Game theoretical Approach for Network Sharing

In this chapter, a cooperative game approach is used for the assessment of MNOs incentives for running a shared H-CRAN (cf. Section 2.2). Given the scenario depicted in Figure 4.1, where a set of MNOs coexist in the same area, each operator may decide to keep running its network independently or to form a joint-venture InP. More in detail, as we are interested in studying the potential in QoS enhancement and efficient resource utilization offered by network densification described in Section 4.1, we focus in this study only on the sharing of the SC layer, composed by RRHs, BBU servers, FH and spectrum. Therefore, according to the nomenclature introduced in Section 3.1, in this cooperative case, MNOs agree on pooling their SC layers into a new MOP, while each member of the coalition becomes a POP with equal priority in using the shared infrastructure and resources.

By cooperating, operators can achieve higher data rates and extended coverage for their UEs, which have the potential to be translated into increased revenues. On the other hand, by sharing the costs of a larger network, it is expected that forming a coalition is profitable only under given conditions, depending on the particular market and spectrum share of the cooperating MNOs. To this end, the problem is formulated as a coalitional game where MNOs are the players, and these conditions and their physical meaning are investigated.

In this context, joint network optimization and sharing at a fine-scale is fundamental and though not allowed by pre-5G networks, which only offer sharing agreements at coarse granularities. Therefore, the programmable infrastructure and the enhanced architecture introduced in Chapter 3 are adopted, which enable joint switching-off of the BBU resources at a fine-scale, and which can flexibly sustain multi-tenant traffic in a transparent way. In particular, applying SDN concepts to the RAN, BBU servers can be pooled by deploying a common switch and managed by a centralized and shared coordinator, that is, the MOP-NM. The latter is responsible for monitoring and reporting RRHs' state information and for performing: i) UE association, ii) dynamic resource allocation and performance optimization with respect to MNOs' objectives and traffic profiles, and, iii) power consumption reduction through joint BBU optimization. Finally, in order to effectively pool spectrum at each SC premises, RRHs need to support the CA paradigm, in such a way that the whole set of carriers aggregated from, and shared by, multiple MNOs can be accessed by UEs compliant with LTE-A (or beyond).

4.3 System Model for H-CRAN

A set $\mathcal{M}$ of mobile MNOs that deployed their own 4G HetNet in a given area $\mathcal{A}$ is defined. According to Figure 4.2, for each MNO m the HetNet consists of a typical RAN MC layer and a H-CRAN SC layer. Because in this chapter we focus only on the sharing of the SC layers, as explained in Section 4.2, in the follwing we avoid modeling MC's resources and infrasture. Besides, we assume that each MNO owns an exclusive and independent spectrum license for MC and SC layers, therefore, interference is generated only by equipments belonging to the same layer of the same MNO. In particular, according to the efficient usage of the spectrum resource enabled by HetNets (cf. Chapter 1), we consider a unitary frequency reuse factor for SCs

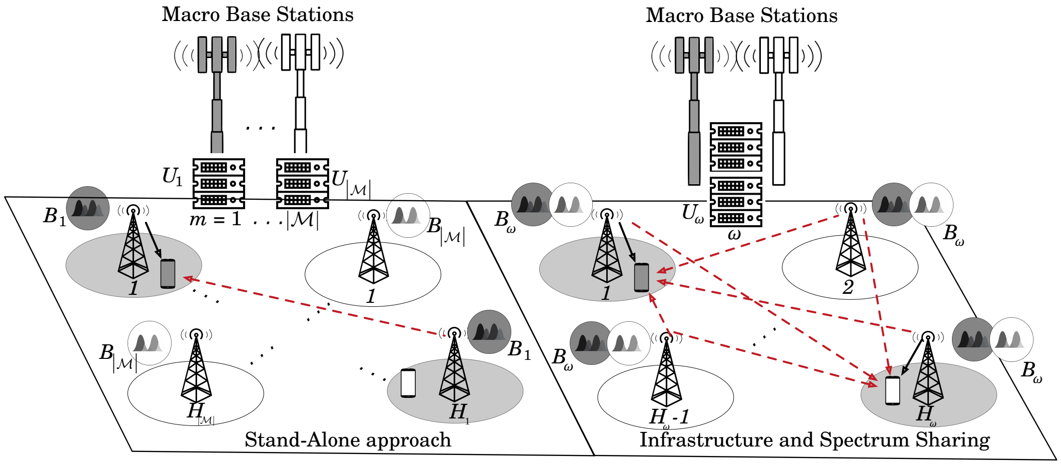


Figure 4.2: System model: coexisting MNOs follow a stand-alone approach, or share their infrastructure and spectrum forming a coalition ω. Interference is represented by the dashed lines.

over a licensed band of B_m MHz for MNO m.

Each operator m has deployed H_m SC RRHs uniformly distributed, which, according to Fig. 4.1, are connected through a FH link to U_m COTS servers used for implementing VNF instances with BBU functionalities. BBU servers are grouped in a centralized physical site (i.e., the BBU-pool). It is assumed that the BBU-pool is co-located with the MC eNodeB and connected to the CN through the eNodeB. Besides, one can consider that MNOs share eNodeB and BBU-pool site.

In the same area, it is assumed the presence of N_{UE} UEs uniformly distributed and with activity factor f_a, which represents the probability for a UE to be active at a given time [146]. Each MNO m owns an exclusive portion of the total subscribers, that is, the market share μ_m over the total number of UEs. It is assumed best Signal-to-Noise Ratio (SNR) association, that is, UEs associate to the BS (i.e., eNodeB or SC) with the highest received power above sensitivity SNR_{min}.

At the end of the association process, on average, a portion of the user equipments will associate to the eNodeB while a given percentage $\mathcal{O}_m^{\text{SC}}$ will be offloaded to the SC layer (i.e., $\mathcal{O}_m^{\text{SC}}\mu_m N_{\text{UE}}$ represents the average number of subscribers offloaded to the SC layer by MNO m). Also it is assumed proportional-fairness as scheduling strategy, that is, each UE associated to a particular BS gets an equal amount of resources. For each operator m, given B_m and μ_m, the number of the deployed

RRHs is constrained by a minimum guaranteed DL data rate R_{min} for the SC layer. Besides, a minimum UE offloading factor $\mathcal{O}_{min}^{\text{SC}}$ to the SC layer is set as a design constraint, because higher data rates are assumed for the SC layer.

4.3.1 Data Rate Model

For the computation of the data rate offered by the SC layer, we assume RRHs of height $h_{\text{SC}}[m]$ and UEs of height $h_{\text{UE}}[m]$, hence, the Signal-to-Noise-and-Interference Ratio (SNIR) at the UE side offered by a RRH located at a distance $d[m]$ can be computed by adopting the path model in [149], that is, with pathloss $PL[dB] = 148.1 + 37.6\, log_{10}(d_{LOS}[Km]) + X_{shad}$[1], where we assume that the shadowing factor follows a lognormal distribution with zero mean and variance σ_{shad} (i.e., $X_{shad} = \mathcal{N}(0, \sigma_{shad})$) [149]. Assuming that RRHs transmit with density power $P^{\text{TX}}[dBm/Hz]$ and antenna gain $g_H[dBi]$, and that UEs' receivers can be characterized by antenna gain $g_{\text{UE}}[dBi]$, thermal noise $N_t[dBm/Hz]$, noise figure $NF[dB]$, the SNIR at the receiver side can be computed according to the modified free-space propagation model[2] as $SNIR[dB] = P^{\text{TX}} + g_H + g_{\text{UE}} - PL - (N + I))$, where $I[dBm/Hz]$ represents the interference density power from other RRHs[3], while the noise density power can be computed as $N[dBm/Hz] = N_t + NF$.

With respect to the data rate provided to a specific UE by MNO m, we adopt the following model for N_axN_a multiple-input and multiple-output (MIMO) scheme [146]: $R_m^{UE}[bit/s] = N_a \cdot PRB_m \cdot R_{PRB} \cdot (1 - \theta_{ov}) \cdot \theta^{UE}$, where θ^{UE} is the percentage of PRBs allocated to the considered UE at a given instant, while R_{PRB} is the data rate provided over a single Physical Resource Block (PRB), and θ_{ov} represents a constant percentage of the total number of PRBs spent by each RRH for physical layer overhead (i.e., control and signaling). On the other hand, PRB_m is the number of PRBs that can be mapped over the bandwidth B_m. More precisely, considering a modulation index i and a coding factor c_F, the data rate provided over a single PRB can be defined as $R_{PRB} = i \cdot c_F \cdot 168/TTI$, where 168 is the number of symbols

[1] Please note that the expression for the path loss refers to the line-of-sight (LOS) distance between RRH and UE antennas, therefore, it can be computed taking into account that it holds $d[m] = d_{LOS} \cdot cos(\alpha)$, with altitude angle $\alpha = arctan(\frac{h_{\text{SC}} - h_{\text{UE}}}{d})$

[2] In PL we account for obstacles between RRHs' and UE's antennas by adding the shadowing factor X_{shad}.

[3] We remind that a unitary frequency reuse factor is assumed for SC layer of a given MNO.

transmitted over one PRB within a Transmission Time Interval (TTI) according to the Orthogonal Frequency-Division Multiplexing (OFDM) scheme adopted in LTE-A, with Frequency-division duplexing (FDD) and normal cyclic prefix. The values of i and c_F to adopt with respect to a specific Channel Quality Indicator (CQI) are provided by 3GPP standard [155]. However, as no standardization exists of the mapping between CQI and SNIR, we adopt the Modulation-and-Coding Scheme (MCS) selection criterion provided by [156].

The data rate R_m^g guaranteed by MNO m to its subscribers is defined as the average data rate offered over the SC layer in the worst-case scenario, where a unitary activity factor f_a is considered, that is, $R_m^g = \sum_{l=1}^{\mu_m N_{\mathrm{UE}}} R_m^{UE,l}/(\mu_m N_{\mathrm{UE}})$[4]. Because reasonable values for the activity factor are always below one (i.e., $f_a < 1$), in general UEs are offered with a greater data rate R_m^{off}. Indeed, resources are shared among less UEs within a given SC; besides, less interference is generated by RRHs to adjacent ones.

4.3.2 Power Model

According to the power model provided by the EARTH [146, 157] and iJoin [158] projects, RAN power consumption of a generic MNO m, henceforth P_m, can be divided into a term related to RRHs and the other to the BBU-pool: $P_m = P_m^h + P_m^u$. In both components, the power consumed is provided as a function of $PRB_{m,n}^{us}$, that is, the number of PRBs used for RRH n out of the available ones PRB_m in bandwidth B_m. Therefore, $PRB_{m,n}^{us}$ represents the total load of a generic RRH n and can be expressed as the sum of the PRBs needed for UE communications (i.e., $PRB_{m,n}^{\mathrm{UE}}$) and for the physical layer overhead (i.e., $PRB_{m,n}^{ov}$). Because equal overhead is assumed in the RRHs, the subscript n is omitted and the physical layer overhead of a generic RRH can be defined as $PRB_m^{ov} = \lceil \theta_{ov} PRB_m \rceil)$. In conclusion, for the total load of RRH n it holds $PRB_{m,n}^{us} = PRB_m^{ov} + PRB_{m,n}^{\mathrm{UE}}$.

[4] We remind that a proportional-fairness scheduling is adopted for the resource allocation of UEs connected to the same SC.

4.3.2.1 RRH Power Model

For computing the power consumption of a generic RRH with index $1 \leq n \leq H_m$, the EARTH model [146, 157] for C-RAN architectures is adopted:

$$P_{m,n}^{h} = \frac{P_{\mathrm{PRB}}^{\mathrm{TX}} PRB_{m,n}^{us} / \eta_{\mathrm{PA}} + N_a P^{\mathrm{RF}} PRB_m}{(1 - \sigma_{\mathrm{DC}})(1 - \sigma_m)} \tag{4.1}$$

where $P_{\mathrm{PRB}}^{\mathrm{TX}}$ is the radio frequency (RF) output power over one PRB assuming that no power adaptation is performed, that is, can be computed from the transmit density power $P^{\mathrm{TX}}[mW/Hz]$ as $P_{\mathrm{PRB}}^{\mathrm{TX}} = P^{\mathrm{TX}} \cdot B_m / PRB_m$. For each RRH, it is assumed that transceivers' power consumption scales linearly with the number of carriers, besides, η_{PA} represents the power amplifier efficiency, N_a is the number of antennas, P^{RF} is the power consumption of the RF transceiver for one PRB and σ_{DC}, σ_m are the loss coefficients due to DC-DC power supply and mains supply. By explicitly expressing $PRB_{m,n}^{us}$ as the sum of PRB_m^{ov} and $PRB_{m,n}^{\mathrm{UE}}$, and by isolating the static contributions to power consumption from the power consumed for UEs' traffic, (4.1) can be rewritten as:

$$P_{m,n}^{h} = P_m^{h,ov} + \Delta_p^h PRB_{m,n}^{\mathrm{UE}} \tag{4.2}$$

where $\Delta_p^h = P_{\mathrm{PRB}}^{\mathrm{TX}} / [\eta_{\mathrm{PA}} (1 - \sigma_{\mathrm{DC}})(1 - \sigma_m)]$ and $P_m^{h,ov}$ accounting for the static RRH power consumption components due to physical layer overhead and RF transceivers (i.e., $P_m^{h,ov}$ can be calculated by substituting $PRB_{m,n}^{us}$ with PRB_m^{ov} in (4.1)). Finally, the total power consumed by all the RRHs in the network is $P_m^h = \sum_{n=1}^{H_m} P_{m,n}^h$.

4.3.2.2 BBU Power Model

BBU functionalities are deployed by using VNF instances in identical COTS servers (e.g., x86) with equal processing capacity X_{cap} expressed in Giga Operations Per Second (GOPS). Each BBU server is able to instantiate multiple RRHs functionalities in the form of VNFs instances, which are soft resources that can be migrated among BBU servers and possibly shared among RRHs. Uniform workload share is considered among the servers and the necessary base-band computation needed for one PRB is modeled with a constant $\mathcal{K}_{\mathrm{TX}}$, when a specific transmission (TX) configuration is used [157].

In the worst-case of saturated RRHs (i.e., $f_a = 1$) the number of deployed BBU servers U_m has to be sufficient for supporting the base-band operations of the RRHs in the area, therefore, it is defined as $U_m = \lceil (\mathcal{K}_{\text{TX}} H_m PRB_m) / X_{cap} \rceil$. Conversely, in case of average load, only some of the available BBU servers need to be active U_m^{act} for supporting the total network load, while the remaining $U_m^{id} = U_m - U_m^{act}$ are idle and can go into sleep mode for energy consumption optimization (i.e., only cooling, power supply, etc. [114, 115]). As already mentioned, one possible way of calculating the optimum BBU-RRH mapping is by solving a Knapsack problem [149]; however, in this context, the ideal minimum number of active BBU servers defined is considered and computed as $U_m^{act} = \lceil \left(\mathcal{K}_{\text{TX}} \sum_{n=1}^{H_m} PRB_{m,n}^{us} \right) / X_{cap} \rceil$.

The power consumption of the whole BBU-pool can be expressed as the sum of a component $U_m^{act} P_{st}^u$ due to functionalities that are independent from the network load (e.g., FFT and IFFT [112]), plus $U_m^{id} P_{id}^u$ accounting for idle-state BBU servers, $H_m P_m^{u,ov}$ representing the power consumption due to RRHs' overhead and, finally, the network load $\Delta_p^u \sum_{n=1}^{H_m} PRB_{m,n}^{\text{UE}}$ [157]:

$$P_m^u = U_m^{act} P_{st}^u + U_m^{id} P_{id}^u + H_m P_m^{u,ov} + \Delta_p^u \sum_{n=1}^{H_m} PRB_{m,n}^{\text{UE}} \tag{4.3}$$

where $P_m^{u,ov} = \Delta_p^u PRB_m^{ov}$ is the consumption due to overhead processing of one RRH, and Δ_p^u is the power consumed per PRB when a specific transmission configuration and server kind are used

By defining $P_m^{ov} = P_m^{h,ov} + P_m^{u,ov}$ and $\Delta_p = \Delta_p^h + \Delta_p^u$, the total power P_m consumed by MNO m, considering both the RRH and BBU components defined in (4.2) and (4.3), can be rewritten as:

$$P_m = U_m^{act} P_{st}^u + U_m^{id} P_{id}^u + H_m P_m^{ov} + \Delta_p \sum_{n=1}^{H_m} PRB_{m,n}^{\text{UE}} \tag{4.4}$$

4.4 Coalitional Game

For the general cooperative game $(\mathcal{M}, V)$, the set of all the $2^{\mathcal{M}} \backslash \emptyset$ possible coalitions is represented with Ω, and with V_ω the *coalition payoff*, which can be considered as the maximum utility value that the set of players in coalition ω can jointly obtain. Let v_m be the portion of V_ω assigned to player m when participating to that coalition, named *player's payoff*, then a payoff allocation $\mathbf{v} \in \mathbb{R}^\omega$ is the vector representing a possible distribution of the payoffs among the $|\omega|$ players in coalition ω. Finally, the *core* $\mathbb{C}$ is the set of payoff allocations such that no group of players is willing to leave the grand coalition $\mathcal{M}$, that is, the one formed by all players, for one of the sub-coalitions.

Coalitional games are a specific class of cooperative games [159] addressing those problems where forming coalitions is preferred by the players of the game. A particular class of coalitional games are the canonical ones where joining the grand coalition $\mathcal{M}$ represents the most convenient choice. This means that the payoff that player m receives out of $V_{\mathcal{M}}$ is at least as large as the payoff it would receive in any of the disjoint sets of sub-coalitions $\Omega \backslash \mathcal{M}$. In this terms, the core $\mathbb{C}$ guarantees the stability of the grand coalition because the players have no incentive for leaving it.

Expressing the payoff allocation in the grand coalition with $\mathbf{v} \in V_{\mathcal{M}}$ and the one for a subcoalition with $\mathbf{v}' \in V_{\omega}$, a possible definition of the core is [159]:

$$\mathbb{C} = \{\mathbf{v} \in V_{\mathcal{M}} \mid \forall \omega,\ \nexists \mathbf{v}' \in V_{\omega}, s.t.\ v'_m > v_m, \forall m \in \omega\} \tag{4.5}$$

The core $\mathbb{C}$ might not exist and, in those cases, the grand coalition is considered unstable.

The objective of this chapter is to determine under which conditions the problem of cooperation between MNOs for sharing SC H-CRAN resources can be considered as a canonical coalitional game, or in other terms, when the grand coalition of MNOs is preferred to the sub-coalitions and when the opposite is true.

MNOs payoff in ω is modeled as their profit [154], defined as the difference between revenues ρ_m and costs C_m, when $m \in \omega$. It is assumed that the revenue of each MNO only depends on its own UEs and is not redistributed among MNOs. On the other hand, operators share the total H-CRAN costs C_{ω}, and $\mathbf{c} \in \mathbb{R}^{\omega}$ is the cost sharing vector, which reports the portion of it that each MNO is willing to pay. The payoff of MNO m according to $\mathbf{c}$ is:

$$v_m = \rho_m - C_m = \rho_m - c_m C_{\omega}, \quad m \in \omega \tag{4.6}$$

where $\sum_{m \in \omega} c_m = 1, 0 \leq c_m \leq 1$. Thus, the value of the generic coalition ω can be defined as the sum of its members' profit:

$$V_{\omega} = \sum_{m \in \omega} v_m = \sum_{m \in \omega} \rho_m - C_{\omega} \tag{4.7}$$

According to this model, the payoff allocation $\mathbf{v}$ among players is formed by a term that is not transferable to other players (i.e., revenues ρ_m), and by a second term on whose redistribution the coalition's players can agree (i.e., costs C_m). MNOs' payoffs v_m depend on the joint actions of the other players in that coalition and, depending on whether there exist restrictions on the distribution of the coalition's payoff V_ω among the players, the coalitional game defined in this section can be regarded as a transferable utility (TU), or non transferable utility (NTU) game. In particular, in NTU games, a flexible redistribution of the coalition's cost C_ω is not allowed, on the other hand, in TU games, the players can agree on the cost sharing vector $\mathbf{c}$ and can divide the coalition payoff V_ω in any manner [159].

When $\mathbb{C}$ exists, we assume that, among all the possible cost shares $\mathbf{c} \in \mathbb{R}^{|\mathcal{M}|}$, MNOs only accept to adopt the market share vector $\boldsymbol{\mu} \in \mathbb{R}^{|\mathcal{M}|}$ as unique and fair solution, thus, opting for a NTU game. This means that each operator pays a portion of the coalition cost proportional to the number of UEs it owns, as it is a rough but logical estimation of its load and cost contribution into the shared H-CRAN (see (4.4)). In other words, $c_m = \mu_m/\mu_\omega$, where $\mu_\omega = \sum_{m\in\omega} \mu_m$ is the market share of coalition ω.

Revenues and costs for the system model introduced in Section 4.3 are now defined.

4.4.1 Revenue Model

The revenue ρ_m is modeled as a price proportional to the guaranteed data rate R^g_ω guaranteed to the UEs when MNO m participates to coalition ω. More in detail, according to conventional business model for mobile data traffic presented in Chapter 1, the operator charges end users with a flat tariff τ_r [17], defined in this context in monetary units per unit of data rate per month [$€/Mbps/month$] [154].

Considering an investment period of one year, the revenue of MNO m over this period can be defined as below:

$$\rho_m = 12\,\tau_r\,R^g_\omega\,\mu_m\,N_{\mathrm{UE}} \tag{4.8}$$

4.4.2 Cost Model

By focusing on OPEX, the cost model reduces to the power consumption P_ω of coalition ω multiplied by a tariff applied over one year. Considering average power consumption $\overline{P_\omega}$ in $[W]$, the total cost function C_ω of coalition ω over one year is:

$$C_\omega = \overline{P_\omega} \cdot 10^{-3} \cdot \rho_{\mathrm{KWH}} \cdot 24 \cdot 365 \tag{4.9}$$

where $\rho_{KWh}[€/KWh]$ represents the price per unit of power set by energy providers for the reference period of one hour. Finally, taking into account that operators agree on pooling together their H-CRAN elements, P_ω is calculated as in (4.4), after substituting m with ω. Hence, the aggregated BW and the total number of PRB in a coalition can be represented as $B_\omega = \sum_{m\in\omega} B_m$ and $PRB_\omega = \sum_{m\in\omega} PRB_m$, respectively, while the total numbers of RRHs and BBU servers in coalition ω become $H_\omega = \sum_{m\in\omega} H_m$ and $U_\omega = \lceil (H_\omega PRB_\omega \mathcal{K}_{\mathrm{TX}}) / X_{cap} \rceil$.

In Table 4.1, we summarize the notations used for the main parameters introduced for the system model and coalitional game in Sections 4.3 and 4.4.

4.5 Results and Discussion

A custom simulator in Matlab has been implemented for estimating the number of RRHs and BBU servers to be deployed for a given combination of market share and available bandwidth, as well as for evaluating the offered QoS in terms of coverage and data rate (i.e., $\mathcal{O}_\omega^{\mathrm{SC}}$, R_ω^g, R_ω^{off}), together with revenues, costs and core existence under different coalitions. In the remainder of the chapter, the network setup and the results are presented.

4.5.1 Simulation Set Up

Adopting the system model depicted in Fig. 4.2, we study two scenarios where three operators have deployed their networks in an area of $4\,Km^2$ with $N_{\mathrm{UE}} = 20000$ users in total. The number of MNOs considered is a typical value in most European countries, as confirmed by [154]. Besides, the values of the considered area and number of UEs are representative of the typical coverage area of a single macrocell,

Table 4.1: System Model's and Coalitional Game's Notations for H-CRAN Sharing.[5]

4.3. System Model for H-CRAN	
Data Rate Model	
Variable	Definition
$\mathcal{A}$	Considered area
$\mathcal{M}$	Set of MNOs coexisting in $\mathcal{A}$
N_{UE}	Total # UEs in $\mathcal{A}$
f_a	Activity factor of UEs
m	Identifier of specific MNO
μ_m	Market share of MNO m over N_{UE}
B_m	Bandwidth of the licensed band owned by MNO m
PRB_m	# of PRBs that can be mapped over the bandwidth B_m
H_m	# of SC RRHs deployed by MNO m
U_m	# of COTS BBU servers deployed by MNO m
h_{SC}	RRHs' height
h_{UE}	UEs' height
d	Distance between a generic RRH-UE pair
d_{LOS}	LOS distance between a generic RRH-UE pair
α	Altitude angle between a generic RRH-UE pair
PL	Pathloss
X_{shad}	Shadowing factor
σ_{shad}	Variance of the lognormal distribution adopted for shadowing
P^{TX}	RRHs' transmit power in $[dBm/Hz]$
g_H	RRHs' antenna gain in $[dBi]$
g_{UE}	UEs' antenna gain in $[dBi]$
N_t	UE's receiver thermal noise in $[dBm/Hz]$
NF	UE's receiver noise figure in $[dBm]$
I	Interference density power from other RRHs to UEs in $[dBm/Hz]$
N	Noise density power in $[dBm/Hz]$
N_a	# of antennas used in MIMO at transmitter/receiver side
θ_{ov}	Percentage of PRBs spent for PHY layer overhead
θ^{UE}	Percentage of PRBs allocated to a generic UE at given time
i	Modulation index

[5] Sub/superscripts are omitted when generic parameters are considered. Sub/superscript m and ω are interchangeable, for representing MNOs' or coalitions (aggregate) parameters.

c_F	Coding factor
R_{PRB}	Data rate provided over a single PRB with specific MCS
$R_m^{UE,l}$	Data rate provided to UE l by MNO m
R_{min}	Constraint on the minimum guaranteed DL data rate used by MNOs for SC layer's design
R_m^g	Average data rate guaranteed by MNO m to its subscribers when $f_a = 1$
R_m^{off}	Average data rate offered by MNO m to its subscribers when $f_a < 1$
SNR_{min}	UEs' power sensitivity during association process
$\mathcal{O}_{min}^{\mathrm{SC}}$	Constraint on the minimum offloading factor used by MNOs for SC layer's design
$\mathcal{O}_m^{\mathrm{SC}}$	Average # of subscribers offloaded to the SC layer by MNO m
Power Model	
Variable	Definition
n	Identifier of specific RRH
$PRB_{m,n}^{us}$	# of PRBs used by MNO m for RRH n
$PRB_{m,n}^{\mathrm{UE}}$	PRBs needed for UE communications
$PRB_{m,n}^{\mathrm{OV}}$	PRBs needed for physical layer overhead
$P_{\mathrm{PRB}}^{\mathrm{TX}}$	RRH RF output power over one PRB without adaptation
P^{TX}	RRH transmit density power
η_{PA}	RRH power amplifier efficiency
P^{RF}	Power consumption of the RRH RF transceiver for one PRB
σ_{DC}	RRH loss coefficient due to DC-DC power supply
σ_m	RRH loss coefficient due to mains supply
Δ_p^h	RRH power consumption per PRB with specific TX config.
$P_m^{h,ov}$	Static RRH power consumption due to overhead and RF transceivers
$P_{m,n}^h$	Power consumption of RRH n in MNO m
P_m^h	Total RRH power consumption of MNO m
$\mathcal{K}_{\mathrm{TX}}$	BBU cloud computing requirements for base-band processing of a single PRB
X_{cap}	Processing capacity of a BBU server
Δ_p^u	BBU power consumption per PRB with specific server

	and TX config.
Δ_p	Radio and BBU power consumption per PRB with specific server and TX config.
U_m^{act}	# of BBU servers needed for supporting traffic of MNO m at specific time
P_{id}^u	Power consumption of BBU servers in idle-state
P_{st}^u	Baseline power consumption of active BBU servers
$P_m^{u,ov}$	BBU power consumption for single RRH's overhead
P_m^{ov}	Radio and BBU power consumption due to overhead of one RRH
P_m^u	BBU-pool power consumption of MNO m
P_m	RAN power consumption of MNO m
4.4. Coalitional Game	
Variable	Definition
$\boldsymbol{\mu}$	Market share vector considereing all MNOs coexisting in $\mathcal{A}$
Ω	Set of all possible coalitions of MNOs
$\mathcal{M}$	Grand coalition (i.e., formed by all players)
$\mathbb{C}$	Core (i.e., payoff allocations for being $\mathcal{M}$ the most convenient in Ω)
ω	Coalition formed by $\lvert\omega\rvert$ MNOs
V_ω	Payoff (also called profit or value) for coalition ω
v_m	Payoff of MNO m in ω
$\mathbf{v}$	Payoff allocation for MNOs in ω
τ_r	Tariff in [€/$Mbps/month$] charged by MNOs to subscribers
ρ_m	Revenues for MNO m in ω
$\overline{P_\omega}$	Average aggregate power consumption for MNOs in coalition ω
ρ_{KWH}	Price in [€/KWh] set by energy providers
C_ω	Total H-CRAN costs for coalition ω
c_m	Percentage of C_ω paid by MNO m in ω
$\mathbf{c}$	Sharing vector agreed by MNOs in ω for C_ω
C_m	Costs paid by MNO m in ω
G_m^ω	Profit gain of MNO m in ω with respect to stand-alone approach

Table 4.2: Scenarios for H-CRAN sharing.

	Scenario A			**Scenario B**		
m	1	2	3	1	2	3
$B_m[MHz]$ [160]	20	20	20	5	15	20
PRB_m [160]	100	100	100	25	75	100
μ_m	1/3	1/3	1/3	0.1	0.3	0.6
H_m	247	193	200	247	234	214
U_m	156	125	130	40	76	139

that is, of the SC-layer extension and of typical European population densities [154]. Below, we present the two sample scenarios described in Table 4.2:

- Scenario A: The operators have approximately the same size, equal market share and bandwidth capabilities.
- Scenario B: The operators have different sizes, different market shares and bandwidth capabilities proportional to their market share.

In both cases, the MNOs have deployed their network in order to satisfy the constraints on $\mathcal{O}^{\text{SC}}_{min}$ and R_{min} defined in Section 4.3. For each operator, the number H_m of RRHs and the guaranteed data rate R^g_ω are calculated in the worst-case scenario where $f_a = 1$. On average $f_a < 1$ and the offered data rate R^{off}_ω is greater than the data rate R^g_ω for which end users are charged. The number of RRHs H_m and of BBU servers U_m deployed by MNOs in different scenarios are provided in Table 4.2, while the setup for remaining system parameters is summarized in Table 4.3.

4.5.2 Performance Results

In order to study the benefits in terms of QoS improvement guaranteed by cooperation among MNOs, the average data rate R^{off}_ω offered to UEs is plotted in Fig. 4.3 for

[6] Computed for a system with 10 MHz [146], that is, over a reference bandwidth of 9 MHz [163] and 50 PRBs [155].

[7] We define the minimum data rate according to video streaming (i.e., 4G killer application [2]), therefore, we model R_{min} as the recommended data rate for the minimim streaming resolution considered, that is, wide 360p [164] with bitrates between 400 $Kbps$ and and 1 $Mbps$ [165].

Table 4.3: System Setup for Coopeative H-CRAN Sharing.

Physical Layer		**Femto Cell RRH [146]**	
Parameters	Values	Parameters	Values
$h_{\mathrm{SC}}[m]$	10 [161]	$P^{\mathrm{TX}}_{\mathrm{PRB}}[mW]$	1[6]
$h_{\mathrm{UE}}[m]$	1.5	η_{PA}	4.4%
$SNR_{min}[dB]$	-4	$P^{\mathrm{RF}}[mW]$	12
$g_H, g_{\mathrm{UE}}[dBi]$	0 [146]	σ_{DC}	9%
σ_{shad}	5 [149]	σ_m	11%
$N_t[dBm/Hz]$	-174 [149]	**Intel Xeon E5540 BBU [157]**	
$NF[dB]$	5 [161]	Parameters	Values
N_a	2 [146]	$X_{cap}[GFLOPS]$	324
Path loss	$148.1 + 37.6\, log_{10}(d[Km])$ [149]	$\mathcal{K}_{\mathrm{TX}}$	2.0978
UE, RRH	uniform distribution [161]	$P^u_{id}[W]$	3 [114]
θ_{ov}	30%	$P^u_{st}[W]$	120
f_a	0.16 [146]	Δ^u_p	0.6125

Cooperative Game			
Parameters	Values	Parameters	Values
N_{UE}	20000 [154]	$\tau_r[(€/Mbps)/month]$	$[0.1, 0.92]$
$\mathcal{A}[Km^2]$	4 [154]	$\rho_{\mathrm{KWH}}[€/KWh]$	0.12 [162]
$m \in \mathcal{M}$	$\{1, 2, 3\}$	$\mathcal{O}^{\mathrm{SC}}_{min}$	80%
Ω	$2^{\mathcal{M}}\backslash\emptyset$	$R_{min}[Mbps]$	0.78[7]

Scenario B over the percentage of UEs associated with the small cells layer (i.e., the offloading factor $\mathcal{O}^{\mathrm{SC}}$)[8]. It can be observed that, in stand-alone scenarios, both the offloading factor and data rate are quite low but always above the values of $\mathcal{O}^{\mathrm{SC}}_{min} = 0.8$ and $R_{min} = 0.78$ defined in Table 4.3. However, the joint DSM of the pooled spectrum is capable of enhancing the coverage and data rate provided with the small cell network, with the offloading factor approaching one when $|\omega|$ increases. In particular, by forming coalitions of two, the MNOs may significantly improve the offered data rate and the offloading potential. Finally, the grand coalition (i.e., the cooperation among all three operators) provides a very high offloading factor and the highest data rate among all scenarios.

[8] Similar results can be obtained for Scenario A, although, with coinciding values of $\mathcal{O}^{\mathrm{SC}}_{min}$ and R_{min} for the cases with one and two MNOs coalitions.

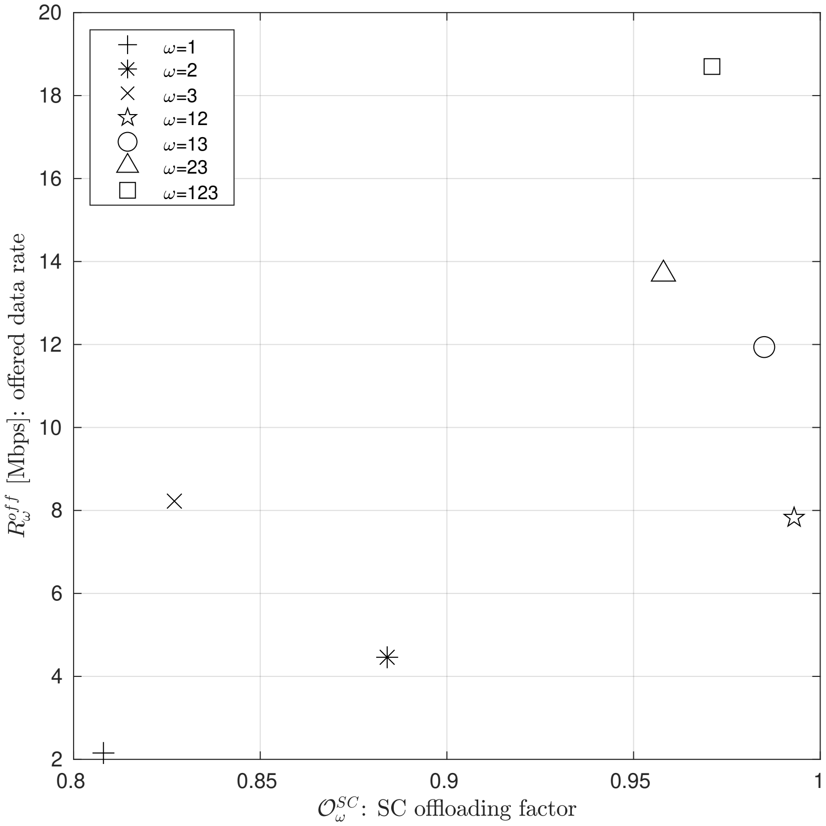


Figure 4.3: Offered data rate R_{ω}^{off} vs offloading factor of UEs to the SC layer $\mathcal{O}_{\omega}^{\mathrm{SC}}$, provided by individual MNOs ($\omega = m$), sub-coaltions of MNOs (i.e., $\omega = m_1 m_2$), or by the grand coalition $\omega = 123$, according to Scenario B defined in Table 4.2.

This can be explained taking into account that by, pooling the network elements (i.e., RRHs and BBU servers) and aggregating the bandwidth, UEs are more likely to be in the proximity of a SC with a wider communication bandwidth, which will be chosen instead of the macro layer, thus increasing the offloading factor and improving available data rates. Hence, the results confirm the expected benefits from fine-scale sharing, thanks to a more efficient utilization of the frequency resource in the spatial dimension. Please note that, by cooperating and without bandwidth extension in the mmWave frequencies, the offered average data rate is already greater than the average 6.5 Mbps estimated for 2020 in [2].

Next, in Fig. 4.4, the profit V_{ω} is plotted for all possible stable coalitions, applying (4.7) when the core is nonempty. For Scenario A, MNOs have similar profits when operating individually, since they all have same market share and spectrum. On

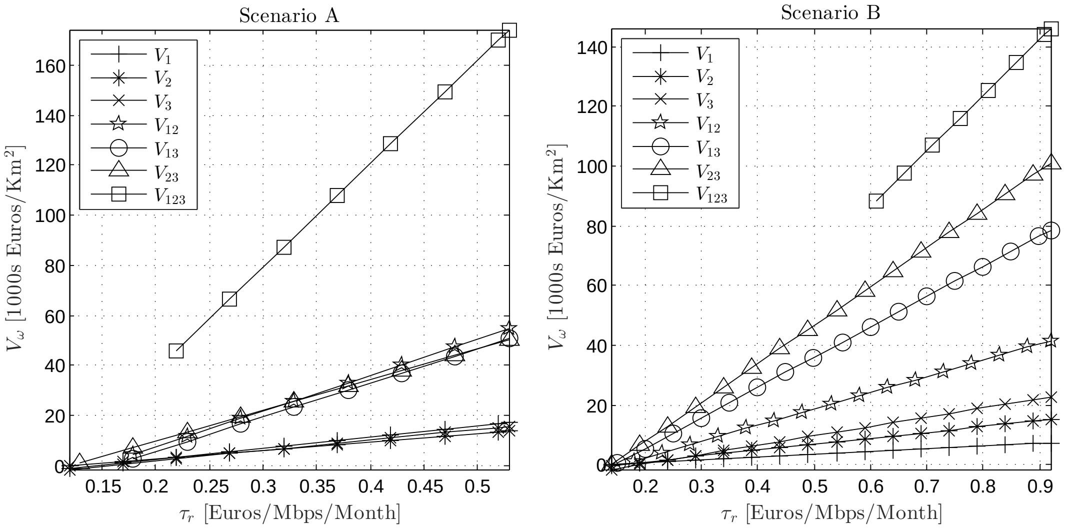
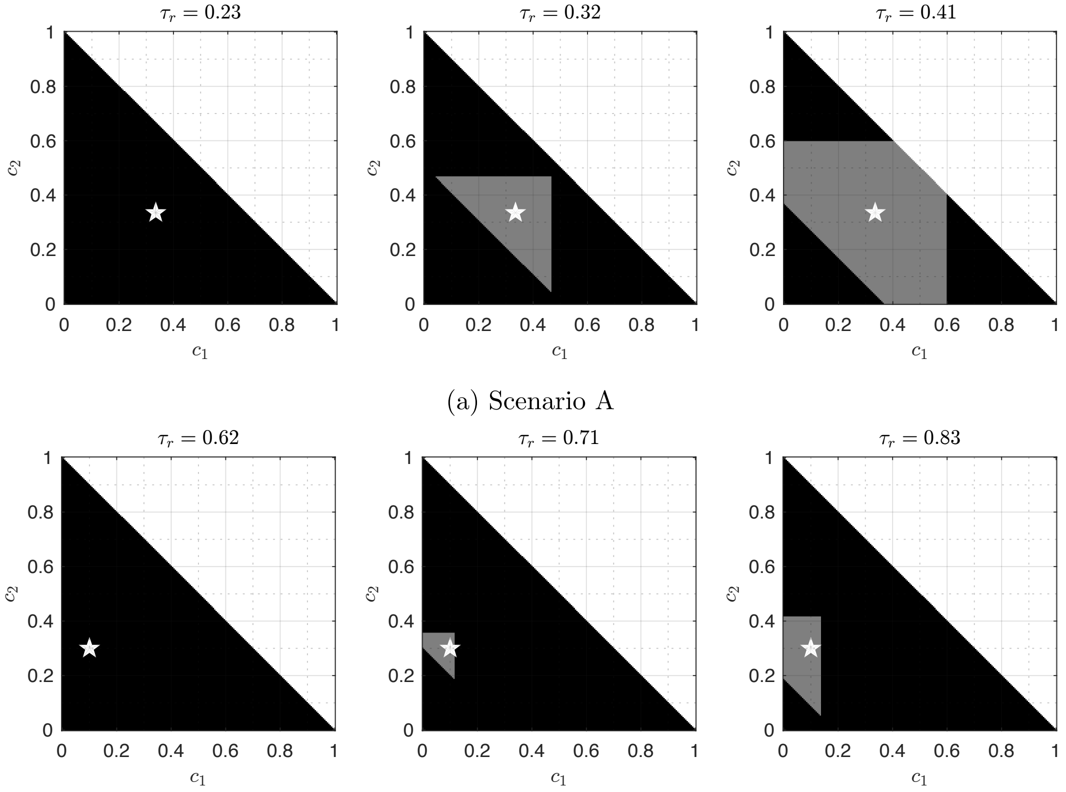


Figure 4.4: Coalitions payoffs V_ω in stable conditions computed over one year. ω equals the identifiers of the MNO's participating in the coalition, that is, $\omega = m$ in case of individual MNOs, $\omega = m_1m_2$ in case of sub-coaltions, and, $\omega = 123$ for the grand coalition.

the other hand, in Scenario B, the MNO with highest market share (i.e., MNO_3) achieves higher profits when compared to the other operators.

It can clearly be observed that, in both scenarios, any pair of MNOs always forms a stable sub-coalition $\omega = m_1m_2$ with payoff $V_{m_1m_2}$, meaning that for their members MNO_{m_1} and MNO_{m_2} it is always preferable cooperating rather than working individually (i.e., $V_{m_1m_2} > V_m$). This can be explained by the fact that, when forming a coalition, the spatial maximization of the pooled resources enable better QoS to the UEs and increased revenues for the MNOs. Besides, as it will be better explained below in comparison with the grand coalition, MNOs achieve higher profits in sub-coalitions, independently of the tariff charged to end users, because the cost increase is negligible with respect to revenues. However, not all sub-coalitions offer the same profit to their members. This can be observed in Scenario B, where sub-coalitions involving the largest operator (i.e., MNO_3) achieve higher aggregate profits thanks to the better QoS achieved.

As far as the grand coalition $\mathcal{M} = 123$ is concerned, note that it can always provide significantly higher profits than any subcoalition (i.e., $V_{123} > V_{m_1m_2} > V_m$), for both scenarios. However, it becomes stable (i.e., the core exists) only when a



(a) Scenario A

(b) Scenario B

Figure 4.5: Grand coalition cost share

minimum tariff τ_r is reached, which equals 0.23 and 0.62 in Scenarios A and B, respectively. This is mainly due to the additional costs associated with operation in bigger aggregated bandwidth, which represents the price to pay for a better spectrum utilization over the spatial dimension. Indeed, more processing power is needed at the BBU servers, as the size of the BBU-pool U_ω and the power consumed for control and signaling depends on the total BW (see Section 4.3.2). In the case of sub-coalitions ω, the bandwidth increase is relatively small, therefore, the extra power consumed is negligible. However, when all resources are pooled into the grand coalition $\mathcal{M}$, the BW aggregation at all RRHs' premises sets additional costs, which become dominant. Hence, a minimum tariff is needed so that the revenues compensate for the increased OPEX, leading to a stable grand coalition.

Fig. 4.5 represents the cost allocations c_m for each MNO_m, with the allocations c_1 and c_2 represented in the x and y axis, respectively, and c_3 derived as $c_3 = 1 - c_1 - c_2$. For each scenario, three sample tariff values are considered, starting from the minimum value that supports the formation of a stable grand coalition (i.e., the existence of the core) and with a difference of approximately ten cents from each other. The grey areas represent the possible allocations within the core in a TU game, whereas the white star represents the point for which the cost allocations coincide with the market share. For the minimum value of the tariff τ_r the only possible allocation within the core is the allocation according to the market share (i.e., the grey area collapses into a single point and the game is NTU), whereas for increasing tariffs more allocations become possible within the core of a TU game (i.e., the grey area becomes larger).

It can be observed that the market share (i.e., the white star) always belongs to the core (i.e., the grey area). In Scenario A, where all MNOs have equal market shares, the core of the TU game is symmetrical, whereas in Scenario B, the core moves towards the low-left corner, as the major part of the cost is assigned to MNO_3, which holds the largest market share. Furthermore, for similar tariffs, the core dimension is much higher in Scenario A, due to the steeper slopes of the profit curves (see Fig. 4.4), which, in case of TU games, would enable a more flexible redistribution of the grand coalition costs among the MNOs.

By comparing the two scenarios in Fig. 4.4 and Fig. 4.5, it can be noticed that the minimum tariff required for a stable grand coalition in Scenario A (i.e., $\tau_r = 0.23$) is much smaller with respect to Scenario B (i.e., $\tau_r = 0.62$). Indeed, as already observed in Fig. 4.5, for low values of τ_r, the core coincides with the market share, and operators are forced to adopt it as the sole stable payoff distribution. Thus, for Scenario A, the three equal sized MNOs have the same incentive for joining the grand coalition when the minimum tariff is charged to the end users. On the other hand, in Scenario B, the market share is unbalanced and, as a result, the operator with the highest market share (i.e., MNO_3) must assume the greater portion of the costs. For that reason, when low tariffs are charged to the end users, MNO_3 prefers forming sub-coalitions, where, as explained before, a better revenue can be achieved in exchange for a negligible increase in OPEX.

Another important observation is that, in case of TU games, although for low tariffs the MNOs only wish to share the grand coalition costs proportionally to the number of UEs owned (i.e., according to the market share), for increasing tariffs, other payoff distributions become feasible. Indeed, the core expands and the MNOs may reach

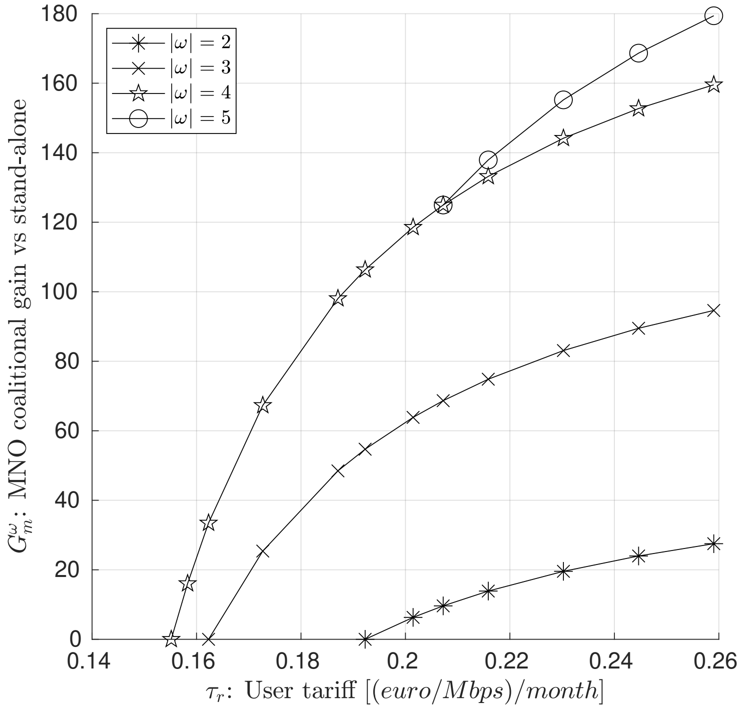


Figure 4.6: Profit gain through fine-scale sharing

new agreements on how to share costs, while maintaining a better profits when compared with a stand-alone approach. For example, as an extreme case, in Scenario A and for $\tau_r = 0.41$, two MNOs could afford to pay for the whole H-CRAN cost, letting the third MNO operating with zero cost. In Scenario B, such configuration is possible for a higher tariff (i.e., $\tau_r = 0.71$), where MNOs could agree on a null cost allocation for the smallest MNO (i.e., MNO_1). Such agreement is justified by the fact that MNO_1's contribution to the pooled infrastructure and resources provides the remaining members of the grand coalition with enhanced QoS and revenue opportunities. However, the reduced market share of MNO_1 generates a very small traffic load and resource usage over the shared network, with a negligible impact on the total network cost.

In order to better understand the potential of a cooperative approach with a higher number of players, an extension of Scenario A is provided for up to five MNOs. More precisely, identical MNOs are considered with deployment equal to MNO_3 in Scenario A described in Table 4.2 (i.e., $H_m = 200$ and $U_m = 130$). Therefore, coalitions ω are classified by means of their cardinality $|\omega|$, rather than by the identity of the participating MNOs. For the grand coalition $\mathcal{M}$, it holds $|\mathcal{M}|= 5$.

Fig. 4.6 depicts the MNO profit gain over the tariff charged to the end users, for the cooperative approach with different coalition sizes, compared with a baseline non-cooperative scenario. The profit gain of MNO_m in coalition ω as a percentage of the profit in the stand-alone approach is defined as $G_m^{\omega} = 100 \cdot (V_{\omega}/|\omega| - v_m) / v_m$, and similar conclusions can be deduced as for Fig. 4.4 on the benefits and conditions for the formation of coalitions of different sizes: i) coalitions with a higher number of MNOs normally achieve higher gains due to better offered data rates, and, ii) a minimum tariff is required for paying back the additional coalition's costs, when compared with the stand-alone case.

In addition, a better detail is provided with respect to Fig. 4.4 on the dependence of the profit gain on the tariff charged to the end users, as well as on the minimum tariff required with respect to the coalition size. More in detail, although the gains achieved through cooperation are always enormous, their slope decreases as the tariff charged to the end users increases. In other words, the real advantage in cooperating comes when low tariffs are set to the end users, that is, when the revenues derive from the boost in QoS offered by a better utilization of the frequency resource in the spatial dimension, rather than from inflated tariffs.

As far as the minimum tariff of subcoalitions is concerned (i.e., up to $|\omega| = 4$), bigger coalitions are always preferable to smaller ones, as a lower minimum tariff can be charged to the end users, while higher gains are provided. In other words, on the one hand, MNOs could charge the end users with the same tariff while achieving higher profits with respect to the stand-alone case. On the other hand, similar profits could be achieved by MNOs in a cooperative approach while charging lower tariffs to the end users, thus attracting the segment of user subscribers looking for cheaper prices for a specific QoS. Different is the case of the grand coalition (i.e., for $|\omega| = 5$), which provides the highest gain only when a minimum tariff is charged in order to compensate the additional costs of a larger network. Consequently, a coalitional approach can provide MNOs with information on the minimum pricing to be charged in order to maximize their profits. Alternatively, the coalitions could be chosen by MNOs according to the tariff that they want to charge to end users, depending on MNOs' target segment within subscribers' market.

4.6 Summary

In this chapter, a novel scheme has been proposed for the assessment of potential cooperation incentives when MNOs coexist in the same area, and it has been defined according to a game theoretic framework. In particular, coalitional game theory has been exploited for studying the performance and financial gains offered through fine-scale network sharing. Different scenarios have been considered for different market and spectrum shares in the three and five operators case. The QoS improvement is highlighted in terms of data rate offered to the end users, and profit gains for the operators. The minimum user pricing schemes have been obtained for the stability of coalitions of different sizes.

The results show that, as long as the defined conditions are respected, collaborating is more convenient with respect to the stand-alone case, providing financial incentives to MNOs for upgrading their networks towards 5G. Besides, although the market share represents a fair cost redistribution within coalitions, for sufficient tariffs MNOs could agree on unbalanced cost redistributions exploiting the better revenue opportunities offered by fine-scale network sharing with respect to the additional costs. In general, MNOs have a higher profit margin when cooperating with equal-sized operators, achieving higher profits with much lower tariffs. Finally, a rationale is provided to MNOs for choosing tariff to charge and the coalition to form with respect to their business objectives, that is, maximizing profits with respect to current market shares, or attracting new segments of the subscribers' market.

Chapter 5

Fine-Scale Slice Allocation for 5G Networks

Overview

Adapting to recent trends in mobile communications towards 5G, infrastructure owners are gradually modifying their systems for supporting the network programmability paradigm and for participating in the slice market (i.e., dynamic leasing of virtual network slices to third-party service providers). Two-fold are the advantages offered by this upgrade: i) enabling next-generation services through programmable policies and customized QoS guarantees, and, ii) allowing new profit opportunities deriving from sharing interactions with service providers. Many efforts exist already in the field of admission control, resource allocation and pricing for virtualized networks. Most of the 5G-related research efforts focus on technological enhancements for making existing solutions compliant to the strict requirements of next-generation networks. On the other hand, the profit opportunities associated to the slice market also need to be reconsidered in order to assess the feasibility of this new business model. Nonetheless, when economic aspects are studied in the literature, technical constraints are generally oversimplified. For this reason, in this chapter, we propose an admission control mechanism for intra-service network slicing that meets 5G timeliness while maximizing network infrastructure providers' revenue, reducing expenditures and providing a fair slice provision to competing service providers. To this aim, we *design an admission policy of reduced complexity based on bid selection, we study the optimal strategy in different circumstances (i.e., pool size of available resources, service providers' strategy and traffic load), analyze the performance metrics and compare the proposal against reference approaches. Finally, we explore the case where infrastructure providers lease network slices either on-demand or on a periodic time basis and provide a performance comparison between the two approaches. This analysis shows that the proposed approach outperforms existing solutions, especially in the case of infrastructures with large pool of resources and under intense traffic conditions.*

Contributions

[J2] M. Vincenzi, E. Lopez-Aguilera, and E. Garcia-Villegas, "Maximizing infrastructure providers' revenue through network slicing in 5g," *IEEE Access*, vol. 7, pp. 128283–128297, 2019. (Area: Telecommunications; Quartile Q1; IF: 3,745).

5.1 Related works

From an economic point of view, the enablers of a healthy slice market for 5G are: i) the monetary incentives to InPs for amortizing the costs of building the next-generation network, and ii) the fairness in the service of competing SPs. On the other hand, from a purely technical point of view, the requirements for 5G are: i) the slice isolation [26], ii) heterogeneous E2E QoS guarantees for 5G use cases [3–6, 138, 166], and, iii) a prompt slice provision, suitable for short-lived services such as emergency services or surveillance [5, 6].

Excluding architectural and technological aspects that have been extensively studied in the literature, the promptness in the slice provision is mainly regulated by two factors, that is, the communication protocol adopted between SPs and InPs, and the mechanisms used at the InPs' side for admission control, resource allocation and pricing. In this context, two macro categories of slice provision approaches exist in the literature, the *on-demand* and *periodic slicing* where, respectively, slice allocation is enforced upon each slice request arrival (e.g., policy-based approaches) or periodically (e.g., auction-based approaches). In on-demand slicing, the typical communication flow for the slice provision process consists in the uncoordinated slice request submission by SPs, followed by the broadcasting of the admitted tenants by InPs. On the other hand, in periodic slicing, an intrinsic latency is systematically added by the time window used for collecting slice requests.

Within this categorization, two strategies are mainly used in the literature for resource pricing. In on-demand slicing, prices are typically set by InPs for a given bundle of resources. On the other hand, in periodic slicing, prices are determined in relation to the resource availability as well as InPs' and SPs' strategies. Besides, a bidding model is generally adopted where the minimum and maximum bid represent, respectively, the reserve price (i.e., the minimum price accepted by the InPs), and the SPs' budget (i.e., the maximum affordable price).

Many contributions exist in the literature for admission control, resource allocation and billing mechanisms in virtualized wireless networks [167], however, rarely both 5G requirements and the economic conditions for a healthy slice market are met. Consequently, the discussion remains open in the scientific community with respect to automated mechanisms for slice provision and pricing in 5G. In particular, [21] and [168] propose, for the admission control in inter-service network slicing (i.e., slice allocation to SPs providing different services), on-demand strategies that maximize the InPs' profit by means of Semi-Markov Decision Processes and optimization theory, respectively. Moreover, [21] introduces the concepts of *inelastic* and *elastic services*, that will be used in the following, and which are associated to SLAs characterized by constant or average QoS requirements, respectively. However, both contributions lack in the review of other performance metrics relevant for 5G, for instance, fairness towards competing SPs.

On the other hand, among the proposed periodic approaches, [169, 170] employ auction theory for the study of the single/heterogeneous resource allocation problem, respectively, nevertheless, neither of the works puts a focus on network isolation, QoS support or fairness. Besides, although InPs are the entities entitled to build next-generation networks, many contributions only take into account the economic return for SPs. For instance, this is the case of the spectrum leasing optimization framework presented in [171], the Fisher market slice allocation approach with strategic tenants in [172], the auction-based approach in [170] and, in general, the sealed-bid auctions [173]. Finally, only limited efforts have been produced in the study of pricing schemes suitable for 5G, for instance, [174,175] propose auction-based solutions for heterogeneous resource slicing with a per-access pricing scheme. It is important to remark how the authors in [174] highlight the need for a pricing scheme based on slices' lifetime in order to account for the real resource occupation, and to reduce the risk of exaggerated slice requests and unused resources.

In conclusion, research efforts focusing in on-demand and periodic slicing tend to study complementary aspects related to the 5G slice market, therefore, we consider interesting a direct comparison between the two strategies through the same analytical framework. In this context, [176] extends the on-demand approach in [21] for the study of InPs' profits to the periodic case with heterogeneous resources. However, static InP strategies are adopted with no hint on the optimal admission strategy, nor on the fairness towards competing SPs. Reference [168] partly completes the contribution in [176] by proposing a genetic-based algorithm for online computation of the admission policy that maximizes InP's profit, however, no performance metric is provided regarding the fair treatment of differnet SPs.

5.2 Policy-Based Slicing for QoS and Profit Optimization

In this chapter, we propose a timely admission control mechanism for intra-service network slicing (i.e., slice allocation to SPs providing the same kind of service) that takes into account the economic conditions for a healthy slice market and addresses the requirements of next-generation networks by maximizing InPs' revenues, reducing operational expenditures, and guaranteeing fairness towards SPs, slice isolation and QoS. In this context, InPs adopt the programmable infrastructure and the enhanced architecture introduced in Chapter 3 and have the joint objective of maximizing the tenants' admission rate while prioritizing the most rewarding slice requests. Therefore, from a technological point of view, InPs have the incentive to perform the slice allocation process as fast as possible once triggered by the arrival of a slice request, since every request represents a potential source of revenue. On the other hand, from a strategical point of view, the InPs have the incentive to prioritize those slice requests with higher bids and characterized by a high ratio among arrival and service rates. More in detail, we adopt the promptness offered by on-demand approaches for the admission of new slices, combined with pricing features typical of periodic slicing, where tariffs are set depending on the resource availability, the InPs' strategy and SPs' behavior. Indeed, we assume that SPs may have a different perception of the market and, therefore, make different bids for the same kind of slice. However, as SPs' strategies have been abundantly studied in the literature and the focus of this chapter remains on InPs' perspective, we assume that SPs are irrational entities that follow a random bidding model. Moreover, we assume that tenants pay for the slices they use only if the associated SLA is met during their permanence in the network, therefore, InPs can reallocate resources only after voluntary tenants' departures.

In order to maximize the slice provision promptness and the InPs' revenue while reducing the computational cost associated to the admission decision, we propose the Above Threshold (AT) policy-based approach that admits slice requests with associated tariff-bids greater or equal than a given threshold. Such an approach is capable of maximizing tenants' admission rate while prioritizing the most rewarding slice requests and, at the same time, it minimizes the admission delay as policies can be enforced instantaneously upon each slice request arrival. In this regard, we compare the performance of two kinds of admission strategies differing in the admission strategy with respect to the resource utilization, named *State Dependent (SD)* and *State Independent (SI)* policies, respectively. In particular, the former uses admission thresholds that can adapt to the current resource utilization and guarantees a maximum revenue for every number of instantiated slices, that is, it depends on the available resources. On the other hand, the second adopts static admission thresholds, thus requiring lower computational expenses and maximizing revenues only in the long term. In this chapter, we model only SLAs associated to inelastic services as they are the strictest class of SLAs. Either way, an extension of this chapter to include elastic services can be achieved by following the modeling approach in [21].

We provide a benchmark of the proposed admission control mechanism for network slicing in 5G by comparing on-demand and periodic slicing performance (i.e., fairness towards SPs, resource utilization, InP's profit, and timeliness) with that of reference strategies when different resource pool sizes, traffic loads, and slicing frequencies are considered: i) in the on-demand case, the always-admit (AA) policy that admits every slice request regardless of the associated bid, and, in the periodic case, ii) the first-come-first-served (FCFS) policy that admits requests according to the order of

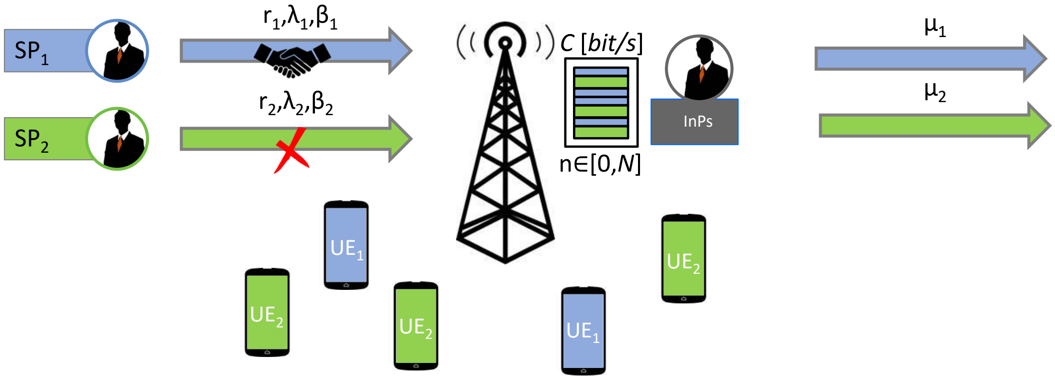


Figure 5.1: System model for slice provision when one InP leases resources to multiple SPs competing for providing service to their UEs. Colors identify the portion of resources used (e.g., channel capacity C) and the UEs served by different slice tenants. Rejected slice requests are marked with a red cross.

arrival (i.e., independently of the associated bids), and, iii) the best bid (BB) policy that admits requests from the highest to the lowest bid (i.e., prioritizes SPs with highest spending power).

5.3 System Model

In this section, we introduce the system model adopted for the analysis and, to this aim, we refer to Fig. 5.1. In the considered scenario, multiple UEs coexist within the coverage area of a given BS, which belongs to a given InP. The BS represents the access point towards other network resources, such as backhaul, IP networks and cloud infrastructures. In Fig. 5.1, the different colors identify different SPs, as well as the UEs served by different slice tenants and the portion of InP's resources accessed (e.g., assigned portion of the total access link capacity C). Within this context, different service instances of the same UE are represented as different logical UEs (i.e., UEs can possibly access multiple service instances at a time, provided by the same or by different SPs).

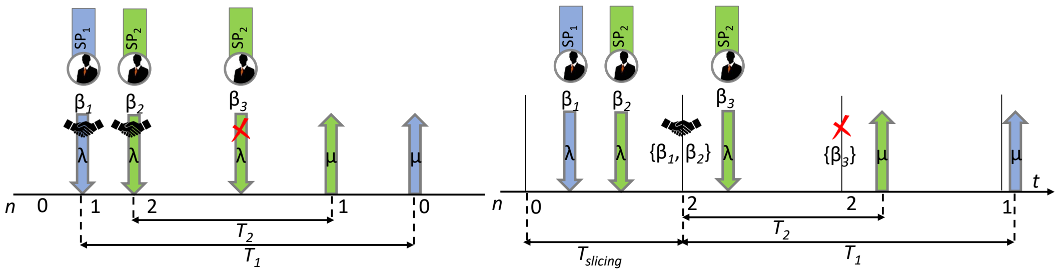
Resources are sliced independently at different BS locations and SPs are allowed to actively request network slices on a continuous time scale, while InPs monitor the resource availability and decide whether to admit them, either in real-time (i.e., on-demand slicing) or on a discrete time-scale (i.e., periodic slicing). Whenever

InPs welcome a new SP, named *slice tenant*, a SLA is stipulated defining the terms for the customization and pricing of the requested slice. In other words, each SLA defines both the QoS to be guaranteed and the tariff β_s in monetary units per second (e.g., $[euros/s]$) to be paid by the SP tenant s during its permanence in the system. Finally, the labeling s is associated with a slice requests rather than with a particular UE or SP, indeed, in Fig. 5.1, a specific tenant can be licensee of multiple network slices simultaneously, especially when SPs opt for serving different UEs by means of separate slices.

In order to specify a clear model for the SLAs, we first introduce the concept of *service (or slice) class* that we define as $c = \{\overline{\boldsymbol{r}}_c, \lambda_c, \mu_c, \tau_c\}$. In this context, $\overline{\boldsymbol{r}}_c$ represents the requirements vector, that is, the set of requirements r_e for each resource kind e accessible from the considered service area, while λ_c and μ_c are the average arrival and service rates of slice requests for the specific service class c, respectively. More precisely, $T_c = 1/\mu_c$ is the average *holding (or service) time* for a specific class c, that is, the average time interval during which resources are retained by SPs providing such service. In other words, it holds $T_c = \mathbf{E}[T_{c|s}]$, where $T_{c|s}$ is the holding time of a specific SP tenant s.

The heterogeneous resource profiles of different slice classes are mapped by the InP into a feasibility region $\boldsymbol{\mathcal{F}}$, whose contours are defined according to the resource pool of the InP. In particular, the allocation state at a specific slice interval is modeled by position vector $\overline{\boldsymbol{n}}$ in a multi-dimensional space, which is defined in each dimension by the number of slices n_c currently allocated to a specific slice class c. The set of feasible allocation states is formed by the number of slices that can be simultaneously allocated to each class (i.e., $\overline{\boldsymbol{n}} \in \boldsymbol{\mathcal{F}}$), and a resource sharing vector $\overline{\boldsymbol{\sigma}}_c$ is associated to each slice class c, where sharing factor σ_c^e indicates the share over total amount of resource e allocated to c. If we consider a policy region within $\boldsymbol{\mathcal{F}}$ that limits the actual number of slices that can be allocated to each class according to InP's prioritization of different services (similar to [135]), we can split the joint allocation of heterogeneous slice classes into c separate allocation problems. Therefore, the projection of the policy region over a specific dimension provides a variable maximum number of slices N_c that can be allocated to a given slice class c at a given instant (i.e., $n_c \leq N_c$).

Fixed a specific service class c, the SLA for a given SP tenant s is defined as the tuple



(a)

(b)

Figure 5.2: Instance of the slice request, tenants' departure and bidding processes in: a) on-demand and b) periodic slicing, when only one service class is supported and $N = 2$. Different colors identify requests and departures of different SPs, moreover, rejected requests are marked by a red cross.

$\{c, \beta_s\}$, where $\beta_s T_{c|s}$ is the price paid to the InPs if the resource requirements are guaranteed during the whole holding time. As introduced in Section 5.2, we examine only the strictest kind of SLAs, that is, those associated to inelastic services [21], characterized by constant requirements during the whole holding time. Besides, we assume that the tariff-bid β_s of a generic SP tenant s can vary within the interval $[\beta_m^c, \beta_M^c]$, that changes for different slice classes c as they are characterized by different associated resources and perceived value. More in detail, the extremes of the bid interval represent, respectively, the minimum tariff accepted by the InP (i.e., the reserve tariff β_m^c) and the maximum tariff that SPs can afford to pay for the considered slice class (i.e., the tariff budget β_M^c).

In this context, as we study the problem of intra-service network slicing, in the following, we investigate the resource allocation to only one service class, reminding that, when a policy region is enforced by InPs within the feasibility region $\boldsymbol{\mathcal{F}}$, the same methodology can be applied separately to each of the service classes. Therefore, all SPs ask the InPs for the same requirements, and, the notation can be simplified by omitting sub/superscript c, while SLAs of different tenants are fully described by the corresponding bids β_s. Besides, we model only access network resources, that is, the channel capacity C of the access link to the BS, measured in $[bit/s]$ because, due to their scarcity they represent the bottleneck in the E2E slice provision **[J2]**, [177]. Hence, the service classes' definition can be projected into a single resource dimension, by substituting the requirements vector $\overline{\boldsymbol{r}}$ with the scalar r, that represents the aggregate nominal rate asked by tenants for the service of UEs in the considered coverage area (i.e., sub/superscript e is omitted)[1]. In this specific case, the maximum number n of slices that can be allocated simultaneously for a given service class and time instant is $N = \lfloor \sigma C/r \rfloor$, and it holds $0 \leq n \leq N$.

The proposed system model is valid for both on-demand and periodic slicing, that is, when n is updated at each new admission and departure, or regularly every $T_{slicing}$ seconds. In the following, we assume that the slice request arrivals can be modeled as a Poisson stochastic process with average rate λ, and the tenants' departure as a general stochastic process with average rate μ. With regards to the pricing model, we describe different SPs' behaviors by adopting a bidding model where β_s is a random variable following a general distribution f_β over the sample space $[\beta_m, \beta_M]$.

In Fig. 5.2, we depict an instance of the slice request, tenants' departure and bidding processes for both approaches. Besides, we highlight the possibility for the InP to reject slice requests depending on the resource availability, the received bids, and the adopted admission policy. Moreover, in the periodic case, slice requests received during a given slicing interval can be admitted at the beginning of the next interval only, when SLAs are enforced. In particular, tenants pay for slices only when they utilize resources, therefore, InPs get no revenue in the time interval between tenants' departure and following slicing interval. We remind that, as we model the problem in function of the aggregate resource demand from the InP perspective only, multiple slice instances can correspond to the same tenant, as represented in Fig. 5.2.

5.4 System Analysis for On-demand Slicing

In this section, we present the mathematical analysis for on-demand slice provision mechanisms when different policies are adopted. Regardless of the policy, the infrastructure can be represented as a cloud server farm with capacity to instantiate N equal virtual servers (i.e., the network slices) that share a common pool of jobs to be executed (i.e., the service requests of a given class). New jobs are characterized by an average arrival and service rate equal to λ and μ, respectively, and the number n of jobs executed is updated upon every new job's arrival and completion. Besides,

[1] We remark that r only depends on the resource requirements of the considered slice class, while σ at a given instant is obtained from the policy region defined by the InP and depends on the allocation state n at the previous slice allocation interval.

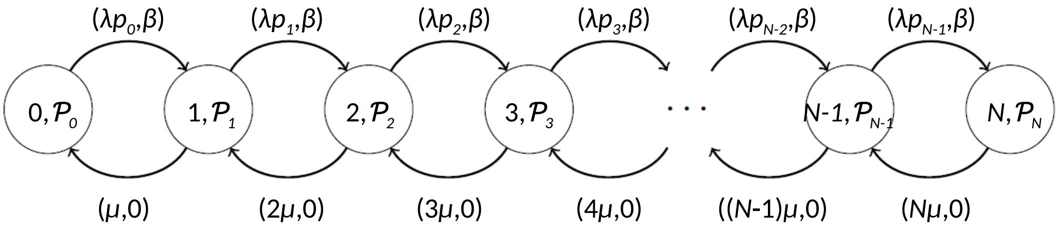


Figure 5.3: Markov chain for on-demand slicing systems, where a different number of instantiated slices n and policy $\mathcal{P}_n$ is associated to each state, while transitions are jointly represented by a transition rate ω_{nn^+} and a reward ϕ_{nn^+}.

we assume that each virtual server can handle one job at a time, in order to model the slice isolation requirement and the QoS guarantee. Therefore, thanks to the memoryless assumption on arrivals and departures, we can model the system as a $M/G/k/k$ queue[2]. Even in cases where these assumptions do not apply (e.g., non-Markovian behavior of SPs), discrete-time Markov chains could be applied. However, the needed transformations lie outside the scope of this chapter.

The mathematical framework offered by continuous-time Markov chain (CTMC) can be used for the mathematical analysis of the considered problem. We can refer to Fig. 5.3, where each state corresponds to a different tuple $(n, \mathcal{P}_n)$, whose elements describe the number of instantiated slices and the admission policy adopted at that state, respectively. Besides, the generic transition from state n to n^+ coincides either with the admission or departure of a slice tenant, and is associated with the tuple $(\omega_{nn^+}, \phi_{nn^+})$ representing the transition rate conditioned to the initial state and the associated reward, respectively.

The state policy $\mathcal{P}_n$ represents any possible bid-based criterion for admitting or rejecting incoming slice requests at state n and we remark that it depends on the maximum number of slices N that can be allocated to the considered slice class at a given instant, according to the policy region defined by the InP:

$$\mathcal{P}_n = \begin{cases} Admit, & \text{if } \beta \in \mathcal{D}_n \wedge n < N \\ Reject, & \text{otherwise} \end{cases} \tag{5.1}$$

where $\mathcal{D}_n \subset [\beta_m, \beta_M]$ is the admitted bid interval at state n. Consequently, the

[2] It shall be noticed that, in the case of periodic slicing, the system can be modeled as a $M^X/G/k/k$ queue, since we could consider that the slice requests received within a given slicing interval arrive in batches at the beginning of the next interval.

probability for a new slice request to be admitted at state n can be defined as $p_n(f_\beta, \mathcal{P}_n) = p_{\{\beta \in \mathcal{D}_n\}} = \int_{\mathcal{D}_n} f_\beta(\beta)\, d\beta$. State policies $\mathcal{P}_n$ can be arbitrarily chosen by the InP when resources are available in the system, that is, for states $0 \leq n \leq N-1$. On the other hand, when the system faces resource shortage (i.e., $n = N$), the only applicable policy is the rejection of any slice request, that is, $\mathcal{D}_N = \emptyset$ and, thus, $p_N = 0$. Finally, the tuple $(\omega_{nn^+}, \phi_{nn^+})$ associated to a transition at state n can be written as $(\lambda p_n, \beta)$ in case of admission, and as $(n\mu, 0)$ in case of departure. In conclusion, for the generic transition nn^+ it holds:

$$\omega_{nn^+} = \begin{cases} \lambda p_n, & \text{if } 0 \leq n \leq N-1, n^+ = n+1 \\ n\mu, & \text{if } 1 \leq n \leq N, n^+ = n-1 \\ 0, & \text{otherwise} \end{cases} \tag{5.2}$$

$$\phi_{nn^+} = \begin{cases} \beta, & \text{if } 0 \leq n \leq N-1, n^+ = n+1 \\ 0, & \text{otherwise} \end{cases} \tag{5.3}$$

As introduced in Section 5.2, we assume that the InP can adopt either SD or SI policies, which differ in the capability of adapting the admission strategy to the number of slices isolated in the system. In particular, different or equal policies $\mathcal{P}_n$ are enforced at different states n, respectively. Hence, InP's strategy is represented with the *policy vector* $\boldsymbol{\mathcal{P}} = (\mathcal{P}_0, \ \cdots, \ \mathcal{P}_{N-1})$ in the SD case, while it can be fully described by the generic state policy $\mathcal{P}$ when SI approaches are adopted (i.e., $\boldsymbol{\mathcal{P}} = \mathcal{P}$).

5.4.1 State-Dependent policies

In CTMC, the stationary probability π_n associated to the generic state n of the system can be calculated through the following balance equations, when SD policies are enforced:

- $0: \quad \pi_0 \lambda p_0 = \pi_1 \mu$
- $1: \quad \pi_1(\lambda p_1 + \mu) = \pi_0 \lambda p_0 + \pi_2 2\mu$
- $n: \quad \pi_n(\lambda p_n + n\mu) = \pi_{n-1}\lambda p_{n-1} + \pi_{n+1}(n+1)\mu$
- $N: \quad \sum_{n=0}^{N} \pi_n = 1$

leading to:

$$\begin{aligned}\pi_0\Big(\frac{\lambda}{\mu}, f_\beta, N, \boldsymbol{\mathcal{P}}\Big) &= \frac{1}{1+\sum_{k=1}^{N}(\frac{\lambda}{\mu})^k/k!\prod_{l=0}^{k-1}p_l} \\ \pi_{n\geq 1}\Big(\frac{\lambda}{\mu}, f_\beta, N, \boldsymbol{\mathcal{P}}\Big) &= \frac{(\frac{\lambda}{\mu})^n/n!\prod_{q=0}^{n-1}p_q}{1+\sum_{k=1}^{N}(\frac{\lambda}{\mu})^k/k!\prod_{l=0}^{k-1}p_l}\end{aligned} \tag{5.4}$$

Intuitively, in a low-load regime (i.e., when $\frac{\lambda}{\mu} \to 0$), the system most likely operates in states corresponding to low values of n (i.e., $\pi_0 \to 1$), independently of the bidding distribution f_β, the maximum number of slices N, and the InP's strategy $\boldsymbol{\mathcal{P}}$. The same result is obtained under high-load regime (i.e., $\frac{\lambda}{\mu} >> N$), and when a very conservative admission strategy is adopted by the InP (i.e., β_m is increased so that most of the bid distribution lies outside $\mathcal{D}_n$). Conversely, when a more permissive policy is used in high-load regime, the system behavior can be reversed (i.e., $\pi_N \approx 1$).

Following, we obtain the analytical expression for the performance metrics used to measure the efficiency of such slice provision system. The admission probability can be expressed as:

$$P_{admit}\Big(\frac{\lambda}{\mu}, f_\beta, N, \boldsymbol{\mathcal{P}}\Big) = \sum_{n=0}^{N-1}\pi_n p_n \tag{5.5}$$

and represents the probability for a new slice request to be admitted independently of the number of slices already instantiated in the system. According to (5.4) and (5.5), P_{admit} totally depends on the admission probability at state $n = 0$ in low-load regime (i.e., $P_{admit} \to p_0$, when $\frac{\lambda}{\mu} \to 0$). Therefore, according to (5.1) and to p_n's definition, the InP can improve the system's fairness (i.e., the general satisfaction of competing SPs) by widening the admission interval $\mathcal{D}_0$. More precisely, the maximum admission probability in low-load regime can be reached when the state policy $\mathcal{P}_0$ admits every request (i.e., $p_0 = 1$) or, in other words, when the admission interval $\mathcal{D}_0$ includes the entire support of f_β.

The average resource utilization U in the system is defined as the ratio between the average and the maximum number of slices instantiated in the system:

$$U\Big(\frac{\lambda}{\mu}, f_\beta, N, \boldsymbol{\mathcal{P}}\Big) = \mathbf{E}[n]/N = \Big(\sum_{n=0}^{N} n\cdot\pi_n\Big)/N \tag{5.6}$$

Subsequently, we introduce the expected tariff $\mathbf{E}[\beta|\beta \in \mathcal{D}_n]$ paid by those slice tenants that are admitted at state n according to state policy $\mathcal{P}_n$:

$$\begin{aligned}\mathbf{E}[\beta|\beta \in \mathcal{D}_n] &= \int_{-\infty}^{\infty} \beta\, p_{\{\beta|\beta\in\mathcal{D}_n\}}\, d\beta \\ &= \frac{1}{p_n}\int_{\mathcal{D}_n} \beta\, f(\beta)\, d\beta\end{aligned} \tag{5.7}$$

where $p_{\{\beta|\beta\in\mathcal{D}_n\}} = (f_\beta(\beta) \cdot 1|_{\beta\in\mathcal{D}_n})/p_n$.

The average revenue rate R_β in $[euros/s]$ for an InP applying a specific policy vector $\boldsymbol{\mathcal{P}}$ can be calculated by averaging, over all the states, the admission rate λp_n in $[admissions/s]$, times the expected price paid by admitted tenants over the average holding time, that is, $\mathbf{E}[\beta|\beta \in \mathcal{D}_n]/\mu$ in $[euros/admission]$:

$$R_\beta\Big(\frac{\lambda}{\mu}, f_\beta, N, \boldsymbol{\mathcal{P}}\Big) = \frac{\lambda}{\mu}\sum_{n=0}^{N-1} \pi_n p_n \mathbf{E}[\beta|\beta \in \mathcal{D}_n] \tag{5.8}$$

5.4.2 State-Independent policies

The analytical expressions for stationary probabilities and performance metrics of a SI system can be obtained as a particular case of the SD case. In particular, by definition of SI policy, it holds $\mathcal{P}_n = \mathcal{P}$, $\mathcal{D}_n = \mathcal{D}$ and $p_n = p$ for every state $0 \le n \le N-1$. Therefore, we can rewrite the stationary probabilities in (5.4) as:

$$\pi_n\Big(\frac{\lambda}{\mu}, f_\beta, N, \mathcal{P}\Big) = \frac{(\frac{\lambda}{\mu}p)^n/n!}{\sum_{k=0}^{N}(\frac{\lambda}{\mu}p)^k/k!}, \qquad n \ge 0 \tag{5.9}$$

Similarly, the system admission probability in (5.5) can be rewritten as:

$$P_{admit}\Big(\frac{\lambda}{\mu}, f_\beta, N, \mathcal{P}\Big) = (1-\pi_N)p \tag{5.10}$$

Finally, the definitions of U and $\mathbf{E}[\beta|\beta \in \mathcal{D}]$ remain unchanged, while the expression for the average revenue rate in (5.8) can be simplified as below and expressed as an explicit function of P_{admit}:

$$R_\beta\Big(\frac{\lambda}{\mu}, f_\beta, N, \mathcal{P}\Big) = \frac{\lambda}{\mu}P_{admit}\mathbf{E}[\beta|\beta \in \mathcal{D}] \tag{5.11}$$

A particular SI admission strategy is the AA policy introduced in Section 5.2 that admits every slice request regardless of the associated bid (i.e., $\mathcal{D} = [\beta_m, \beta_M]$ and $p = 1$), such that, according to (5.7), $\mathbf{E}[\beta|\beta \in \mathcal{D}] = \mathbf{E}[\beta]$.

5.4.3 Optimal policy and Complexity

In Section 5.2, we motivated the maximization of the average revenue rate as the main InP's objective, therefore, we seek the solution $\boldsymbol{\mathcal{P}}_\nu^{opt}$ of the following maximization problem:

$$\begin{aligned} &\boldsymbol{\mathcal{P}}_\nu^{opt} = \arg\max_{\boldsymbol{\mathcal{P}}} R_\beta\Big(\nu, f_\beta, \boldsymbol{\mathcal{P}}\Big) \\ &\boldsymbol{\mathcal{P}} = (\mathcal{P}_0, \ \cdots, \ \mathcal{P}_{N-1}) \\ &\mathcal{P}_n \text{ see (5.1): } \mathcal{D}_n \subset [\beta_m, \beta_M] \end{aligned} \tag{5.12}$$

where $\nu = (\lambda/\mu, N)$ represents the state condition, that is, the traffic load and resource availability, of the network node considered. The problem highlights that the InP has to compute $\boldsymbol{\mathcal{P}}_\nu^{opt}$ offline for values of λ/μ and f_β that are representative of SPs' behavior in its network in order to adopt convenient strategies accordingly[3].

In order to define the search space for the optimal policy, we remind that, according to (5.1), the admission interval of a generic state policy $\mathcal{P}_n$ can be any subset of the bid interval. Hence, the admission interval can be generically represented as the composition of multiple disjoint admission intervals[4]. However, in order to reduce the complexity of the problem described in (5.12), we propose the adoption of AT policies where an admission threshold $\dot{\beta}_n$ is set at state n, such that according to (5.1) $\mathcal{D}_n = [\dot{\beta}_n, \beta_M]$ and $\dot{\beta}_n \geq \beta_m$. Accordingly, the system policy $\boldsymbol{\mathcal{P}}$ can be fully described by the *threshold vector* $\dot{\boldsymbol{\beta}} = (\dot{\beta}_0, \ \cdots, \ \dot{\beta}_{N-1})$ in the SD case and by the scalar $\dot{\beta}$ in the SI case, respectively. Thus, the search space for the optimal policy is reduced, and the problem in (5.12) can be transformed into an N-dimensional or mono-dimensional continuous optimization problem for SD and SI policies, respectively. On the other hand, a reduction in the achieved revenue rate is expected when compared to the optimal policy. However, as we demonstrate in the next section, the relative loss remains constrained with respect to different load regimes.

For the particular case of AT policies, the performance metrics' expressions can be adapted as explained below. Because the InP admits slice requests at state n only when the tariff-bid is higher than threshold $\dot{\beta}_n$, the admission probability at state n is

[3] The InP can estimate the SPs' traffic patterns using network tracing, and employ traffic forecasting mechanisms [178–180] together with machine learning tools for adapting the strategy on-the-fly.

[4] i.e., $\mathcal{D}_n = \bigcup_\alpha \mathcal{D}_n^\alpha$, with $\mathcal{D}_n^\alpha = [\beta_m^\alpha, \beta_M^\alpha] \subset [\beta_m, \beta_M]$ and $\mathcal{D}_n^{\alpha_1} \cap \mathcal{D}_n^{\alpha_2} = \emptyset$, $\forall \alpha_1 \neq \alpha_2$.

$p_n(f_\beta, \dot{\beta}_n) = 1 - CDF(\dot{\beta}_n)$. It is straightforward that p_n is a monotonically decreasing function of $\dot{\beta}_n$ as $\frac{dp_n}{d\dot{\beta}_n} = -f_\beta(\beta) \leq 0$. Besides, for the most conservative and permissive admission strategies it holds, respectively, $p_n(f_\beta, \beta_m) = 1$ and $p_n(f_\beta, \beta_M) = 0$.

The admission probability P_{admit}, the average resource utilization U and the average revenue rate R_β remain unchanged. On the other hand, the expected tariff-bid for tenants admitted at state n equals $\mathbf{E}[\beta|\beta \geq \dot{\beta}_n] = \frac{1}{p_n}\int_{\dot{\beta}_n}^{\beta_M} \beta\, f_\beta(\beta)\, d\beta$, that is a non-negative function of $\dot{\beta}_n$ (i.e., according to Leibniz's integral rule $\frac{d\mathbf{E}[\beta|\beta \geq \dot{\beta}_n]}{d\dot{\beta}_n} = \frac{f_\beta(\beta)}{p_n}\Big(\mathbf{E}[\beta|\beta \geq \dot{\beta}_n] - \dot{\beta}_n\Big) \geq \dot{\beta}_n \frac{f_\beta(\beta)}{p_n}\Big(\frac{1}{p_n}\int_{\dot{\beta}_n}^{\beta_M} f_\beta(\beta)\, d\beta - 1\Big) = 0$). For the most conservative and permissive admission strategies it holds $\mathbf{E}[\beta|\beta \geq \beta_m] = \mathbf{E}[\beta]$ and $\mathbf{E}[\beta|\beta \geq \beta_M] = \beta_M \geq \mathbf{E}[\beta|\beta \geq \beta_m]$, respectively. In the particular case of uniformly distributed bids, it holds for AT policies $p_n = \frac{\beta_M - \dot{\beta}_n}{\beta_M - \beta_m}$, $\mathbf{E}[\beta|\beta \geq \dot{\beta}_n] = \frac{\beta_M + \dot{\beta}_n}{2}$ and AA policies can be considered as a particular case of SI AT policies with threshold $\dot{\beta} = \beta_m$, that is, $\mathbf{E}[\beta|\beta \geq \dot{\beta}] = \mathbf{E}[\beta] = \frac{\beta_M + \beta_m}{2}$ when $p = 1$.

The average revenue rate for SD AT, SI AT and AA policies can be written as:

$$R_\beta^{SD} = \frac{1}{2}\frac{\lambda}{\mu}\frac{1}{\beta_M - \beta_m}\sum_{n=0}^{N-1}\pi_n(\beta_M^2 - \dot{\beta}_n^2)$$

$$R_\beta^{SI} = \frac{1}{2}\frac{\lambda}{\mu}(1 - \pi_N)\frac{\beta_M^2 - \dot{\beta}_n^2}{\beta_M - \beta_m}$$

$$R_\beta^{AA} = \frac{1}{2}\frac{\lambda}{\mu}(1 - \pi_N)(\beta_M + \beta_m)$$

In order to further improve the tractability while conserving accuracy, we convert the problem into a combinatorial optimization problem by discretizing the sample space $[\beta_m, \beta_M]$ into a finite number h of intervals. Hence, the thresholds that can be used for the state policies' definition are:

$$\dot{\beta}_n = \beta_m + j\frac{(\beta_M - \beta_m)}{h}, \quad j = 0, \ldots, h-1 \tag{5.13}$$

and the choice of a suitable value of h guarantees results' accuracy while keeping computational costs at acceptable levels, as it is demonstrated in the following section. Therefore, the combinatorial version of the problem described in (5.12) can be adapted for AT policies as described below, and, in the following, its solution will be referred to as *optimal AT policy*:

$$\begin{aligned}
&\dot{\boldsymbol{\beta}}_{\nu}^{opt} = \arg\max_{\dot{\boldsymbol{\beta}}} R_{\beta}\Big(\nu, f_{\beta}, \dot{\boldsymbol{\beta}}\Big) \\
&\dot{\boldsymbol{\beta}} = (\dot{\beta}_0, \ \cdots, \ \dot{\beta}_{N-1}) \\
&\dot{\beta}_n = \beta_m + j\frac{(\beta_M - \beta_m)}{h}, \, j = 0, \ldots, h-1
\end{aligned}$$

We remind that, as introduced in Section 5.2, the objective of this chapter is to propose a prompt admission control mechanism for network slicing in 5G and to compare its performance with that of baseline solutions. Because proposed AT policies enable admission strategies at reduced complexity, we adopt in this chapter an exhaustive search (ES) of the optimal policy for demonstration purposes only, leaving for future extensions the search of a more computational efficient method. Fixed the size of the pool of resources N, the complexity of an exhaustive search for the optimal AT policy in SD and SI systems is polynomial (i.e., $\mathcal{O}(h^N)$) or linear (i.e., $\mathcal{O}(h)$), respectively, with regards to the discretization levels h. Note that, depending on the value of h, multiple solutions of the problem may exist, and, in those cases, we choose the solution that maximizes P_{admit}; that is, the solution that minimizes the Euclidean norm of the threshold vector (i.e., $||\dot{\boldsymbol{\beta}}||_2$ or $\dot{\beta}$ for SD and SI systems, respectively).

5.5 Results and Discussion

In this section, we present and compare the performance of different slice provision mechanisms for both on-demand and periodic slicing when different policies are employed.

5.5.1 System Setup

We examine different pool of resources and the extreme case where SPs follow a per-UE slicing strategy. In the case of small cells, according to [181] up to 5 simultaneously active UEs can be served, hence, we assume a maximum number of slices $N = 6$. For the traffic model, we consider low, medium and high arrival rates λ, ranging from 0.5 to 100. On the other hand, we adopt only one service class with exponentially distributed departures and unitary average service rate μ. The bid interval varies within the range $[\beta_m, \beta_M] = [0, 100]$ representing, respectively, the minimum tariff accepted by the InP and the SPs' budget. Finally, we provide results for the case where SPs make uniform bids over the admitted interval (i.e., $\beta \sim \mathcal{U}[\beta_m, \beta_M]$).

For the solution of the combinatorial problem for AT policies associated to the problem described in (5.12), we employ a number h of discretization levels for the bidding region that ranges from a minimum of 2 (i.e., *low* and *high bid region*) up to a maximum of $h = 10$, allowing a higher precision. Besides, we develop a tool in Matlab for the performance evaluation of the different considered mechanisms. In particular, for the case of on-demand slicing with uniformly distributed bids, AT and AA performance is evaluated according to the expressions introduced in Sections 5.4.2 and 5.4.1. On the other hand, for periodic slicing, a simulator generates instances of the request arrivals, tenants' departure and bidding processes, and enforces AT, FCFS and BB policies accordingly for different slicing intervals. Finally, we remind that the optimal AT policy is computed by means of exhaustive search, and, in the periodic case, it is obtained separately for different values of the slicing interval $T_{slicing}$.

In the remainder of this section, first we focus in on-demand slicing, computing the optimal AT policy and comparing SD and SI approaches, when AA policy is used as a benchmark. Lastly, for the periodic case, we study the optimal AT policy for different slicing intervals, and we compare the performance with that of FCFS and BB policies.

5.5.2 Performance evaluation

5.5.2.1 On-demand slicing

In Section 5.4.3, we anticipated that a reduced-complexity solution to the problem introduced in (5.12) exists in the form of AT policy with discretized thresholds, but this approach may suffer some penalty on the revenue. Consequently, we now study the limits of its performance by comparing the average revenue rate of the optimal AT policy with that of an ideal tool we named *Oracle*. More in detail, in this context, we consider the most flexible type of AT policy, that is, the SD approach with maximum

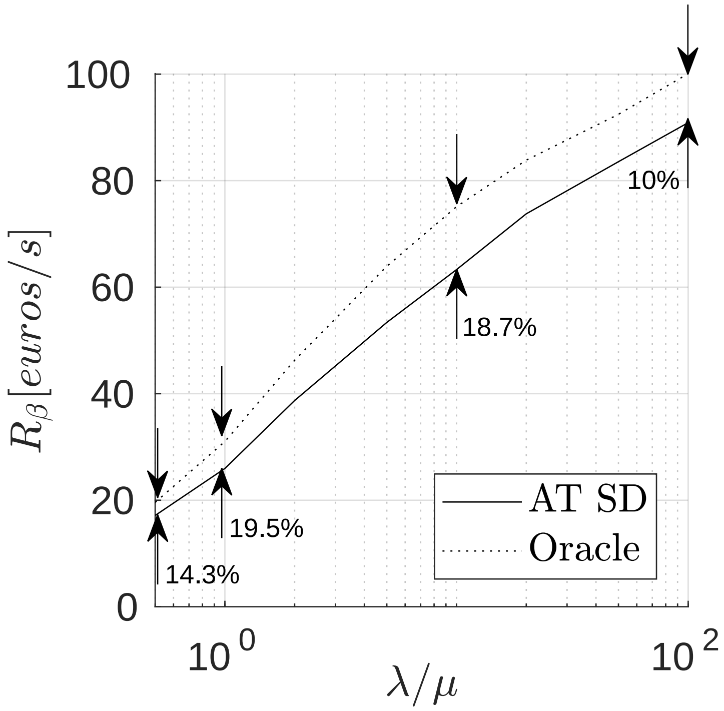


Figure 5.4: Assessment of the revenue loss for AT policy with respect to an ideal Oracle, when SD systems are considered, $N = 1$, and $h = 10$.

definition over the bid interval (i.e., $h = 10$). Oracle, on the other hand, is capable of recognizing the most rewarding bids. Oracle is applied a posteriori (i.e., once the simulation is finished) and, therefore, it can apply admission decisions based on its full knowledge of all the events in the simulation (i.e., slice requests, tenants' departures and bids). Hence, Oracle is only used for benchmarking purposes as it cannot be implemented in practice.

In Fig. 5.4, we present the average revenue rate for both optimal AT policy and Oracle with respect to the load regime (i.e., λ/μ) in logarithmic scale. To this aim, we study the most resource-limited case (i.e., $N = 1$), which leaves AT policies with the least flexibility in terms of resource availability, for counterbalancing Oracle's knowledge of future events. We remind that InPs aim at the joint maximization of admission rate and prioritization of highest bids and that, according to Section 5.4.1, resources are exhausted (i.e., $\pi_N \approx 1$) in high-load regime (i.e., when $\frac{\lambda}{\mu} >> N$). Consequently, when a larger pool of slice requests is received by InPs, the latter are motivated to adopt a more selective admission criterion by raising the bid threshold, which leads to a revenue enhancement at the expense of the admission probability (i.e., according to (5.5) it holds $P_{admit} \approx 0$). It can be observed from the figure that both Oracle and AT policies can achieve a logarithmic increase with respect to $\frac{\lambda}{\mu}$. On the other hand, a loss in revenues is expected with respect to Oracle, as raising the admission threshold translates in revenue maximization in the long term, while Oracle is capable of selecting best bids over each realization of the slice request process. The graph shows that the loss in revenues remains bounded for any load regimes, and, in particular, a 14.3% loss is experienced when few revenue opportunities are available (i.e., $\frac{\lambda}{\mu} \to 0$), it increases to 19.5% when arrivals are N times the departures (i.e., $\frac{\lambda}{\mu} \approx N$), while it reduces for high-load regimes (i.e., $\frac{\lambda}{\mu} >> N$). For instance, AT policies undergo a loss in revenue of 10% when $\frac{\lambda}{\mu} = 100$. Therefore, AT policies offer a near-optimal but viable solution to the generic optimization problem represented in (5.12).

Before comparing the optimal strategies in SD and SI systems, we study the influence of discretization over the complexity of the optimization problem and the accuracy of results. In order to study the feasibility of adopting an exhaustive search for benchmarking analysis, we provide the computation times associated to an exhaustive search of the optimal AT policy in this system setup for an infrastructure capable of hosting up to six slice tenants (i.e., $N = 6$). To this aim, we employ an Intel(R) Core(TM) i9-7900X CPU @ 3.30GHz with 64GB of RAM, and results reveal that when $h = 4$, 1.8 milliseconds are necessary for a SI system against the 4.5 milliseconds for a SD system. Besides, when $h = 10$, 6.6 seconds are necessary for a SI system against the 64.5 minutes for SD systems. Therefore, within the considered system setup, computation times remain limited for both systems, although SI systems are preferable when big infrastructures are being studied, and when many combinations of $\frac{\lambda}{\mu}$ and f_β have to be considered for modeling SPs' behavior.

Comparing the performance accuracy for SD and SI systems, we represent in Fig. 5.5 the average revenue rate offered by AT policies when different discretization levels h are used. The figure proves that both systems react the same way to discretization, except for some specific values of h showing very small differences in revenue due to the lower degrees of freedom of SI systems. For instance, for $N = 6$ and $h = 8$, a 1.2% difference in revenue rate can be observed between the two systems. Besides, a floor exists for R_β when a minimum number of discretization levels h is used, or, in other words, that a solution to the problem described in (5.12) can be sought in

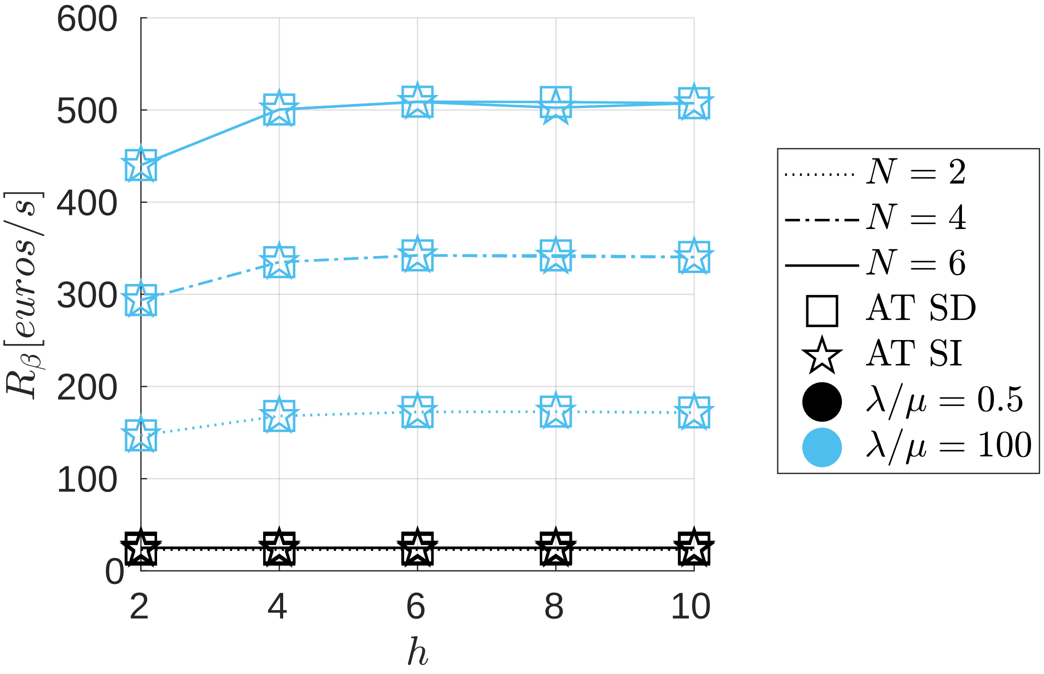
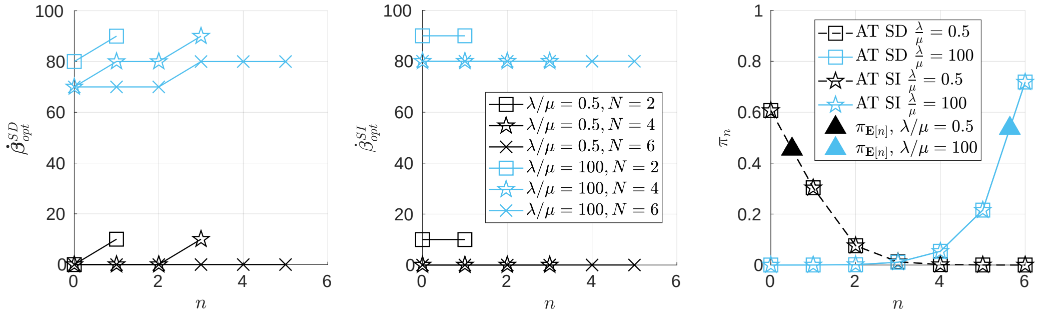


Figure 5.5: Average revenue rate for SD and SI AT policies with respect to the discretization granularity h, for different values of N, and λ/μ.

the discrete domain with no significant performance loss when a suitable accuracy is adopted. In particular, the constraint on h is approximately independent of the size of the resource pool (i.e., N), however, it is more evident in high-load regimes, as a better granularity allows a more rewarding bid selection over a bigger pool of service requests. For instance, according to Fig. 5.5, InPs may decide to apply a minimum number of discretization levels equal to $h = 2$ and $h = 4$ when $\lambda/\mu = 0.5$ and $\lambda/\mu = 100$, respectively, in order to jointly minimize complexity and the loss in revenue opportunities. However, in the following, we adopt $h = 10$ for a better graphical detail.

In order to study the behavior of SD and SI systems adopting AT policies under different load regimes and systems sizes, we represent in Fig. 5.6a and 5.6b the optimal policies for both solutions, when $h = 10$ discretization levels are used for all values of N and λ/μ. When comparing the two graphs, it can be observed that, independently of the load regime λ/μ and of the size of the resource pool N, similar AT policies are optimal for SD and SI systems. More in detail, in low-load regime (i.e., $\lambda/\mu = 0.5$), the low arrival rate of service requests and the small holding time of



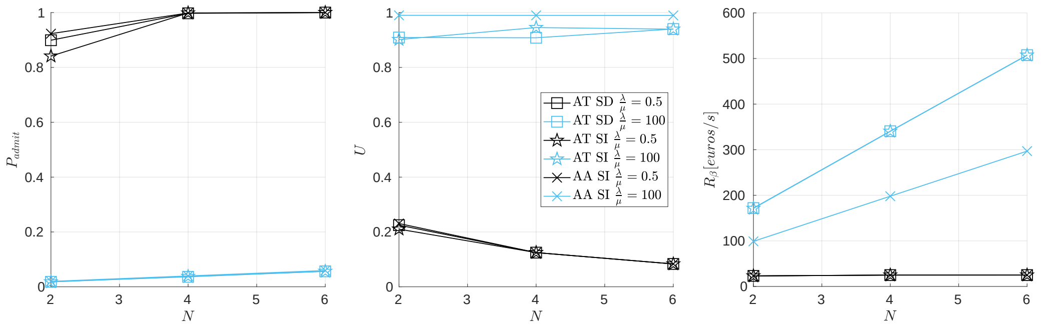
(a) (b) (c)

Figure 5.6: Optimal AT policy $\dot{\boldsymbol{\beta}}_{opt}$ in a) SD and b) SI systems with different N and λ/μ, and c) stationary probabilities π_n in SD and SI systems with $N = 6$ and different λ/μ. Besides, it is represented the interpolation of π_n corresponding to the average state $\mathbf{E}[n]$ (i.e., $\pi_{\mathbf{E}[n]}$). $h = 10$ in all the graphs.

slice tenants encourage the InPs to adopt in both systems low admission thresholds, thus, maximizing revenues by increasing the admission probability. On the other hand, in high-load regime (i.e., $\lambda/\mu = 100$), the system is saturated (i.e., $\mathbf{E}[n] \approx N$) and suffers from resource scarcity due to the high arrival rate of slice requests and the big holding time of slice tenants. Hence, InPs are motivated to increase the admission threshold in order to block the less rewarding slice requests.

In both load regimes, the higher flexibility of SD systems enables step-like policies, where lower admission thresholds are adopted when the system is far from saturation, while higher ones are employed when the system is about to exhaust its resources. Moreover, with increasing size of the resource pool N, SD systems tend to be less selective by relaxing the policy when far from saturation, in order to achieve a better balance between admission probability and revenue rate. Despite different strategies can be generally considered optimal for SD and SI systems, it can be noted that the difference in the admission thresholds adopted at each state n is, at most, equal to the discretization step (i.e., $|\dot{\beta}^{SD}_n - \dot{\beta}^{SI}_n| \leq (\beta_M - \beta_m)/h$, $n < N$). Therefore, independently of the load regime and the pool of resources, the optimal policy for the two approaches leads to the same system behavior, on average, that is, to the same stationary probabilities π_n, as illustrated by Fig. 5.6c for the case $N = 6$. This aspect, in turn, translates into a close performance matching, as demonstrated below.

After having computed the optimal admission thresholds for on-demand AT policies,



(a) (b) (c)

Figure 5.7: Performance of on-demand systems: a) admission probability, b) average resource utilization and c) average revenue rate. SD and SI AT policies with $h = 10$ and AA policies are compared.

we now compare the performance of SD and SI approaches with that of an AA policy when different load regimes and pools of resources are considered. In particular, in Fig. 5.7, we study the admission probability P_{admit}, the average revenue rate R_β and the average resource utilization U when $h = 10$ bid levels are used. Firstly, it can be observed that, by enforcing the constraint on the discretization accuracy (i.e., $h \geq 4$), a close performance match can be obtained between SD and SI approaches not only for the average revenue rate but also for the other performance metrics. This result holds independently of the load regime λ/μ and the size of the resource pool N.

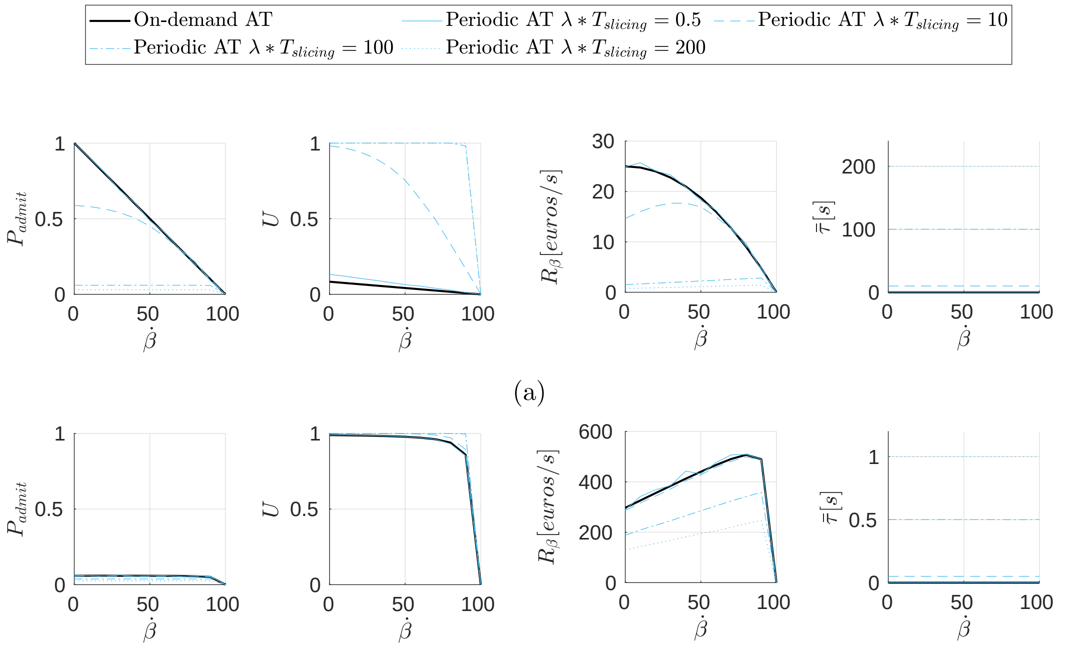
In low-load regime (i.e., $\lambda/\mu = 0.5$) it can be observed that the performance metrics of different admission strategies (i.e., AT or AA) are very close and tend to coincide when big resource pools are considered. Indeed, due to the limited revenue opportunities, AT strategies imitate the behavior of the AA approach by admitting as many requests as possible (see Fig. 5.6a and 5.6b), resulting in high admission probabilities (Fig. 5.7a). However, in resource-limited systems (i.e., $N = 2$), the higher flexibility of SD approaches is capable of guaranteeing a slightly higher admission probability when compared to SI strategies. At the same time, due to the low rate of service requests, the average number of instantiated slices (i.e., $\mathbf{E}[n]$) remains approximately constant, independently of the size of the pool of resources (i.e., N). Therefore, according to (5.6), the average resource utilization decreases with respect to N (Fig. 5.7b), while the average revenue rate does not vary (Fig. 5.7c).

In high-load regime (i.e., $\lambda/\mu = 100$), the average admission probability decreases with respect to the low-load regime for AT policies in both SD and SI systems (Fig. 5.7a). However, as results coincide with those for the AA policy, this is not the consequence of the adoption of higher admission thresholds in AT policies, but rather of the limited resources with respect to the demand. Consequently, both the admission probability and the average operational expenditures (i.e., $\mathbf{E}[n]$) increase linearly with the size of the resource pool N, as more resources can be accessed by competing SPs. Therefore, according to its definition in (5.6), the average resource utilization U remains approximately constant with respect to N (Fig. 5.7b). However, the more restrictive admission strategy of AT policies is demonstrated by a slightly lower utilization when compared to AA policy, especially for SD systems due to their greater flexibility. Likewise, because of the higher revenue opportunities, the revenue rate is higher than the one achievable in low-load regime and increases linearly with respect to the resource pool size N, as represented in Fig. 5.7c. Besides, due to the higher admission thresholds, AT policies are capable of admitting the most rewarding slice requests and consistently offer much higher revenue rates when compared to the AA strategy (i.e., 68.6% improvement).

In conclusion, AT policies provide a great advantage in terms of revenue rate and resource utilization while conserving the admission probability of less restrictive strategies, such as the AA policy. Besides, when sufficient accuracy is adopted for the bid interval discretization (i.e., $h \geq 4$), SI AT policies are reduced complexity solutions of the problem represented in (5.12) when compared to SD policies, at the expense of a slightly lower admission probability for resource-limited systems.

5.5.2.2 On-demand and periodic slicing comparison

In the remainder of this section, we first compare the performance of on-demand and periodic slicing mechanisms when AT policy is adopted. Afterwards, the comparison is extended to reference admission control strategies (i.e., the AA policy in on-demand case and the FCFS and BB policies in the periodic case). The analysis introduced in Section 6.4 can be extended to the periodic case by using discrete-time Markov chains (DTMCs), where transitions among states take place at regular time intervals. Therefore, P_{admit}, U, R_{β} and the optimal AT policy $\dot{\boldsymbol{\beta}}_{opt}$ become dependent on the slicing interval $T_{slicing}$. In this context, extending the model introduced in Section 5.3, n represents the number of slices instantiated and reserved during a given slicing



(a)

(b)

Figure 5.8: Performance of on-demand and periodic slicing with respect to the admission threshold $\dot{\beta}$ when SI AT policies are adopted and: a) $\lambda/\mu = 0.5$, b) $\lambda/\mu = 100$. $N = 6$ and $h = 10$ are considered in all the graphs, and, in the periodic case, performance metrics are estimated over $N_{slicing} = 10000$ slicing intervals.

interval, considering also those tenants that fulfilled their SLA within the considered interval (i.e., tenants leaving the system and interrupting their contribution to InPs' revenues). Therefore, the definition of U in (5.6) takes on a connotation of average resource reservation for periodic slicing, however, for the sake of comparability, we maintain same name and symbol as for on-demand slicing. As shown in previous paragraphs, both SD and SI AT strategies can be utilized for this comparison when sufficient discretization accuracy is guaranteed, thus, in the following, we consider only SI policies due to the lower complexity needed for computing the optimal policy.

In Section 5.2, we highlighted that, once policies are defined, the promptness of a specific slice admission method strictly depends on the delay added by the communication flow between SPs and InPs and the complexity for computing the admission decision. In order to provide a complete comparison between on-demand and periodic systems, we introduce in this context a new performance metric measuring the delay added by the admission control mechanism. More precisely, we define the average waiting time $\bar{\tau}$ as the average time delay from service request arrivals, up to their admission or blockage. For on-demand slicing, it holds $\bar{\tau} = 0$ because, according to Section 5.2, slice requests are evaluated right upon arrival. On the other hand, in periodic slicing, $\bar{\tau}$ is the average time interval between slice request arrivals and the beginning of next slicing interval. Therefore, exploiting the properties of Poisson processes, the instants t^a corresponding to slice requests arrivals within the *i-th* slicing interval are uniformly distributed (i.e., $t^a \sim \mathcal{U}[iT_{slicing}, (i+1)T_{slicing}]$). Hence, $\bar{\tau} = T_{slicing} - \mathbf{E}[t^a] = T_{slicing}/2$ independently of the adopted policy. With respect to the computation of the admission decision, both AA and FCFS strategies introduce null delay, as they only enforce the admission decision whenever resources are available. Assuming that the optimal admission thresholds are pre-computed for different values of λ/μ, f_β, and N, the same holds for AT policies. Finally, the BB admission mechanism implies the implementation of sorting algorithms with higher computational expenses than previous strategies, however, as better processors are made available every year, we assume that the dominant component of the total delay is $\bar{\tau}$ for all the analyzed strategies.

In order to compare how AT policies behave in on-demand and periodic strategies, we analyze how the performance metrics vary with respect to the admission threshold $\dot{\beta}$ defined in (5.13) and slicing interval $T_{slicing}$. In particular, in Fig. 5.8, we provide the representation of the admission probability P_{admit}, the average resource utilization U, revenue rate R_β, and waiting time $\bar{\tau}$ for the whole range of admission thresholds and slicing intervals defined in the system setup. On the other hand, without loss of generality, only a fixed system dimension is considered (i.e., $N = 6$). Finally, Fig. 5.8a and 5.8b illustrate the cases with low and high-load regimes (i.e., $\lambda/\mu = 0.5$ and $\lambda/\mu = 100$), respectively.

With respect to the system's fairness P_{admit} and the utilization of resources U, it can be observed from Fig. 5.8 that both are monotonically decreasing functions of $\dot{\beta}$, for every load regime and admission strategy (i.e., either on-demand or periodic). Therefore, a global maximum exists for both performance metrics over the admitted bid interval and it coincides with the most permissive threshold (i.e., $\dot{\beta} = 0$), while they tend to decrease when less permissive strategies are enforced. Besides, periodic slicing provides same performance as on-demand slicing when a small number of arrivals takes place per slicing period (i.e., $\lambda T_{slicing} = 0.5$). On the other hand, when slices are offered less frequently than the service rate (i.e., $T_{slicing} \geq 1/\mu$), the number

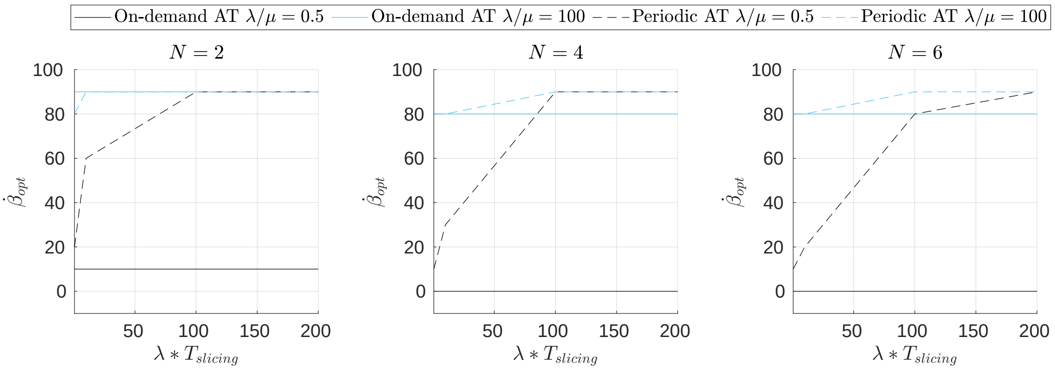
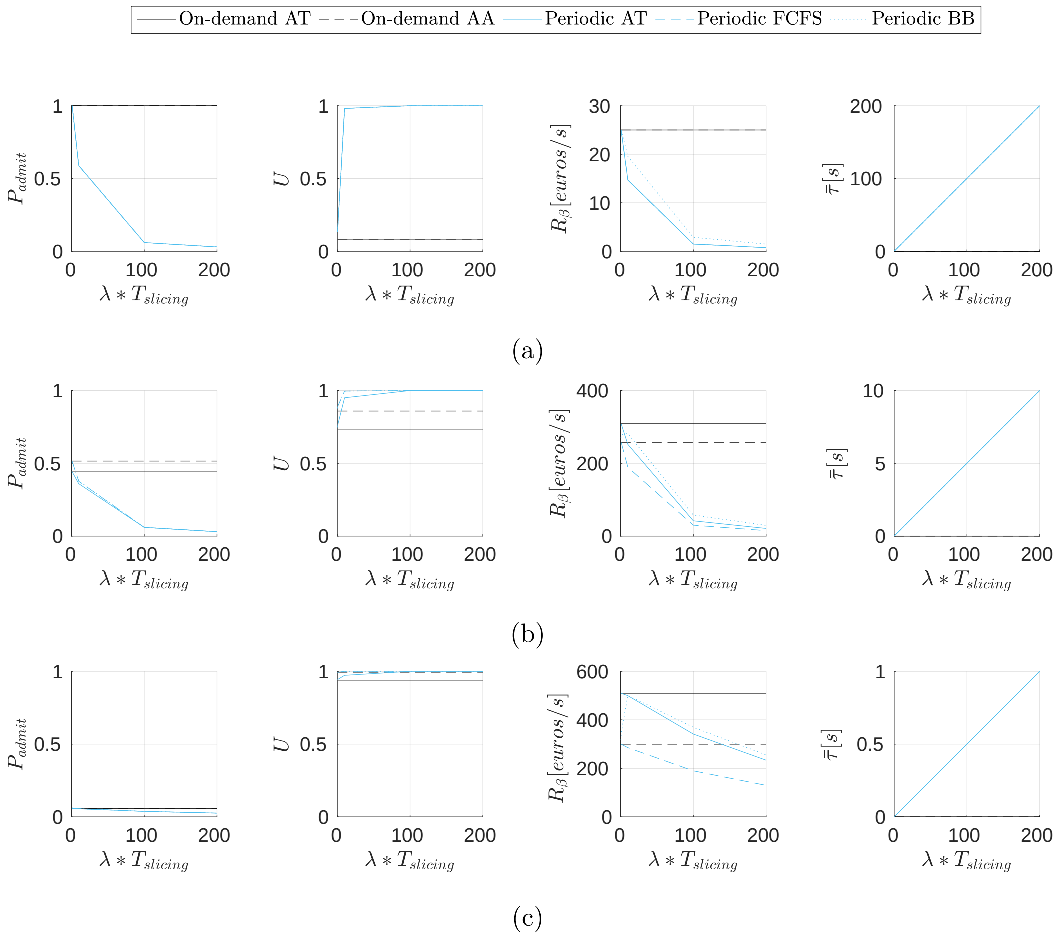


Figure 5.9: Optimal threshold $\dot{\beta}_{opt}$ for on-demand and periodic slicing when SI AT policy is used and different values of N are considered. $h = 10$ is considered in all graphs and, in the periodic case, the optimal threshold is computed over $N_{slicing} = 10000$ slicing intervals.

of SPs competing within the same slicing interval increases, and a higher optimal AT threshold is adopted. Accordingly, the admission probability decreases, and the resource reservation deviates from the resource utilization of the on-demand case. Note that for very high values of $\lambda T_{slicing}$ the level of saturation is comparable to that of on-demand slicing mechanisms in case of high-load regimes (i.e., $P_{admit} \to 0$ and $U \to 1$).

On the other hand, R_β manifests different behavior and shows a global maximum depending on the load regime and slicing strategy. When the number of competing SPs is low (i.e., $\lambda/\mu = 0.5$ in the case of on-demand slicing, joint to $T_{slicing} < 1/\mu$ for the periodic slicing case), R_β is a monotonically decreasing function of $\dot{\beta}$. As limited revenue opportunities exist, the unconditional admission (i.e., $\dot{\beta} = 0^5$) outperforms any other admission criterion. However, when the load regime increases in on-demand slicing, or when lower slicing frequencies are adopted in periodic slicing (i.e., $T_{slicing} \geq 1/\mu$), the competition among SPs increases and R_β becomes a concave function of $\dot{\beta}$. We remind that InPs have the joint objective of maximizing the admission rate and the resulting revenue, hence, when slice requests exceed the resource availability, on the one hand, revenue opportunities increase, on the other hand, the resources tend to be exhausted. Therefore, an optimal admission threshold exists as a tradeoff between the maximization of the admission rate and

[5] We highlight that, even though a null threshold is enforced, positive revenue rates are possible, on average, as SPs' behavior is modeled according to a uniform random bid distribution.



(a)

(b)

(c)

Figure 5.10: Performance metrics for network slicing with respect to $\lambda T_{slicing}$ when SI AT and AA policies are adopted for on-demand approaches and AT, FCFS and BB policies for periodic approaches. For the periodic case, the optimal threshold is calculated for each value of $\lambda T_{slicing}$, over $N_{slicing} = 50000$ slicing intervals, besides results are provided for: a) $\lambda/\mu = 0.5$, b) $\lambda/\mu = 10$ and c) $\lambda/\mu = 100$. $N = 6$ and $h = 10$ are considered in all graphs.

the prioritization of the most rewarding requests. To confirm what we just said, independently of the load regime, the horizontal coordinate that maximizes R_β corresponds to a value of P_{admit} not too far from its maximum. Besides, the optimal AT threshold also reduces U with respect to its maximum, thus limiting the operational expenditures while guaranteeing maximum revenue. Finally, it is confirmed that the average waiting time $\bar{\tau}$ is null for on-demand slicing, while it increases with respect to the slicing interval for periodic slicing (i.e., $\bar{\tau} = T_{slicing}/2$).

After having studied how the performance metrics vary with respect to the adopted threshold and to the enforced slicing interval, we analyze now the properties of the optimal AT policy for on-demand and periodic cases. In particular, in Fig. 5.9, we represent $\dot{\beta}_{opt}$ as a function of $\lambda T_{slicing}$, while considering different resource pool sizes (i.e., $N = 2$, $N = 4$, and $N = 6$), as well as low and high-load regimes (i.e., $\lambda/\mu = 0.5$ and $\lambda/\mu = 100$). First, we can observe that, for small values of $\lambda T_{slicing}$, the optimal AT policy for periodic slicing is well approximated by the one for on-demand slicing for all pools of resources and load regimes. Indeed, the high slicing frequency makes periodic slicing systems receive fewer slice requests per slicing interval, thus approximating the behavior of on-demand slicing. Besides, we can observe how, for increasing number of arrivals per slicing interval (i.e., $\lambda T_{slicing}$), the optimal AT policy for periodic slicing becomes more selective than in the on-demand case, tending to the maximum admitted threshold for every λ/μ and N.

In order to benchmark the optimal AT policy in both the on-demand and periodic cases, we compare its performance with that of reference slicing mechanisms. More in detail, in the on-demand case, we consider the AA policy that admits all slice requests, independently of the associated bids, whenever resources are available. Note that, in the case of inelastic slices only, AA coincides with the admission strategy proposed in [21]. On the other hand, in the periodic case, we study the adaptation of AA to discrete time case, which operates as a FCFS policy within a given slicing interval. Finally, for periodic slicing we also provide comparison with the BB policy that, within a given slicing interval, admits requests with highest bids up to resource exhaustion. Hence, in Fig. 5.10, we represent the admission probability P_{admit}, the average resource utilization U, the average revenue rate R_{β}, and the average waiting time $\bar{\tau}$ as a function of $\lambda T_{slicing}$. The comparison is performed over the whole range of slicing intervals according to the system setup, while, without loss of generality, only a fixed system dimension is adopted (i.e., $N = 6$). Besides, low, medium and high-load regimes (i.e., $\lambda/\mu = 0.5$, $\lambda/\mu = 10$ and $\lambda/\mu = 100$) are illustrated in Fig. 5.10a, 5.10b and 5.10c, respectively.

First, it can be observed how, in on-demand slicing, AT always outperforms AA in terms of offered revenues and resource utilization at the cost of a small loss in admission probability. Besides, AT and AA policies for on-demand slicing act as best-case scenario for their natural extensions to periodic slicing, that is, periodic AT and FCFS policies, respectively. More precisely, FCFS well approximates the AA performance for low values of $\lambda T_{slicing}$, while it provides worse performances for less frequent slicing (i.e., $T_{slicing} \geq 1/\mu$).

Observing into more detail the performances of different periodic slicing schemes, periodic AT proves to be more selective and resource efficient than the other two policies, in the sense that it is characterized by a slightly lower admission probability and by the reservation of less resources for the revenue maximization. Besides, FCFS represents the lower bound in terms of revenue rate with respect to periodic AT and BB policies. Indeed, for low values of $\lambda T_{slicing}$, BB behaves like a FCFS policy, while periodic AT improves revenues by rejecting low bids and keeping resources for future requests with higher bids. On the other hand, when sufficient service requests are received within a given slicing interval, BB outperforms the unconditional admission of FCFS and tends to the revenue rate offered by the periodic AT policy. Finally, for slicing intervals greater than one tenth of the service time (i.e., $T_{slicing} \geq 0.1/\mu$), periodic AT and BB offer comparable revenue rates. The effectiveness of the most rewarding policies (i.e., periodic AT and BB) is emphasized when high values of λ/μ are explored, that is, when more revenue opportunities exist. On the other hand, independently of the adopted policy, the admission probability decreases and the resource utilization increases inevitably due to the limited resources with respect to the demand. With respect to the average waiting time $\bar{\tau}$, it is null for on-demand strategies and for very frequent slicing (i.e., $T_{slicing} \approx 0$), while it increases linearly with $T_{slicing}$ for periodic slicing (i.e., $\bar{\tau} = T_{slicing}/2$), regardless of the analyzed mechanism.

In conclusion, a slicing system that employs the optimal AT admission policy (with respect to load regime, bid distribution and pool of resources) outperforms all the considered reference mechanisms, either on-demand or periodical. Indeed, it offers the highest revenue rate and smallest resource utilization, with a negligible loss in terms of admission rate. Besides, on-demand slicing solutions minimize the response time to slice requests.

5.6 Summary

In this chapter, we proposed a slice provision mechanism for enabling the slice market envisioned for 5G. The proposed approach consists in a policy that selects the most rewarding bids offered by SPs and exceeding a given threshold (i.e., AT policy), and a reduced complexity solution is provided for adapting the optimal policy to different resource pool sizes, traffic loads and SPs behavior. We demonstrated that our proposal enhances the slice provision promptness, with QoS guarantees and fairness towards SPs, while guaranteeing two-fold economic incentives to InPs: revenue maximization and reduction of operational expenditures. Besides, we presented a comparison of the proposal's performance with reference policies, both when enforced upon every service request (i.e., on-demand slicing) or at regular time-intervals (i.e., periodic slicing). In particular, we consider always-admit policy (i.e., AA) in on-demand slicing, and first-come-first-served (i.e., FCFS) and best bid (i.e., BB) policies in periodic slicing.

Provided that the optimal bid threshold is chosen for actual network conditions, the proposed AT policy in on-demand slicing outperforms the other considered mechanisms, including a best bid selection strategy for periodic slicing. Indeed, the optimal AT policy provided in this chapter offers the highest revenue rates while reducing operational expenditures and offering real-time slicing, in exchange for a negligible loss in terms of fairness towards SPs. On the other hand, if only periodic slicing is possible, AT policy still offers the same advantages, however, slice requests experience larger response times, regardless of the adopted policy, and decreases with the slicing frequency. Finally, the AT approach enables reduced complexity solutions when compared to other strategies, such as the BB policy. The effectiveness in terms of revenues is highlighted especially in systems characterized by limited resources and high-load regimes. Because the computation at runtime of the optimal admission strategies would result in high costs and complexity, in the next chapter, we provide an offline implementation based on machine learning and clustering approaches.

Chapter 6

Timely Slice Allocation for 5G with Machine Learning

Overview

After one decade since the first studies on next-generation networks, and a few years since early regulations and rollouts, 5G deployments are entering into a more mature phase. Indeed, if research and standardization efforts initially focused on architectures and enabling technologies, 5G ecosystem's drive is becoming progressively service-oriented. On the one hand, manufacturers and network owners are willing to fully exploit the potential of 5G's marketplace, on the other hand, regulation authorities and standardization bodies implement solutions for a healthy coexistence among parties. For guaranteeing the strict requirements foreseen for 5G, network slicing has been proposed as a dynamic and scalable mechanism for customized resource sharing among infrastructure providers and allocation to service providers. Many solutions have been proposed in the literature for the scenario where multiple service providers share the same pool of resources, while the exclusive allocation to different providers is still an open issue due to the associated complexity. In this chapter, we define a policy-based admission mechanism for exclusive intra-service slice allocation, at fine and adaptable timescales. In particular, we consider the case where optimal admission strategies are pre-computed offline for network state conditions that are representative of typical traffic loads and resource availability. This offline phase is also used to train a Machine *learning algorithm; a neural network (NN) learns the best admission policies from a more computationally expensive mechanism in previously studied network conditions. Thus, the NN is used for providing near-optimal admission decisions at runtime under network conditions for which no optimal policy has been computed. Besides, clustering-based solutions are considered for limiting the complexity associated to the pre-computation of the admission decisions over the whole network. The potential of the 5G marketplace in terms of revenue and quality of service is demonstrated for the particular case of services with strict latency constraints by means of a proof of concept implementation tested over network traces from a real network operator. Different strategies are compared for the computation of the admission strategies and results are provided in terms of efficiency in resource utilization, fairness to the service providers, network owners' revenue and complexity. This chapter confirms the feasibility of the policy-based approach defined in previous chapter for exclusive intra-service resource allocation. More precisely, a computationally-efficient mechanism for achieving near-optimal admission strategies is provided, especially in the case of missing information about network states.*

Contributions

[J3] M. Vincenzi, E. Lopez-Aguilera, and E. Garcia-Villegas, "Timely admission control for network slicing in 5G with machine learning," *IEEE Access*, vol. 9, pp. 127595–127610, 2021. (Area: Telecommunic ations; Quartile Q2; IF: 3,367).

6.1 Related works

In Chapter 5, we introduced some of the open issues related to 5G slice admission control mechanisms with strict QoS guarantees. Indeed, one of the greatest challenges lies in the definition of mechanisms for the management and orchestration of mobile network slices composed of heterogeneous resources from different infrastructures (e.g., access and core network, transport network, cloud infrastructure), while guaranteeing, among others: i) E2E QoS, ii) isolation from other tenants, iii) efficient resource utilization, and, iv) timely adaptation to traffic fluctuations in time and space [3–6,38, 138,166]. The scenario is very similar to that of cloud computing where computational, storage, and communication resources are combined in order to abstract customized virtual machines out of the same infrastructure. However, because of the scarcity and high cost of access network resources, standard over-provisioning mechanisms, typical of cloud computing, cannot be exploited for network slice allocation [182].

Two macro categories for slice allocation approaches exist, based on different InPs' business models and target services: reservation-based and share-based, respectively [182]. The first category foresees the reservation of exclusive and customized resources for different network slices, thus providing tenants with strict and stable QoS guarantees, at the cost of lower efficiency in resource utilization and higher complexity, in terms of parallel management of diverse multi-service requirements, and reconfiguration overheads. On the other hand, in share-based allocation schemes, multiple tenants coexist within a given slice according to prearranged shares, thus improving resource utilization efficiency by exploiting the statistical multiplexing of tenants' traffic across multiple slices [118], and limiting complexity by performing joint allocation and reconfiguration of network slices for multiple tenants. However, the sharing of slice resources harms tenants' isolation and provides guarantees only on a statistical basis. If fairness is naturally guaranteed in share-based approaches by fixing prearranged shares among tenants, admission control mechanisms are needed when adopting reservation-based solutions, thus leading to a possible degradation in fairness. Efficient solutions exist in the literature for share-based slice allocation, on the other hand, the high complexity associated with reservation-based mechanisms represents an open issue, as it could harm timeliness, customization and efficiency, thus preventing InPs from meeting SPs' requirements [177].

A methodology for defining a reservation-based admission strategy is provided in Chapter 5 [**J2**]. A CTMC is employed for the computation for an AT policy of the optimal admission criterion for slice requests, that is, the threshold to adopt for bids associated with incoming requests. In case of admission, bids are registered in the SLAs as the tariff per unit of time charged by the InP to tenants throughout their holding time. Slice admission control is studied both at fixed timescales (i.e., periodic) and upon each request arrival (i.e., on-demand). While on-demand approaches allow a faster response to slice requests, thus minimizing its contribution to delay, periodic admission control limits technological and complexity requirements. When sufficiently small timescales (i.e., negligible with respect to the average service time) are adopted in a way that it is suitable for short-lived services such as emergency or surveillance services [5,6], both schemes show very similar performance; hence, the interest of this chapter in the timescales used for the admission process. Finally, both SI and SD policies are studied, which foresee fixed or adaptable thresholds for different states of the CTMC. Optimal AT admission policies for specific congestion levels (i.e., the ratio between the arrival and departure rates with respect to available resources) are computed according to exhaustive search. Results show that, when optimal admission policies are computed with sufficient granularity in the search space (i.e., accuracy in the discretization of the bid interval), comparable results are provided by less complex SI solutions with respect to more accurate SD alternatives. Besides, when compared with reference approaches (i.e., on-demand AA, and, periodic FCFS and BB admission strategies), the AT strategy is capable of providing near-optimal revenues to InPs, reducing expenditures and providing a fair slice provision to competing SPs.

An alternative approach is the one described in [177], where an *online* and reduced complexity admission control policy is derived by means of reinforcement learning, which is capable of maximizing InP's revenue while reducing the penalties due to SLAs' violation (i.e., on rejection of slice requests) under different network conditions. One of the key contributions of this solution is its applicability to a scenario where slice requests are issued simultaneously over the same infrastructure for different service types (i.e., eMBB, uRLLC, and mMTC). Three possible algorithms are considered for the computation of the optimal admission policies (i.e., Q-Learning, Deep Q-Learning, and Regret Matching), and performance is assessed by means of computer simulation in terms of: i) maximization of the revenue-to-penalty ratio, and, ii) learning ability of online and offline strategies. Despite the great flexibility offered by online approaches in terms of capability of adapting to new network conditions, they are typically characterized by a reduced promptness in terms of slice provision, due to: i) the time needed by the traffic forecasting algorithm for collecting sufficient data on the network conditions, and, ii) the execution time of machine learning (ML) approaches for efficient enforcement of the admission strategy at runtime (i.e., including the learning phase) [182]. In this chapter, we adopt an offline approach in order to take prompt decisions during the slice allocation process, besides, we employ fine and adaptable timescales in order to improve timeliness, resource utilization efficiency and admission rate, contrarily to [177] and to most of the solutions in the literature that perform slice allocation at fixed timescales.

Indeed, slice allocation at timescales coinciding with the holding time specified in SLAs is generally associated with low efficiency in resource utilization, mostly if coarse timescales are used for slice provision **[J2]**, [183]. A possible approach to improve the efficiency of slice management mechanisms when fixed timescales are used consists in the implementation resource reallocation within slices [183, 184].

From a service modeling perspective, a general characterization is provided in the literature for different service types (e.g., eMBB, mMTC and uRLLC), together with studies on their coexistence and prioritization [135, 177, 185, 186]. The majority of the solutions in the literature adopt a per-SP slicing approach, foreseeing a two level resource allocation: i) per-SP slice allocation used by each tenant for serving multiple customers, ii) a lower level, per-user allocation, adopting more complex mechanisms for resource allocation within a given slice (e.g., scheduling) or across multiple slices [182]. In addition, performance is typically assessed over constant arrival and departure rates **[J2]**, [177,182], with the exception of [183,185,187], which provide results on real network traces. Finally, performance is usually provided by aggregating results from different cells, which is a reasonable strategy in order to provide a network representation. However, this approach hides the suitability of a specific approach to cells with different features (e.g., coverage, pool of resources, traffic patterns and location).

6.2 Slice Allocation with Adaptive Timescales for 5G Services

In this chapter, we make an effort to demonstrate the feasibility of a reservation-based slicing mechanism for services characterized by strict QoS requirements (e.g., uRRLC), by providing a PoC on real network traces for the reservation-based slicing presented in Chapter 5 **[J2]** and adapted for timescales suitable for 5G services. More into detail, InPs adopt the programmable infrastructure and the enhanced architecture introduced in Chapter 3 and perform *intra-service slice allocation* (i.e., slice allocation to SPs providing the same kind of service), by enforcing a policy based admission control mechanism for bid selection at fine and dynamic timescales, pursuing maximum revenues to the InPs, while improving efficiency and guaranteeing timeliness and fairness towards SPs. A *periodic* and *offline* schemes is used, which requires an initial training phase when compared with online admission control algorithms, whose computational burden is typically justified by a better performance [177]. The statistical nature of the CTMC-based scheme in Chapter 5 **[J2]** is exploited for pre-computing, during a one-time training phase, the optimal admission strategies for known states of InPs' networks (e.g., obtained from historical data), which are computed according to the following approaches: i) an exhaustive search over a limited set of network state conditions, and, ii) by using ML mechanisms for providing near-optimal admission strategies for untested network conditions.

We include the slicing timescale as one of the admission control parameters adapted by the InP. Therefore, the optimal admission strategies include both the bid admission policies and the slicing timescale. Indeed, although the highest performance in terms of customization and efficiency is achieved by adopting the smallest timescales for slicing **[J2]**, [182], we propose the adoption of slicing mechanisms with adaptive timescale with respect to network congestion. This novel approach enables: i) the limitation of the overall computational requirements without experiencing significant losses in performance, ii) congestion reduction and customization guarantees by adapting the admission strategy on-the-fly with respect to SPs' traffic fluctuations in time and space, and, iii) performance comparable to an on-demand scheme in a cost efficient manner. This strategy, combined with *edge computing*, has the potential to provide the promptest type of slice provision to services with very strict time constraints (e.g., uRLLC), while maintaining a good revenue.

The number and values of network conditions considered for the pre-computation of the admission strategies sets a tradeoff between performance and complexity. In this regard, a NN is trained with optimal decisions provided by exhaustive search (ES) for state conditions that are representative of the system. Therefore, an efficient solution is provided for extending the admission strategy to unexplored network conditions (both in time and/or space), and customization is improved in exchange for a limited complexity increase (i.e., the initial learning phase). Besides, in order to limit the complexity needed for the network-wide pre-computation of the admission strategies, cluster analysis is performed for grouping similar network cells together according to their historical congestion levels. In particular, under the assumption of a centralized network architecture, we explore the possibility of pre-computing the aptimal admission strategies for groups of cells (i.e., the clusters), instead of performing such operation separately for each network node. Finally, in this chapter, we study performance on real network traces from a real mobile operator, and we provide a comparison for urban cells of different sizes and traffic patterns.

Performance is assessed in terms of fairness to the SPs, resource utilization efficiency, and InP's revenue for AT, FCFS and BB admission strategies. A comparison is provided on network traces from a real mobile operator with respect to reference solutions applied to urban cells of different sizes and traffic patterns. In addition, as centralized architectures are being standardized for 5G networks based on SDN principles (see Chapter 3), the room for a further complexity reduction is investigated by adopting the following procedure: i) clustering cells according to available network traces, ii) obtaining the adaptable admission strategies only for a candidate cell in each cluster, and, iii) comparing the gap in performance when candidates' admission strategies are enforced to other cells within, or outside of, a given cluster. As far as is known, this is the first study considering a variable timescale for improved customization in slice provision at a reduced increase in complexity.

6.3 System Model

In this section, we present the system model considered for performance assessment of policy-based slice admission control mechanisms, performed on real network traces representing Y different network nodes. In this regard, we refer to Fig. 5.2a, where multiple UEs subscribing services offered by different SPs coexist within a given geographical area. SPs issue requests for QoS-tailored network slices (i.e., slice requests) to the InP providing coverage over the area, submitting a bid β_s for each request, while the latter takes decisions on which requests to admit. From the InP's perspective, requests from different SPs for a specific service class c are associated with: i) vector $\overline{\boldsymbol{r}}_c$, specifying resource requirements for each resource kind e, ii) average arrival rate λ_c, iii) average service (or departure) rate μ_c, and, iv) maximum waiting time τ_c accepted, from the slice request until its provision. Every time a SP is admitted in the network, it is regarded as a slice tenant with identifier s, with whom the InP stipulates a SLA containing information on the slice customization (i.e., $c = \{\overline{\boldsymbol{r}}_c, \lambda_c, \mu_c, \tau_c\}$) and the agreed tariff β_s in monetary units per second (e.g., [*euros/sec*]). Conversely, in case of rejection, requests are dropped, and no mechanism is implemented for recovery in successive allocation intervals. In Fig. 5.2a, different colors are used for identifying different SPs and corresponding UEs, SLAs and resources allocated (e.g., assigned portion of the total access link capacity C).

According to the system model in Chapter 5, $\overline{\boldsymbol{n}}(i)$ is the allocation state at the i-th slice interval, that is, the number of slices n_c currently allocated to a specific slice class c, which can be represented as a position vector $\overline{\boldsymbol{n}}(i)$ in a multi-dimensional space. Besides, for each slice class c, a sharing factor $\sigma_c^e(i)$ indicates the share over total amount of resource e allocated at interval i, resulting in resource sharing vector $\overline{\boldsymbol{\sigma}}_c(i)$. The resource pool of the InP together with the heterogeneous resource profiles of different slice classes define a feasibility region $\boldsymbol{\mathcal{F}}$ and the set of feasible allocation states such that $\overline{\boldsymbol{n}}(i) \in \boldsymbol{\mathcal{F}}$. Finally, assuming that the InP defines a policy region within $\boldsymbol{\mathcal{F}}$ that prioritizes different services by limiting the actual number of slices that can be allocated to each class (similar to [135]), c separate allocation problems can be solved for the resource allocation to heterogeneous slice classes, with a variable maximum number of slices N_c that can be allocated to each slice class at a given instant (i.e., $n_c(i+1) \leq N_c(i+1)$).

We remark that, according to the system model in Chapter 5 **[J2]**, the problem is modeled focusing on the aggregate resource demand to the InP, therefore, multiple slices can correspond to the same tenant, or even to the same UE subscribing services from one or multiple SPs. For a given service class, we assume that bids can vary between a minimum and maximum tariff: β_m^c and β_M^c, respectively. Besides, as the focus of this chapter is on the timeliness of the slice admission control process rather than on strategic bidding, we model SPs as irrational entities following a random bidding model. We represent with $T_c = 1/\mu_c$ the average *holding time* of slice class c, while we employ T_s for referring to the exact time interval during which resources are exclusively retained by a generic tenant with identifier s. In the periodic case, we remark the difference between the holding time T_s of a generic s-th tenant and the timescale $T_{i,c}^{slicing}$ adopted by the InP for periodic slice allocation to service class c. More precisely, T_s is the exact holding time for a generic slice tenant s, during which the agreed tariff is applied if the SLA is respected (i.e., the total price paid equals $\beta_s T_s$). On the other hand, $T_{i,c}^{slicing}$ is the length of the time interval during which InP collects slice requests for service class c, which will be admitted or rejected at the beginning of the following allocation interval. We assume that $T_{min}^{slicing}$ is the minimum timescale offered by InP to SPs in order to keep complexity and overhead costs limited. A possible instance of the slice allocation process is proposed in Fig. 5.2b for the case with a single service class.

From the slice allocation mechanism's perspective, time is a discrete variable represented as a sequence of Ψ slice intervals $\{T_{i,c}^{slicing}\}_{i=1,\dots,\Psi}$. In order to account for InP's capability to timely adapt the slicing timescale as a part of the admission strategy, we adopt the following representation for the initial time instant of the i-th interval of service class c: $t_{i,c}^{0} = t^{0} + \sum_{\zeta=1}^{i} T_{\zeta-1,c}^{slicing}$, with t^{0} and $T_{0,c}^{slicing}$ representing, respectively, the first time instant observed, and the first interval for slice request collection. For a specific slice class c, we represent the ρ_i^c slice requests received within the i-th interval with $\{s_{i,q}^{c}\}_{q=1,\dots,\rho_i^c}$, disposed in order of arrival according to index q. Assuming that the average arrival rate varies in time, thus identifying periods with higher or lower load in terms of traffic, it holds $\mathbf{E}[\rho_i^c] = \lambda_c(i)T_i^{slicing}$. On the other hand, we assume that departure rates for a given service class do not vary with time. Similar to the arrival rate, we assume that resource requirements $r_c(i)$ can also vary with time, thus accounting for QoS customization within a specific service class. Therefore, every slice allocated for the i-th interval deduces an amount $r_c(i)$ from the resource pool until departure.

The *admission policies* $\mathcal{P}_i^c$ that InPs can enforce for a specific service class c at the end of allocation interval i, are defined below for the sequence $\{\beta_{s_{i,q}^c}\}_{q=1,\dots,\rho_i^c}$ of bids received within the i-th slice interval. We represent with $n_c^a(i)$ the number of slice requests admitted at the beginning of current slice interval, and we remark that policies enforced at the next interval $i+1$ depend on the maximum number of slices $N_c(i+1)$ that can be allocated to class c according to the policy region defined by the InP.

6.3.1 First-Come-First-Served and Best Bid

FCFS and BB represent two antithetical admission strategies in terms of fairness towards SPs and revenue to InP because, although they both maximize the number of admissions by allowing resource exhaustion, the former admits requests according to the order of arrival (i.e., independently of the associated bids), while the latter orders requests from the highest to the lowest bid (i.e., prioritizes SPs with highest spending power). In other words, FCFS applies the policy described below to incoming bids $\{\beta_{s_{i,q}^c}\}$ for increasing values of index q. On the other hand, BB first sorts bids values from the greatest to the smallest according to a new listing index $\hat{q}$, then, it applies the policy described below to $\{\beta_{s^c_{i,\hat{q}}}\}$ for increasing values of index $\hat{q}$.

$$\begin{aligned} \mathcal{P}^c_i(\beta^c_{s_{i,q}}) &= \begin{cases} \textit{Admit}, & \text{if } n_c(i+1) \leq N_c(i+1) \\ \textit{Reject}, & \text{otherwise} \end{cases} \\ \textit{FCFS: } & \{s^c_{i,q}\}_{q=1,\dots,\rho^c_i} \\ \textit{BB: } & \{s^c_{i,\hat{q}}\}_{\hat{q}|\beta_{s^c_{i,\hat{q}}} \geq \beta_{s^c_{i,\hat{q}+1}}} \end{aligned} \tag{6.1}$$

6.3.2 Above Threshold

AT strategy represents a tradeoff between FCFS and BB solutions in terms of fairness and revenue to the InP. Indeed, similarly to the FCFS approach, slice requests are admitted in order of arrival, but only if associated bids are above a specific threshold $\dot{\beta}^c_i$, which can be set by the InP to any value within the interval $[\beta^c_m, \beta^c_M]$ based on the congestion level of the network. In other words, on the one hand, it enforces a more conservative strategy in terms of resource utilization and, on the other hand, it can pursue the maximization of InP's revenue by choosing a suitable admission thresholds, or it can favour fairness by adopting thresholds closer to $\dot{\beta}^c_i = \beta^c_m$ (i.e., tending to a FCFS strategy). Consequently, AT applies to incoming bids $\{\beta_{s^c_{i,q}}\}$ the policy described below for increasing values of index q.

$$\begin{aligned} \mathcal{P}^c_i(\beta^c_{s_{i,q}}) &= \begin{cases} \textit{Admit}, & \text{if } \beta_{s^c_{i,q}} \geq \dot{\beta}^c_i \wedge n_c(i+1) \leq N_c(i+1) \\ \textit{Reject}, & \text{otherwise} \end{cases} \\ & \{s^c_{i,q}\}_{q=1,\dots,\rho^c_i} \end{aligned} \tag{6.2}$$

6.4 System Analysis for Optimal and ML-based solutions

In this section, we first introduce the metrics used for performance assessment, then we adapt and study the optimization problem introduced in Chapter 5 **[J2]** for offline pre-computation of optimal admission strategies. In particular, the approach proposed in Chapter 5 **[J2]** has to be implemented in parallel for each of the c service classes supported by the InP. However, as introduced in Section 6.2, in this context the focus is on the timeliness of an admission control mechanism suitable for slice classes with strict requirements in terms of latency (e.g., short-lived uRLLC). Therefore, rather than studying the resource allocation and slice provision to different service classes, we study and provide performance results for the slice provision to SPs belonging to a specific service class (i.e., sub/superscript c is omitted in the following).

In Chapter 5 [**J2**], we assessed the lower delays offered by on-demand slice admission schemes with respect to a periodic approach, however, as explained in Section 6.1, performing slice admission control at fixed timescales allows reducing technological and complexity requirements. Besides, as studied in Section 5.5.2.2, the periodic scheme approaches on-demand scheme's performance when sufficiently small timescales are adopted. For these reasons, in order to achieve a reasonable tradeoff between complexity and promptness of the slice provision mechanisms, on the one hand, we adopt the periodic scheme with the lowest complexity (i.e., enforcing the SI policies introduced in Section 5.4.2 [**J2**]) and we perform offline pre-computation of the optimal admission strategies and, on the other hand, we enforce the admission policies at adaptable timescales $T_i^{slicing}$ according to network congestion level's fluctuations in time and space.

With respect to the resource profile associated to this specific service class, we study a simplified model where only access network resources are considered for slice allocation (i.e., channel capacity C of the access link) because, due to their scarcity, they are the most valuable asset in the slice marketplace (see Section 6.2) and, therefore, they represent the bottleneck in the E2E slice provision [**J2**], [177]. For this specific case, the number $n(i)$ of slices in the system at i-th instant can take values between zero and $N(i) = \lfloor \sigma(i)C/r(i) \rfloor$ (sub/superscript e is omitted as only one resource kind is considered). We remark that $r(i)$ only depends on the resource requirements of the considered slice class, while $\sigma(i)$ is obtained from the policy region defined by the InP and depends on the allocation state $\overline{\boldsymbol{n}}(i)$.

We consider the highest levels of customization and isolation, that is, per-user slice allocation. This choice is due to two main reasons: i) we consider only access network resources, therefore, it is possible to enforce slice allocations by means of scheduling algorithms, thus removing the complexity deriving from a two-level resource allocation, and, ii) we want to provide a PoC for short-lived uRLLC services expecting timely slice allocations, avoiding the delays related to the aggregation of slice requests coming from multiple users.

Finally, we assume that slice requests arrivals can be modeled as a Poisson stochastic process with average rate $\lambda(i)$, and SPs' departures as a general stochastic process with average rate μ. With respect to the SPs' bidding strategy, we assume that bids β_s can be modeled as a random variable following a general distribution f_β over the sample space $[\beta_m, \beta_M]$.

6.4.1 Performance metrics

The analytical definitions provided in Chapter 5 **[J2]** for the performance metrics of the on-demand case can be easily adapted to the periodic case and expressed as a function of the system model's variables introduced in Section 6.3. More in detail, assuming that the *admission strategy* ξ_i enforced at the end of slice interval i can be fully described by tuple $(\mathcal{P}_i, T_{i+1}^{slicing})$, we represent with $A_{i+1} = n^a(i+1)/\rho_i$ the *admission ratio* at the next slice interval, expressed as the ratio between slice requests admitted and total number of arrivals. Until its departure, an admitted slice s (received at slice interval i) implies a decrease of r_i from the available capacity C at slice interval $i+1$, and a contribution to InP's revenue equal to $\beta_s T_s$ (paid proportionally at each of the following slice intervals). Consequently, for a specific service class, if we represent with C_i^{av} the portion of network capacity available out of $\sigma(i)C$ at slice interval i, we define the *percentage of resource utilization* of the service class as $U_i = 1 - C_i^{av}/(\sigma(i)C)$. Finally, if R_i^{tot} represents the aggregate revenue paid by all tenants at a specific slice interval i, we can compute the total *revenue rate* as $R_i^{tot}/T_i^{slicing}$. An average or aggregate version of the same metrics is also provided over the whole observed time interval. In particular, the *average admission rate* $\bar{A}$, the *average percentage of resource utilization* $\bar{U}$, and *the average admitted bid* $\bar{\beta}_s$ are computed averaging over the Ψ slice intervals considered. On the other hand, the *total aggregate revenue* is provided as $R^{tot} = \sum_{i=1}^{\Psi} R_i^{tot}$. Finally, as a measure for the timeliness of the slice admission control method, we employ the average waiting time from the moment a slice request is received, until an admission decision is made[1], that is, $\bar{\tau}_i = T_i^{slicing}/2$, which has to be lower than the maximum

[1] According to Section 5.5.2.2 **[J2]**, the properties of Poisson processes can be exploited for computing the average value for the arrival instant t_i^a within the i-th slice interval, that is, $\mathbf{E}[t_i^a] =$

timescale τ accepted by SPs to meet latency requirements for the slice allocation.

6.4.2 Optimal Strategy and Complexity

For the pre-computation of the optimal admission strategies at specific network conditions, we follow the approach presented in Section 5.4.3 **[J2]**, which aims at the maximization of InP's revenue rate. The maximization problem extended to the periodic and adaptive case can be defined as follows:

$$\begin{aligned} \xi_{\nu}^{opt} &= \arg\max_{\xi} R_{\beta}\Big(\nu, f_{\beta}, \xi\Big) \\ \textit{FCFS, BB: } \xi &\equiv T^{slicing} \\ \textit{AT: } \xi &\equiv (\dot{\beta}, T^{slicing}) \\ T^{slicing} &\in [T^{slicing}_{min}, \tau] \\ \dot{\beta} &\in [\beta_m, \beta_M] \end{aligned} \tag{6.3}$$

where R_{β}, represents the revenue rate that InP would obtain in the long term by enforcing a given admission strategy ξ over a network node with state condition $\nu = (\lambda/\mu, N)$. We remind from the system model presented in Section 6.3 that the triple $(\lambda/\mu, r, f_{\beta})$ represents SPs' model, in terms of traffic load, resource requirements and bidding behavior. On the other hand, N is a measure of the maximum resource availability at a specific network location with respect to SPs' requirements at a specific time instant. Finally, ξ represents test strategies in the search space for the considered continuous optimization problem, which is mono-dimensional in the case of FCFS and BB strategies, where only the slicing timescale $T^{slicing}$ can be tuned, and bi-dimensional in the case of AT approach, where we can configure both slicing timescale and admission threshold for incoming bids β_s. Because we implement an offline strategy for the pre-computation of the admission policies, the optimization process is performed only once and its outcome can be used for building a lookup table that will be used on-the-fly for different network nodes and time instants. Justified by the computational power of current technologies, we explore in this chapter the exhaustive search of the optimal strategies, and we compare its performance with more computationally-efficient and flexible methods based on ML.

In order to limit the complexity of the offline pre-computation of optimal admission strategies, we discretize the search space independently over its dimensions, transforming

$T_i^{slicing}/2$. Then, $\bar{\tau}_i = T_i^{slicing} - \mathbf{E}[t_i^a]$.

the problem in (6.3) into a combinatorial optimization problem. More into detail, we assume that InP can arbitrarily choose for $T^{slicing}$ a finite number l of sample values in $[T^{slicing}_{min}, \tau]$. On the other hand, for bid selection in the AT strategy, a finite number h of admission thresholds $\dot{\beta}$ is selected uniformly within the sample space $[\beta_m, \beta_M]$, as defined in Section 5.4.3 **[J2]**, that is, $\dot{\beta} = \beta_m + j(\beta_M - \beta_m)/h, \quad j = 0, \ldots, h-1$. A sufficient value needs to be adopted for h in order to guarantee optimal performance to InPs and SPs while keeping computational costs limited, as studied in Section 5.5 **[J2]**. In conclusion, the candidate admission strategies ξ are defined over a space $\mathcal{W}$ of cardinality $|\mathcal{W}| = l$, or $|\mathcal{W}| = l \cdot h$ in FCFS and BB case, or in AT case, respectively. The discretization of the search space could lead to the curse of dimensionality, where a higher number of sample values is translated into increased complexity, although not necessarily associated with a better statistical significance. Therefore, the particular choice of the sample admission strategies (considering both cardinality $|\mathcal{W}|$ and selected values) could lead to very different performance and, in general, the adoption of a decomposition algorithm is recommended for the discretization of the sample space according to its most representative features. However, in this chapter, we decide to limit complexity by choosing few sample strategies, while relying on the NN for extending the admission strategies to unexplored regions of the sample space. Indeed, the NN is trained by using the input-target pairs (ν, ξ^{opt}_{ν}) for providing near-optimal strategies ξ in correspondence of generic state conditions ν.

As remarked in Chapter 5 **[J2]**, the InP is responsible for pre-computing convenient strategies in correspondence of network conditions that are representative of real SPs' behavior and resource availability at different nodes of the network. Therefore, InP has to properly choose the tuples ν over the discrete sample set $\mathcal{V}$ to be used for the offline solution of problem in (6.3). In order to limit complexity while improving the versatility of pre-computed strategies, rather than solving the optimization problem separately for all possible conditions of different nodes at different times, we select sample state conditions that are statistically representative of the whole network (e.g., observing historical data gathered from different locations and time instants). Strategies need to be employed for mitigating the curse of dimensionality, which could lead to the overfitting of the neural network if the input state conditions selected for the initial training do not have statistical significance for all the network nodes in different hours. In this case, we first compute the union $\mathcal{V}'$ of all the tuples $(\lambda/\mu, N)$ obtained from network traces over different nodes' location. Afterwards, we perform an initial coarse and homogeneous sampling over $\mathcal{V}'$ and, finally, we run a fine scale sampling over the most occurring tuples.

We remark that, in the case of BB admission strategy[2], by definition, additional burden is required at runtime for the ordering of incoming slice requests with respect to bid values, when compared with FCFS and AT approaches. In particular, assuming that a quicksort algorithm is used, the average complexity associated with the BB's bid selection at the end of slice interval i is $\mathcal{O}(\rho_i)$.

6.4.2.1 Exhaustive search

The complexity of the offline pre-computation for a specific network condition by means of exhaustive search is linear with respect to the cardinality $|\mathcal{W}|$ of the search space for ξ (i.e., $\mathcal{O}(|\mathcal{W}|)$). The overall time required for the admission strategies' pre-computation strictly depends on the number of samples states considered (i.e., on the cardinality $|\mathcal{V}|$ of $\mathcal{V}$), which also determines the complexity of implementing the lookup table at runtime. More in detail, for an arbitrary network condition ν_i at a specific node location and time interval i, we enforce the admission strategy ξ_i^{opt} corresponding to the tuple ν in $\mathcal{V}$ that minimizes the squared euclidean distance $d(\nu_i, \nu)^2$. Assuming that the minimization is performed by implementing the quicksort algorithm over the squared euclidean distances plus the selection of the smallest value, the average complexity is $\mathcal{O}(|\mathcal{V}|)$ independently of the admission strategy considered.

InP needs to implement the runtime process described above in parallel for all the Y network nodes, thus, with a network complexity at runtime equal to $\mathcal{O}(Y \cdot |\mathcal{V}|)$. As a possible solution for the reduction of the complexity over the network, we consider the approach introduced in Section 6.2, that is, performing offline clustering of network nodes according to historical data, and applying optimal admission strategies of few candidates (i.e., nodes corresponding to clusters' centroids) to the rest of the nodes in the network. In particular, we perform clustering according to k-means implementation [188], that partitions Y nodes into k clusters based on δ-dimensional features extracted from network traces, while considering as objective function the global minimization of the squared euclidean distance to the clusters' centroid.

Although clustering requires an increase in the overall computational complexity, this process is performed only once offline, in exchange for a complexity reduction at

[2] BB is the most greedy and unfair strategy from the InP's and SPs' perspectives, respectively.

runtime by a factor k/Y, which is the dominant component of adopting a policy-based solution on the long-term. However, in scenarios where the network is expected to experience drastic changes, clustering can be repeated according to a given periodicity in order to maintain an updated and accurate representation of clusters and centroids that fits the network.

6.4.2.2 ML-based search

As detailed in the previous subsection, the exhaustive search approach is used to generate a discrete solution set for different network conditions. This solution set is then used to train a neural network, which will be capable of providing effective strategies for new network conditions, not previously explored by the exhaustive search (i.e., $\nu \notin \mathcal{V}$).

NNs are computing systems designed to model biological neural networks embedded in animal brains, which are composed by simple processing units (i.e., the neurons) and dense interconnections (i.e., the synapses), which, all together, allow for building up knowledge by means of experiential learning [189]. Inspired by biological neural networks, artificial NNs (or simply NNs) are composed by multiple layers of processing units, representing populations of neurons, each layer transforming a set of inputs into output signals. Neurons within a layer are interconnected by multiple links and the output of each neuron is obtained by performing non-linear operations on the weighted sum of its inputs. The NN as a whole is then composed by a sequence of layers, possibly with different number of inputs and outputs, and such that one layer's outputs constitute the following layer's inputs. Therefore, NN's input signals are transformed succesively from the input layer (i.e., the first layer) to the output layer (i.e. the last layer) going through multiple inner layers. A learning algorithm is then employed for the training of the NN, that is, for the modification of the synaptic weights among neurons within a layer and between different layers according to the objective functionality of the NN. More precisely, in a supervised learning approach, inputs are transformed into a feature vector descriptive of the input and, for each input vector, a target output vector is provided. For each couple of input features and target outputs, NN's prediction error is computed, that is, the difference between the computed and target outputs, which is then used to adjust, according to a specific learning rule, the NN's weights to be used in following iterations. After a sufficient number of iterations, where different couples of input samples and target outputs are used, the learning algorithm is terminated and the NN can be used for computing outputs also in case of inputs that have not been explored during the training phase.

This computational approach fits very well to our study case, indeeed, InPs dispose of full access to the network's information, therefore, sufficient sample network conditions (i.e., $\nu \in \mathcal{V}$) can be collected as input features for the training process. Besides, the corresponding optimal strategies ξ_{ν}^{opt} can be computed by means of ES and used as target outputs for the training of the NN. The outputs of the trained NN represent a near-optimal solution of the problem in (6.3) and are used for enforcing admission strategies at runtime in correspondence of untested conditions. We remind that the NN can be applied to any node in the network, because it is trained with sample conditions that statistically represent behaviors that could be observed throughout the whole network.

For optimizing the NN training, we perform K-fold cross-validation [190] over the following hyperparameters: i) number of hidden layers n_{HL}, ii) number of neurons per layer s_{HL}, and, iii) training function. Therefore, we divide the sample network conditions $\mathcal{V}$ into K groups, then, for each configuration of hyperparameters, we use $K-1$ groups (i.e., the $(K-1)/K\%$ of the sample set) for training the NN, and the remaining group (i.e., the $1/K\%$ of the sample set) for validating how close the estimated strategies are to the target ones for the considered configurations of hyperparameters. This process is repeated K times such that each subset is used exactly once for validation. In the case of AT strategies, we compare the option where a single NN is used for computing both admission threshold and timescale, with the alternative approach where two parallel NNs are used for computing separately $\dot{\beta}_i$, and $T_i^{slicing}$.

The complexity corresponding to the training phase of a NN depends on all the parameters introduced above, in addition to the stop criterion adopted. Finding a strict definition is out of the scope of this chapter because, similarly to the case of clustering, the training of the NN is performed offline only once. Besides, justified by the computational power offered by existing technologies, we neglect the corresponding increase in the overall complexity count. On the other hand, the enforcement of NN-based admission strategies at runtime at a specific node location and time interval

i requires linear algebraic operations over the input network condition ν_i, whose complexity depends only on the NN's topology, that is, $\mathcal{O}(n_{HL} log(s_{HL}))$ [191]. In this chapter, we consider NN with reduced topology, therefore, the corresponding computational burden at runtime is expected to be lower when compared to the implementation of a lookup table over the pre-computed $|\mathcal{V}|$ admission strategies as described above (see Section 6.5.1).

In Table 6.1, we summarize the notations used for the main parameters introduced for the system model and system analysis in Sections 6.3 and 6.4, and that will be defined for results and discussion in Section 6.5.

6.5 Results and Discussion

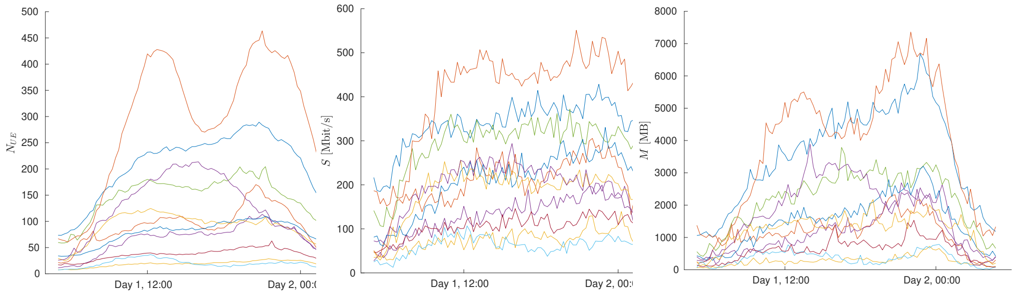
In this section, we first describe the system setup, then we compare the performance obtained when different admission strategies are adopted. Finally, we study the case where ML strategies are employed for efficient computation of admission strategies, as well as the possible reduction in complexity offered by the offline clustering of network nodes.

6.5.1 System Setup

For the performance assessment, we consider the system setup described in the following. SPs slice request arrivals are realized according to a Poisson distribution with average arrival rate λ extracted from network traces, as explained below. On the other hand, for departures we consider an exponential distribution, with average service rate $\mu = 1/60$ set according to the upper limit on the holding time at link layer provided in [192]. SPs' bids follow a uniform distribution within the range $[\beta_m, \beta_M] = [0, 100]$. Finally, both channel capacity C of the access link and resource requirements r are extracted from network traces as explained next.

6.5.1.1 Network traces

Network traces are provided by a mobile operator for a 4G network operating in an European city over a time interval of one week for eleven network nodes (i.e., $Y = 11$)



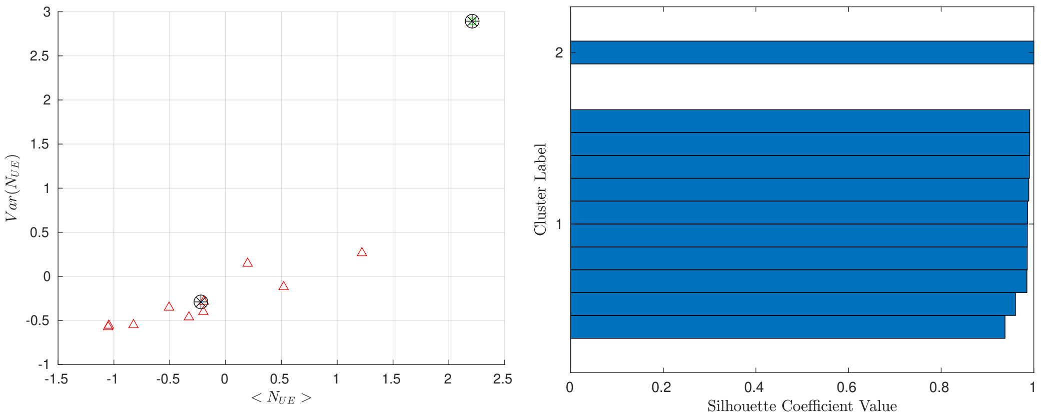
(a) (b) (c)

Figure 6.1: Daily averages of network traces from a real mobile operator from 5AM to 4:45AM of the next day: a) average number of active UEs N_{UE}, b) maximum throughput S of network nodes in downlink, and, c) aggregate data M sent to UEs in downlink. Different colors are used for different cells.

at a regular periodicity, with trace intervals of size $T_{trace} = 900[s]$. For each network node, information is provided on the average number N_{UE} of active UEs, maximum throughput and aggregate amount of data exchanged with UEs. In the following, and without loss of generality, we only consider downlink resources. We represent with S the maximum throughput in $[Mbit/s]$ considering all UEs, and with M the total amount of data in $[MB]$ sent by the network node. In Figure 6.1, we provide the daily averages computed over the network traces, which clearly show that different nodes support diverse volumes of traffic, although with similar patterns, as it will be studied in detail in Section 6.5.2.

6.5.1.2 Clustering

We perform k-means clustering on the Y network nodes by considering different values of the number of clusters k and different combinations of δ-dimensional parameters from traces. The maximum number of iterations is set to $\gamma = 100$, and the algorithm is run ten times with random initial centroids in the attempt to filter out the dependence on the starting point. The highest separation in terms of squared euclidean distance between clusters is provided when $k = 2$ is used, and when clustering is performed over the average and the variance of N_{UE} computed over the week (i.e., $\delta = 2$), respectively, $< N_{UE} >$ and $Var(N_{UE})$. This result shows a high correlation between N_{UE}, S and M. Resulting clusters are represented with different colors in Figure 6.2a, with triangle and circle star markers representing, respectively, network nodes and



(a)

(b)

Figure 6.2: k-means clustering on network nodes' traces with $k = 2$: a) clustering with respect to average and variance of N_{UE} over the week, and, b) silhouette plot. In a) coordinates are normalized with respect to mean and standard deviation computed over the network. Besides, nodes of cluster 1 and 2 are represented in red and green triangles, respectively, while a circle star marker is used for the centroids within each cluster.

geometrical centroids for each cluster. In the following, we consider as centroids the network nodes in each cluster that minimize the squared euclidean distance to the geometrical centroid (i.e., *Centr1* and *Centr2*). Note that the coordinates for each node are normalized with respect to the network's mean value and standard deviation. We can conclude from Figure 6.2, that one particular cell in the studied data set shows a very unique behavior and is therefore isolated in its own cluster, perhaps corresponding to the macro cell over the considered geographical area. For cluster 1, we also compute over its nodes the average characterization in terms of $< N_{UE} >$ and $Var(N_{UE})$ and we identify the network node that minimizes the squared euclidean distance to this coordinate (i.e., *avNode1*). Besides, we represent in Figure 6.2b the values of the silhouette coefficients for each network node, representing the similarity of nodes within a cluster, with respect to those in the other cluster. With a mean silhouette value in cluster 1 equal to 0.98 we are sure that a good similarity is achieved among nodes in that cluster, as well as an excellent separation with respect to *Centr2*.

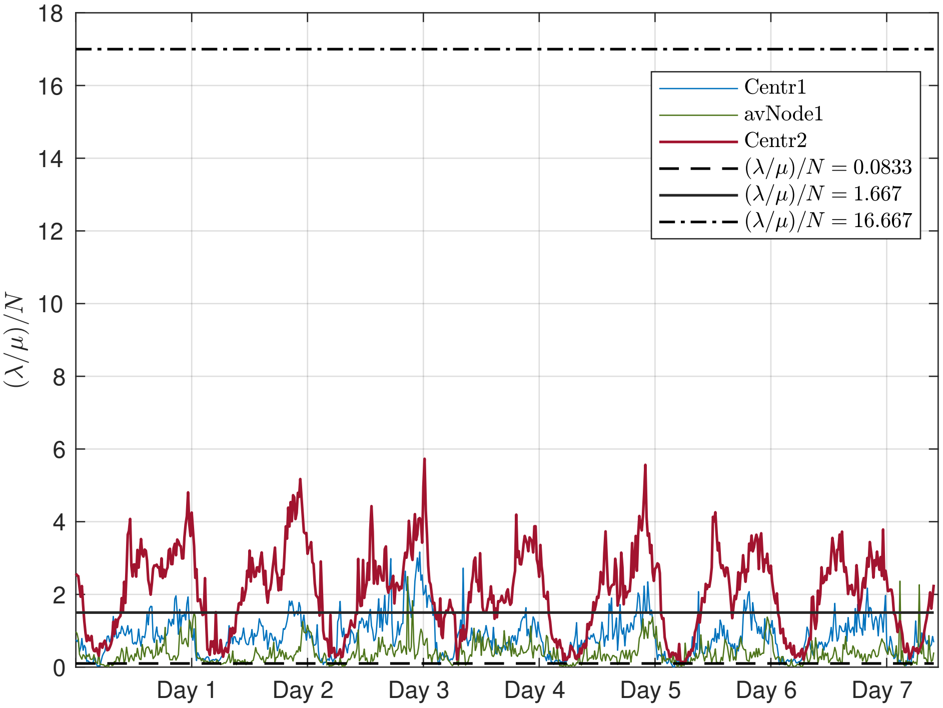


Figure 6.3: Values of $(\lambda/\mu)/N$ over the weekly traces in colored lines for *Centr1* and *Centr2*, as well as for *avNode1*. As a reference, the levels of congestion studied in Chapter 5 [**J2**] are also represented in dashed lines.

6.5.1.3 Adaptation of network traces to the system model

In order to adapt 4G network traces to network conditions that take into account the high traffic demands expected for 5G networks, we introduce a scaling factor $\alpha = 40$ such that $\lambda/\mu = \alpha N_{UE}$. Besides, for the resources available at a specific node's location, we assume that the channel capacity of the access link can be approximated by $C = \max_t S(t)$. Finally, SPs' resource requirements in $[Mbit/s]$ at a specific slice interval are computed as $r = 8 \cdot M/(N_{UE} \cdot T_{trace})$. For simplicity and without lack of generality we assume for access network resources a sharing factor $\sigma(i) = 1$ (i.e., $N(i) = \lfloor C/r(i) \rfloor$ and all the capacity reserved for the considered service class). Indeed, as introduced in Section 6.4, the focus of this chapter is in the slice provision to SPs of the same service class, rather than between different service classes. However, the study of the adaptability of this approach to variable levels of resource availability is still guaranteed by the fluctuations of the resource requirements in time, according to the model defined above.

Table 6.1: Table of notations for Timely Slice Allocation in 5G.[3]

6.3. System Model	
InP	
Variable	Definition
Y	# network nodes
i	Slice interval identifier from 1 up to Ψ
$t^0_{i,c}$	Initial instant of i-th slice interval for service class c
$T^{slicing}_{i,c}$	Duration of i-th slice interval for service class c
$T^{slicing}_{min}$	Min. timescale supported by InP for slicing
$n_c(i)$	# slices allocated to service class c at interval i
$\overline{\boldsymbol{n}}(i)$	Vector of slice allocation to each service class at i
$\boldsymbol{\mathcal{F}}$	Feasibility region for heterogeneous slices allocation
$N_c(i)$	Maximum # of slices for class c at interval i
ρ^c_i	# slice requests received for class c at interval i
$s^c_{i,q}$	Identifier of slice request q at interval i for class c
$\mathcal{P}^c_i$	Admission policy applied at the end of interval i for c
$\dot{\beta}^c_i$	Admission threshold applied by AT at the end of i for c
$n^a_c(i)$	# slice requests admitted at the beginning of i for c
SPs	
Variable	Definition
c	Identifier of supported service classes
$r^e_c(i)$	Requirements of class c at interval i for resource e
$\overline{\boldsymbol{r}}_c(i)$	Resource requirements vector for class c at interval i
$\lambda_c(i)$	Average arrival rate for class c at interval i
μ_c	Average departure rate for class c
τ_c	Max. waiting time (request to allocation) for class c
T_s	Exclusive holding time for a generic tenant s
T_c	Average holding time for class c
β^c_m, β^c_M	Min./Max. tariff for class c
$\beta_{s^c_{i,q}}$	Bid associated with slice request $s^c_{i,q}$
$\sigma^e_c(i)$	Sharing factor for class c at interval i over resource e
$\overline{\boldsymbol{\sigma}}_c(i)$	Resource sharing vector for class c at interval i

[3] Sub/superscripts c, i, and q are omitted when a generic service class, slice interval, and/or slice request are considered.

6.4. System Analysis (for a generic class c)	
Variable	Definition
$\nu = (\lambda/\mu, N)$	State condition at a generic instant
$\mathcal{V}$	Set of test state conditions ν for strategy pre-computation
$\mathcal{V}'$	Union of ν from all network nodes' locations
$\xi_i = (\mathcal{P}_i, T_{i+1}^{slicing})$	Admission strategy applied by InP at the end of interval i
$\mathcal{W}$	Search space for admission strategies ξ
ξ_ν^{opt}	Optimal admission strategy for state condition ν
l, h	# values explored for $T^{slicing}$ and $\dot{\beta}$
f_β	Probability density function of β_s
A_i	Admission ratio at the beginning of interval i
$\bar{A}$	Average admission rate
C_i^{av}	Portion of network capacity available at interval i
U_i	Percentage of resource utilization at interval i
$\bar{U}$	Average percentage of resource utilization
R_i^{tot}	Aggregate revenue by all tenants at interval i
R^{tot}	Total aggregate revenue
R_β	Long term revenue rate for specific ν and ξ
$\bar{\beta}_s$	Average admitted bid
$\bar{\tau}_i$	Average waiting time from request until allocation
k, δ	# clusters and features considered for nodes clustering
K	Coefficient for crossfold validation of the NN
n_{HL}, s_{HL}	# hidden layers and neurons/layer considered for the NN
6.5. Results Evaluation	
Variable	Definition
T_{trace}	Network trace interval
N_{UE}	Average # of active UEs for a specific node
S	Maximum throughput of a given node considering all UEs
M	Total amount of data sent through a network node
γ	Maximum # iterations for clustering
α	Scaling factor for adapting 4G traces to 5G requirements

In Figure 6.3, we compare the levels of congestion[4] $(\lambda/\mu)/N$, for network conditions

[4] The average traffic load λ/μ with respect to the maximum number of available slices N.

corresponding to traces of clusters 1 and 2, with respect to the values studied in Chapter 5 [**J2**]. In particular, we represent in colored lines the values of $(\lambda/\mu)/N$ when $\alpha = 40$, specifically, for centroids of cluster 1 and 2, as well as for the network node with average characterization within cluster 1. On the other hand, we represent in dashed lines the values of $(\lambda/\mu)/N \in \{0, 0833, 1.667, 16.667\}$ used for Figure 10 in Chapter 5 [**J2**], which define the higher limits for $(\lambda/\mu)/N$ when scaling factors $\alpha \in \{0.5, 10, 100\}$ are set, respectively. Therefore, the congestion levels considered in this chapter range between low (i.e., conventional overscaled networks) and medium values.

For the discretization of the search space for optimal InP's admission strategies ξ_{ν}^{opt}, we study and compare the performance offered by a fine, intermediate and coarse slicing timescale. More in detail, we assume $l = 3$ possible values for the slice intervals $T^{slicing} \in \{0.1/\mu, 1/\mu, 3/\mu\}$, where the extreme values represent, respectively, $T_{min}^{slicing}$ and τ. Besides, for AT strategies, we consider $h = 4$ possible admission thresholds because, according to results in Chapter 5 [**J2**], it is sufficient for enabling the full potential in terms of revenue maximization for any network condition. For the selection of the state conditions to be used for the offline solution of the problem in (6.3), we first perform a coarse selection of 100 samples chosen homogeneously over $\mathcal{V}'$, that is, the union of the state conditions according to network traces of different nodes. Afterwards, we run a fine scale sampling over the most occurring state conditions and achieve a sample set with cardinality $|\mathcal{V}|= 268$.

6.5.1.4 Offline pre-computation of optimal admission strategies

For the offline pre-computation of the optimal admission strategies by means of exhaustive search, we develop in Matlab a simulator that generates instances of request arrivals, tenants' departure and bidding processes, on which it enforces FCFS, BB and AT admission strategies accordingly, making sure that at least 500 thousand arrivals are detected for each of the tested network conditions. To this aim, we employ an Intel(R) Core(TM) i9-7900X CPU @3.30GHz with 64GB of RAM. On the other hand, when a NN-based solution of the problem in (6.3) is performed, we reserve 20% of pre-computed strategies for final test while, at each fold of the K-fold cross-validation process, we use the remaining 80% of pre-computed strategies for the optimization of the NN training over the following hyperparameters:

i) number of hidden layers $n_{HL} \in \{1,2\}$, and, ii) number of neurons per layer $s_{HL} \in \{5,10\}$, and when the following training functions are tested: *Levenberg-Marquardt backpropagation*, *Bayesian Regularization*, and *Bayesian Regularization*. In particular, K-fold cross-validation with $K = 8$ is used, therefore, as explained introduced in Section 6.4.2.2, the 70% and 10% of the sample state conditions $\mathcal{V}$ is used, at each fold, for training and validation, respectively.

In case of AT admission strategies, the outcome of cross-validation highlights that the best performance in terms of convergence time and output to target error minimization is obtained when two different NNs are used in parallel for computing independently ξ_{ν}^{opt} components (i.e., $\dot{\beta}$ and $T^{slicing}$), both with $n_{HL} = 2$ hidden layers and, respectively, with $s_{HL} = 10$ and 5 neurons per hidden layer. Finally, *Levenberg-Marquardt backpropagation* training function is the one that provides the best performance in terms of convergence to error ratio. Optimal $T^{slicing}$ provided by the NN for AT are also applied for the cases of FCFS and BB.

6.5.1.5 Runtime enforcement

For the performance assessment, we use a simulator similar to the one described above, with the main difference that optimal strategies are enforced this time over dynamic network conditions obtained from real traces. Given that those traces have a periodicity of $T_{trace} = 900$ seconds, we assume that network conditions remain constant within each trace interval. We remark that, due to time discretization, an intrinsic delay is introduced when enforcing periodic admission strategies with respect to on-demand ones. Indeed, optimal admission strategies ξ_i^{opt} are enforced at allocation interval i, over the vector $\{\beta_{s^c_{i-1,q}}\}$ of bids received during the previous allocation interval. Finally, because we implement slice allocation at fine timescale, we assume that network conditions remain approximately constant within a given slice interval, therefore, no traffic forecasting mechanisms are needed at instant t_i^0 (i.e., the beginning of i-th allocation interval) for guaranteeing the optimality of admission strategies within the slice interval.

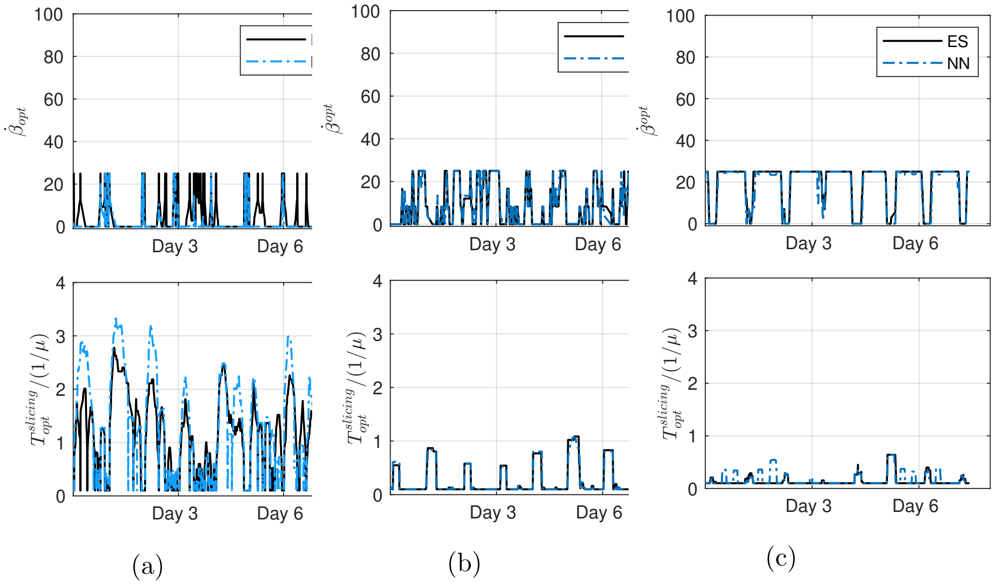


Figure 6.4: Optimal strategies ξ_i^{opt} (i.e., the timescale $T_i^{slicing}$ for slice allocation normalized to the average service time $1/\mu$ and, for AT strategies, admission threshold $\dot{\beta}_i$) computed over different network nodes' traces: a) *avNode1*, b) *Centr1*, and, c) *Centr2*. Results are compared for ES and NN-based approaches in solid black line and discontinuous blue line, respectively. The moving average over one hour is used for a clearer representation.

6.5.2 Performance evaluation

Below, we first present the admission strategies pre-computed by means of exhaustive search and NN-based search, as well as corresponding performance with respect to different admission strategies. Afterwards, we compare results with the case of admission strategies optimized on a per-node and a per-cluster basis.

6.5.2.1 Optimal strategies

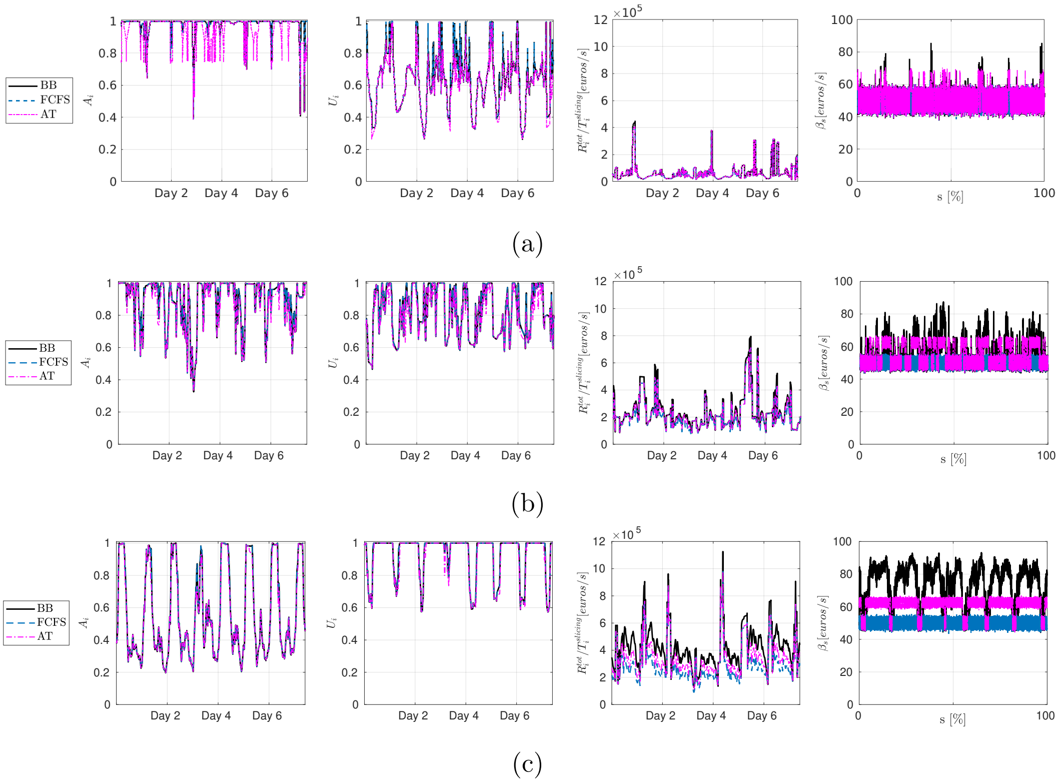
In Fig. 6.4, we can observe the fluctuation of the optimal strategies ξ_i^{opt} in time, expressed as: i) the timescale for slice allocation normalized to the average service time (i.e., $T_{opt}^{slicing}/(1/\mu)$), and, ii) the admission threshold for incoming bids when AT strategies are studied (i.e., $\dot{\beta}_{opt}$). Optimal strategies are provided over different network nodes' traces: a) *avNode1*, b) *Centr1*, and, c) *Centr2*. The strategies computed by means of exhaustive search and NN-based approach are provided in solid black line and discontinuous blue line, respectively.

We can observe in the figure how the chosen admission strategies change in presence of different average levels of congestion, increasing from Fig. 6.4a to Fig. 6.4c. More precisely, according to Fig. 6.4c, for high levels of congestion the recommendation to InPs is to adopt very fine timescales (i.e., small values for $T_i^{slicing}$) in such a way that more slice requests can be served in time. Furthermore, in the case of AT strategy, admission thresholds are set to the 25% of the bidding range for most of the time, while it mimics the FCFS scheme (i.e., the minimum admission threshold is adopted; $\dot{\beta}^{opt} = \beta_m$) only when very low congestion levels are perceived; note that the lowest threshold values $\dot{\beta}^{opt}$ observed in Fig. 6.4c correspond to nightly hours, during which the utilization of the network is low. On the other hand, according to Fig. 6.4a and 6.4b, more relaxed strategies are preferred when congestion levels get lower. Indeed, if arrivals are less frequent, less resolution is needed in time for serving all incoming slice requests, consequently coarser timescales can be adopted. Besides, in those cases, lower admission thresholds are preferred on average by the AT strategy.

Finally, we can observe that more flexible admission strategies are adopted by the NN-based approach, indeed, intermediate admission strategies are provided in the continuous domain for state conditions that were not explored for optimal strategy pre-computation. This phenomenon is particularly visible in the case of low congestion levels (see Fig. 6.4a), where coarser timescales are provided and combined, in the case of AT strategy, with lower values of the admission thresholds. However, we can see that NN's recommended strategies follow quite well those found by the exhaustive search approach. This means that the NN model is well adjusted to the training data provided by the ES study (cf. Section 6.4.2)

6.5.2.2 Performance evaluation with exhaustive search

As we introduced in Section 6.3, a measure of the timeliness of a slice admission control method is provided by the average waiting time $\bar{\tau}_i$, equal to half of the admission timescale $T_i^{slicing}$. Besides, we remind that the minimum timescale $T_{min}^{slicing}$ allowed by InP is defined by technological and complexity factors, while its maximum value τ depends on SPs' latency constraints. In Fig. 6.4, we observed how the optimal admission strategies tend to provide relaxed timeliness for decreasing values of congestion level. Therefore, we remark the importance of defining in the SLA



(a)

(b)

(c)

Figure 6.5: Performance assessed by adopting strategies ξ_i^{opt} computed by means of exhaustive search over different network nodes' traces: a) *avNode1*, b) *Centr1*, and, c) *Centr2*. A comparison is provided between different admission strategies (i.e., BB, FCFS and AT) in terms of admission ratio A_i, percentage of resource utilization U_i, revenue rate $R_i^{tot}/T_i^{slicing}$ and accepted bids β_s. The moving average over one hour is used for a clearer representation.

both τ and the penalty to the InP when this condition is not met, especially in the case of SPs with very strict requirements in terms of $\bar{\tau}_i$ and in presence of very low congestion levels (e.g., see Fig. 6.4a).

In Fig. 6.5, performance is assessed for different admission schemes (i.e., BB, FCFS, and AT) when strategies are computed by means of exhaustive search. In particular, different nodes' traces are considered: a) *avNode1*, b) *Centr1*, and, c) *Centr2*, and, from left to right, we represent the results for the rest of the performance metrics introduced in Section 6.3: i) the admission ratio A_i, ii) the percentage of resource utilization U_i, iii) the revenue rate $R_i^{tot}/T_i^{slicing}$, and, iv) the accepted bids β_s. Finally, in order to have a quantitative measure of performance over the week, we provide in

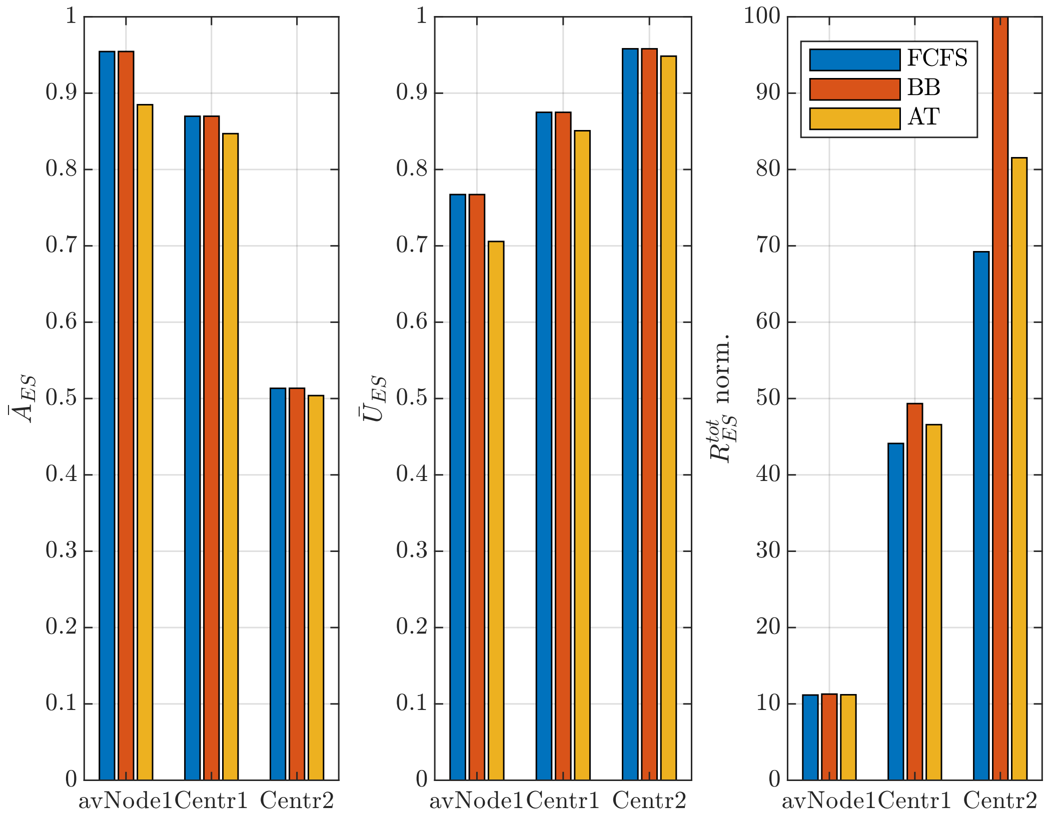


Figure 6.6: Performance over the week when adopting different admission schemes, with strategies computed for different network nodes by means of ES: i) average admission rate $\bar{A}$, average percentage of resource utilization $\bar{U}$, and total aggregate revenue R^{tot}. For R^{tot}, values are normalized to the maximum over the three strategies and network nodes.

Fig. 6.6 the average admission rate $\bar{A}$, the average percentage of resource utilization $\bar{U}$, and the total aggregate revenue R^{tot}.

It can be observed in both figures that BB and FCFS always provide the same values for the admission rate, as they both allow slice allocation up to resource-exhaustion. On the other hand, the AT strategy reduces utilization by rejecting bids below a given threshold, which also corresponds to a lower admission ratio. This is particularly evident in the case of low congestion levels (e.g., *avNode1* according to Fig. 6.3), as the relative ratio of rejections increases with respect to the number of arrivals. Similar considerations hold for the percentage of resource utilization because, thanks to the lower number of admissions, less resources are used on average by the AT strategy when compared with FCFS and BB schemes.

In terms of revenue to the InP, any strategy provides similar revenues in case of low levels of congestion (e.g., *avNode1*). Indeed, because resources are overdimensioned with respect to the economic opportunities, each of the considered schemes tries to admit every incoming request. When congestion increases, the choice of the bids to admit becomes crucial for the revenue maximization, however, only AT and BB strategy can exploit the potential offered by the bigger number of incoming slice requests for achieving higher revenues. Comparing into more detail the revenue offered by different admission schemes, FCFS strategy provides the minimum revenue at zero complexity for its enforcement at runtime (i.e., it admits every new slice request up to resource saturation). On the other hand, BB approaches allow InPs to always select the highest bids at the cost of higher complexity in the long term, as explained in Section 6.4.1. Finally, AT approaches represent a tradeoff between FCFS and BB schemes in terms of revenue and complexity. Indeed, they always offer intermediate revenues between FCFS and BB schemes. Besides, strategies can be computed offline only once and enforced at runtime by comparing incoming bids with a threshold.

Together with complexity, admission rate, resource utilization and revenue, another term of comparison for the admission strategies is represented by the admitted bids β_s, which are shown in Fig. 6.5 with respect to the order of arrival s normalized to the total number of arrivals over the week. As explained for revenue, in case of low congestion levels, all admission schemes admit slice requests independently of the associated bids due to the scarcity of incoming revenue opportunities with respect to resources available. Consequently, according to the figure, the average admitted bid equals the mean value of β_s (i.e., $\bar{\beta}_s = (\beta_M - \beta_m)/2$ for a uniform bid distribution). We remark that the moving average over one hour is used for a clearer representation in Fig. 6.5. On the other hand, when the congestion level increases, FCFS does not change its admission strategy, while both AT and BB schemes become more selective and admit slice requests with higher associated bids.

The admission rate together with the average value for admitted bids can be interpreted as a measure of the fairness of InPs towards SPs accounting for: i) InP's greediness in resource usage for revenue maximization, and, ii) fair treatment of SPs' spending power, respectively. In conclusion, FCFS is the admission strategy

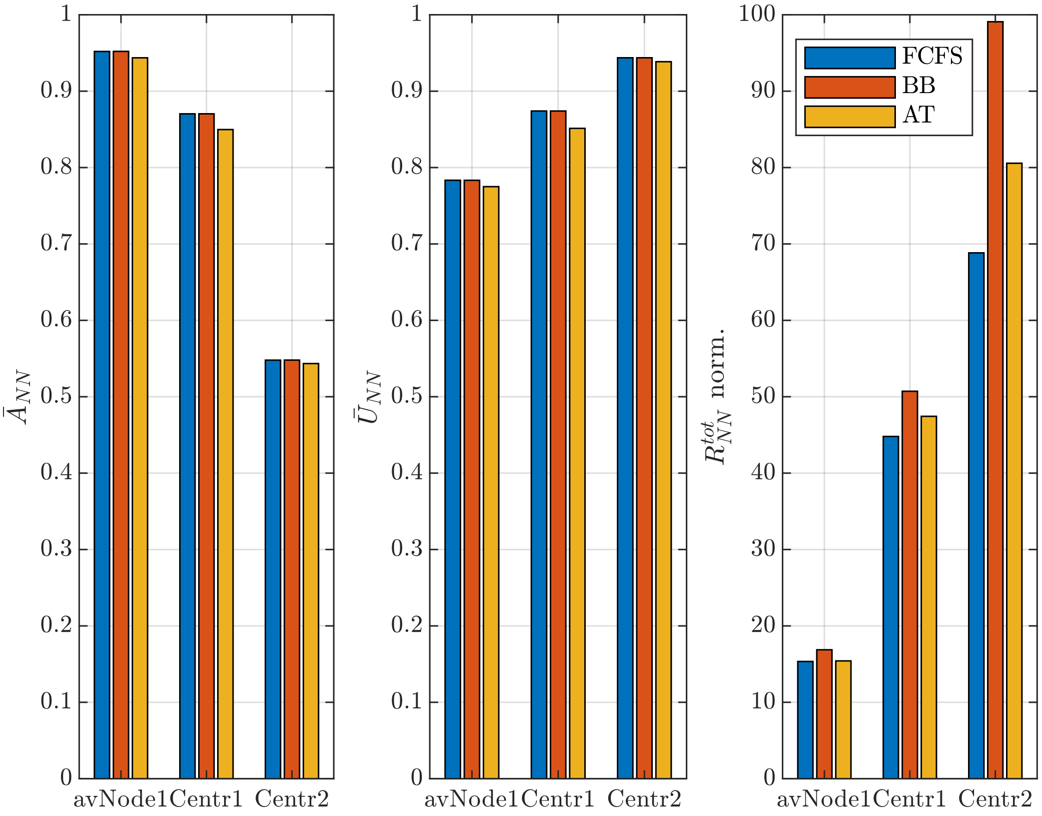


Figure 6.7: Performance over the week for different network nodes and admission schemes, with NN-based strategies. For R^{tot}, values are normalized to the maximum over the three schemes and network nodes when optimal strategies are adopted.

with lower complexity and highest level of fairness, as it provides highest admission rates and lowest average values for the admitted bids. On the other hand, BB scheme maximizes revenues at the cost of increased complexity and lowest fairness towards SPs' spending power, as it sets the highest average value for the admitted bids. Finally, AT approach represents a tradeoff between the other considered schemes, as it provides slightly lower admission rates while requiring less resources. Besides, it is capable of providing higher revenues than FCFS strategy and, when compared with BB approach, it limits complexity and provides a more fair solution in terms of SPs' spending power, by setting lower average value for the admitted bids.

6.5.2.3 Performance evaluation with ML-based strategies

In Sections 6.2 and 6.1, we introduced the possibility of adopting ML-based solutions for providing near-optimal admission strategies for network conditions that have not been directly explored by the InP during the pre-computation phase. More in detail, in Section 6.4.2, the advantages in terms of computational efficiency have been detailed for the case of a NN trained on the exhaustive search's output, thus providing custom admission strategies for different network nodes and congestion levels. In Fig. 6.7, we provide the performance study when strategies are chosen by means of a NN-based approach, which can be compared to that in Fig. 6.6 for optimal strategies.

In the case of network nodes with low congestion levels (e.g., *avNode1*), it can be observed that FCFS and BB strategies do not have much margin for improving the admission rate due to the very low number of slice requests arriving. This does not hold for the AT scheme as an increase in admission rate can still be achieved by adopting lower admission bids (see Fig. 6.4a). On the other hand, a great benefit in terms of revenue is offered by the adoption of a NN-based approach independently of the admission strategy. This can be explained by the better customization achieved in terms of admission timescales with respect to the input network conditions. The revenue increase comes at the cost of an increase in resource utilization for all admission schemes, which is more evident in the case of AT strategy because of the adoption of lower admission thresholds. More precisely, BB is the strategy experiencing the higher gain thanks to the increase in the average timescale adopted (see Fig. 6.4a), because a better opportunity is provided for selecting the highest bids among the arrivals. However, we remind that a more unfair behavior is experienced by SPs with respect to their spending power (see Fig. 6.5a).

When considering network nodes with medium congestion level (e.g., *Centr1*), the NN-based approach provides only a slight performance improvement with respect to the exhaustive search approach, which is confirmed by the fact that very similar strategies are adopted by the two approaches (see Fig. 6.4b). Finally, in the case of very high congestion levels (e.g., *Centr2*), admission rate can be improved by the better customization of admission timescales in time, although, with a slight increase in the average timescale used (see Fig. 6.4c). Consequently, a worse timeliness is achieved when serving incoming requests, which corresponds to a slightly lower resource utilization and revenues over the week.

In conclusion, the adoption of a NN-based solution for the computation of optimal admission strategies is recommended for network nodes with low or medium congestion levels. In particular, it can provide great gains in terms of revenue, mostly if a BB strategy is adopted and some flexibility exists in terms of fairness towards SPs. Besides, NN-based approaches are suitable for improving AT scheme's admission rate when fairness is preferred over revenue maximization. On the other hand, in case of high congestion levels, there is no incentive in adopting NN-based strategies due to the suboptimal nature of their solutions.

From a different perspective, the limited drop in performance when compared to optimal strategies motivates the adoption of a NN-based approach in case of lack of information on the precise statistics on the network conditions. Indeed, because the NN has been trained on a collection of state conditions from different network nodes with different congestion levels, it represents a suboptimal but more general solution for any network node under any circumstance. Consequently, the trained NN itself could be used by InPs as a computationally-efficient way to provide admission strategies for newly deployed nodes, or for adapting to changes in the congestion levels of already deployed nodes. On the other hand, it could also be used as a tradable asset leased among InPs, or, as a possible object of standardization for guaranteeing comparable performance across different InPs' networks.

6.5.2.4 Performance evaluation with clustering

In Section 6.2, we discussed the possible reduction in complexity offered by clustering solutions when performing the computation of admission strategies at a network level. More precisely, in Section 6.5.1.2, we described the methodology for clustering network nodes according to traces, allowing the computation of the admission strategies only for one candidate within each cluster (i.e., the centroid of the cluster). Below, we assess the difference in performance obtained when the optimal strategies of one cluster's centroid are used both for a different node within the cluster and for a node belonging to another cluster. In Fig. 6.8, we show performance when *Centr1* strategies are enforced at different network locations (i.e., *avNode1* and *Centr2*), which can be compared to that in Fig. 6.6 for optimal strategies.

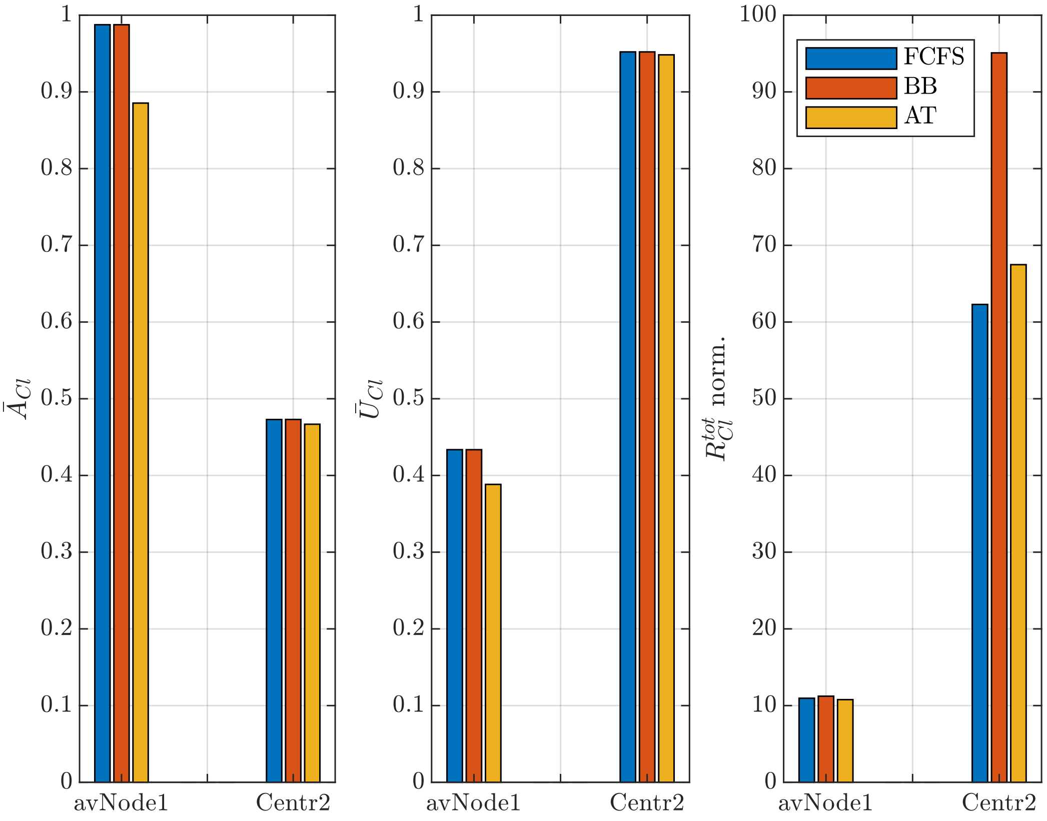


Figure 6.8: Performance over the week for different admission strategies, when *Centr1*'s optimal strategies are applied to *avNode1* and *Centr2*. For R^{tot}, values are normalized to the maximum over the three schemes and network nodes when optimal strategies are adopted.

When applying *Centr1*'s strategies to the network node with average characterization within cluster 1 (i.e., *avNode1*), we can observe that the admission rate slightly increases for FCFS and BB strategy thanks to the average decrease in the admission timescales adopted (see Fig. 6.4a and 6.4b). This does not hold for AT approach as the improved timeliness is counterbalanced by the choice of higher average admission thresholds. On the other hand, resource utilization considerably decreases for all strategies, because of the choice of strategies that are not optimal for the low congestion levels typical of *avNode1*. For the same reason, a negligible reduction in revenue is also registered. Finally, the enforcement of *Centr1*'s strategies in presence of other clusters' conditions (i.e., *Centr2*) provides lower admission rates, resource utilization and revenues, as expected by observing the difference in the admission strategies represented in Fig. 6.4b and 6.4c.

In conclusion, adopting clustering strategies represents a valid option for reducing the complexity associated with the enforcement at runtime of optimal strategies over InPs' networks with centralized architectures. Moreover, it could be used in the case of network nodes with well-known statistics on congestion levels and uncertain information about current states. Indeed, in both cases, instead of monitoring and adapting optimal strategies independently for each network node, the InP can alternatively divide network nodes into clusters and apply the strategies that are optimal for a candidate node (e.g., *Centr1*) to the rest of the nodes within the cluster (e.g., *avNode1*), with a negligible difference in terms of performance. Besides, if InP's priority is placed on the maximum reduction in complexity, the same strategies could be also adopted for nodes belonging to other clusters (e.g., *Centr2*) with limited decrease in performance.

6.6 Summary

In this chapter, we target the potential offered by 5G's marketplace both to network owners and SPs, in terms of revenue and QoS guarantees for services with strict latency constraints (e.g., uRLLC services). In particular, an intra-service reservation-based slicing mechanism has been defined for fine and adaptable timescales, with optimal strategies pre-computed offline for state conditions that are representative of both SPs' behavior, and resource availability in the network. A PoC on real network traces is implemented for studying and comparing complexity and performance of three reference admission strategies (i.e., FCFS, AT, and BB), the latter expressed in terms of efficiency in resource utilization, fairness to the SPs and InP's revenue. Finally, results obtained for optimal admission strategies are compared with those of more computationally efficient solutions.

In this context, this chapter proves that FCFS and BB strategies provide the minimum and maximum revenue to the InP, respectively, while the opposite holds true in terms of fairness towards SPs and complexity required for enforcement. On the other hand, the AT scheme provides a tradeoff in terms of complexity and performance, while reducing the average resource utilization when variable timescales are used. Furthermore, in case of low congestion levels, the improvement in terms of admission rate and revenue has been demonstrated when using ML-based solutions, at the cost of slightly higher resource utilization and lower fairness with respect to SPs' spending power.

Results show that, if InP's objective is a reduction in complexity, or, the computation of near-optimal strategies in absence of full information about network conditions, approaches based on ML and clustering are good solutions that come at the cost of a negligible or limited decrease in performance. Finally, we remind that only approaches based on periodic admission control are considered in this chapter. Therefore, as discussed in previous chapter, better performance could possibly be achieved by adopting on-demand approches, at the cost of higher complexity.

Chapter 7

Conclusions and Future Works

The heterogeneity of services foreseen for 5G use cases, together with the high number of devices that will populate next-generation networks are expected to determine a traffic explosion that will set huge challenges for network owners. In particular, besides a dense network deployment for an efficient utilization of the spectrum resource, the adoption of architectural and technological solutions capable of enabling programmability and efficient fine-scale sharing is fundamental. E2E network slicing has the capability to make next-generation networks attractive for SPs and profitable for the MNOs and, in this thesis, we highlighted possible solutions for achieving the full potential of the 5G market place, both from an architectural and methodological perspective.

In Chapter 2, we presented the SoA architecture and technologies proposed so far for network programmability and scalability at different segments of the network (i.e., access, core, and transport network). In Chapter 3, we introduced the roadmap for the deployment of 5G networks, by describing the main challenges and solutions related to the enabling of multi-tenancy and fine-scale network slicing. More in detail, after reviewing the main reasons for the incompatibility of conventional networks with the network programmabilities paradigms, we introduced the concept of network slicing for QoS customization and fine-scale sharing support. An an enhanced architecture has been defined and compared with the conventional one and with existing proposals by standardization bodies, and ongoing industrial/academical projects. We highlighted the importance of adopting a centralized architecture together with network-wide virtualization for E2E QoS programmability, efficient network sharing, and flexible negotiation of SLAs exposed to third-party SPs.

In Chapter 4, we proposed a novel scheme based on game theory for the assessment of cooperation incentives for coexisting MNOs. The performance and financial gains offered by fine-scale network sharing are studied by means of coalitional game theory, and different scenarios have been considered in terms of number of operators, market share, and spectrum license. The performance of a given cooperative approach is defined in terms of the data rate enhacements for the end users, and the profit gain to the MNOs. Besides, the tariffs charged to the end users are used as a decision paramenter for determing the convenience of specific coalitions. Results show the benefits achievable through collaboration, which can be considered as the financial incentives for MNOs to upgrade their networks according to the 5G's roadmap. Besides, it is highlighted that unbalanced cost redistribution can be also accepted within coalitions when additional costs are counterbalanced by higher revenue rates with respect to the stand-alone case. However, when cooperating with equal-sized operators, higher profit opportunities can be typically achieved by MNOs while charging much lower tariffs to the end users. Finally, MNOs are provided with guidelines for choosing the tariff to adopt and the coalition to form taking into account their business models.

In Chapter 5, we defined a policy-based slice provision mechanism that prioritizes SPs' bids above a certain threshold and optimizes, at a reduced complexity, the admission policy with respect to different sizes of the resource pool, traffic loads and SPs' behavior. Results demonstrated the enhancements provided by such approach in terms of slice provision promptness, revenue maximization and expenditure reduction to the InPs, while guaranteeing QoS and fairness towards SPs. The performance has been compared for two approaches that serve slice request on-demand or at regular time-intervals. Besides, the performance of the proposed policy is compared with that of two reference policies, the first one admits SPs' requests in order of arrival, independently of the associated bids, while the second admits requests according to a decreasing listing of the associated bids (i.e., from the highest to the lowest). Results proved that the proposed policy offers real-time slicing with the highest revenues and lowest expenditures to the InPs, in exchange for a negligible loss in terms of fairness towards SPs. Finally, the proposed policy allows the achievement of high revenues in case of regimes with scarce resource availability and high-loads, while keeping complexity limited when compared with the reference policies.

In Chapter 6, we demonstrated the potential of 5G's marketplace in terms of revenues to the network owners and QoS guarantees to the SPs, especially for services with strict latency constraints. To this aim, we defined an exclusive allocation mechanism for slice requests of the same kind, which is performed at fine and adaptable timescales. Sample congestion levels are used for the offline pre-computation of optimal admission strategies in reference state conditions, while more computationally efficient solutions are used for the extension of the admission strategies to unexplored conditions. Performance associated to different admission strategies (i.e., above threshold, in order of arrival, and following a decreasing bid listing) is compared in a proof of concept with real netowork traces, assessing the efficiency in resource utilization, the fairness to the SPs, the InPs' revenue and the complexity for the computation of the admission strategies. In particular, it is proved that the admission strategies that follow the order of arrival and the decreasing bid listing provide, respectively, the minimum and maximum revenue to the InP, while the opposite holds true in terms of fairness towards SPs and complexity required for the strategy enforcement. In comparison, the proposed policy, which admits bids above a specific threshold adapted in time with respect to the state conditions, provides a tradeoff in complexity and performance, and reduces the average resource utilization. Finally, we demonstrated the possible improvement achievable by means of machine learning-based solutions in terms of admission rate and revenues. Machine learing-based solutions proved to be particularly useful in case of low congestion levels, at the cost of slightly higher resource utilization and lower fairness towards SPs'. On the other hand, when the objective is the limitation of complexity, or, if there is a lack of information about the network conditions, machine learning-based solutions and clustering methodologies turned out to be good approaches in exchange for a negligible or limited decrease in performance.

7.1 Future Research Directions

In this thesis, we studied possible architectural and methodological solutions for fully exploiting the potential offered by the 5G market place, in terms of revenues provided to the InPs for building next-generation networks, cost reduction for making future networks sustainable, and enhanced service provision for enabling 5G services. In this section, we discuss some of the open issues that, in our opinion, are worth to be validated in the future in order to continue the research presented in this thesis. Besides, we provide a discussion on the alternative research directions explored in the literature for E2E and elastic network sharing.

With respect to the sharing mechanisms among MNOs for QoS improvement and profit increase to the MNOs, it would be interesting to extend the proposed coalitional game presented in Chapter 4 to different scenarios, with different number of operators, and combinations of market shares and spectrum licenses. Besides, a more sophisticated model could be introduced for traffic, with different classes of services, and for highlighting the statistical multiplexing gain achievable in a network wide perspective thanks to resource pooling. Finally, heterogeneous resources could be considered for end-to-end service provision, and alternative models could be considered for MNOs revenues, taking into account the possibility of capturing other MNOs' subscribers.

In order to improve the utilization of network resources at a fine scale while guaranteeing custom QoS to end users with fast innovation cycles accessible to any SP, we introduced in Chapters 5 and 6 a slice provision mechanism. The proposed mechanism enables flexible sharing among InPs, and allows multiple tenants to provide services over the same network without the need of owning neither spectrum nor infrastructure. Considering the financial incentives deriving from the dynamic lease of network slices to SPs, the defined slice provision mechanism paves the way for future applications in multiple scenarios. However, many open directions remain to be explored for the work presented in this thesis.

According to the main research trends in slice admission control mechanisms for 5G [193], both exclusive and shared slice allocation could be modeled for competing SPs, while multi-queuing models could be adopted for studying the coexistence of diverse service classes, each with different resource requirements. It would also be worth extending the queuing model considered in this work for handling rejected slice requests.

An extension of the proposed slice provision approach could be studied for the case with multiple InPs composing E2E slices out of heterogeneous resources. In particular, the distributed blockchain technology could be employed for extending the slice bidding mechanisms described in Chapters 5 and 6 to the E2E [194]. Indeed, the E2E network slice auctioneer introduced in Chapter 2, equipped with contextaware network management functionalities, could perform coordinated policy-based admission strategies over different network segments.

From the SPs modeling perspective, the bidding model could be extended by adding the adaptability to the market perception, thus allowing a study on the effects of rational bidding strategies on the proposed mechanism. Indeed, SPs could react to the fluctuations in price and admission rate by adapting their bidding strategy in time. Finally, we foresee the implementation of the proposed methodology on real testbeds for proving the feasibility of adopting adaptive timescales in available technology.

Considering different approaches emerged in the literature for slice admission control, many foresee the solution of optimization problems with different objective functions (e.g., high InP's revenues, low rejection rate, efficient resource utilization, low congestion, or any combination of the above), however, the associated complexity could prevent an efficient and real-time implementation for E2E slicing. Among the alternative strategies surveyed in [193], many converge in the adoption of ML approaches for keeping complexity limited, while achieving suboptimal admission decision.

In order to optimize the target performance metrics while guaranteeing tenants' isolation, low complexity and flexible resource utilization, our approach employs the adaptation of the admission strategies and timescales at runtime, which are pre-computed and enforced for exclusive slice allocation to different SPs. Therefore, this mechanism provides a single-step framework for executing joint admission control and resource allocation, guaranteeing isolation by performing exclusive resource allocation (i.e., per-tenant slicing), enabling flexibility by adapting the slicing timescales, controlling congestion by adapting the admission thresholds, and limiting complexity by adopting ML-based solutions for the offline pre-computation of the admission strategies. In this work, we proved the advantages in the adoption of this approach in case of simple infrastructures managed by a unique orchestrator, and in case of single service types, mostly those with strict time constraints.

Many of the alternative solutions existing in the literature, on the other hand, perform slice admission control at constant and coarse timescales (i.e., in the order of hours) [193,195], and achieve elasticity in resource allocation by adapting the resource associated to each slice at finer timescales (i.e., in the order of minutes) [184,195].

This two-step slice admission and resource (re)allocation approach seems to be a good solution for supporting multiple service types with heterogeneous requirements and in case of complex infrastructures managed by uncoordinated orchestrators. Indeed, a single-step slice admission control as the one adopted in our solution would require context-awareness over the whole network and, most probably, high control overheads as well as high complexity for the offline pre-computation of network-wide admission strategies. Contrarily, a two-step approach reduces complexity and the need of network-wide context-awareness by first performing the admission control step at different network segments (i.e., RAN, transport network, cloud infrastructure), while the dynamic resource (re)allocation is employed for elasticity support and efficient resource usage.

Although the application of our approach to E2E slicing with heterogeneous resources and service types still has to be demonstrated, in this work, we proved its advantages for intra-service slice allocation, that is, a resource allocation phase that could be executed for a slice with multiple tenants with same service requirements. In this context, the proposed approach could be integrated with two-step slice admission strategies, by being jointly executed with scheduling algorithms and, thus, enabling per-tenant slice allocation.

Bibliography

[1] CISCO White Paper, *Cisco Annual Internet Report (2018-2023)*, March 2020.

[2] CISCO White Paper, *Cisco visual networking index: Global mobile data traffic forecast update, 2016-2021*, February 2017.

[3] TS 28.530 3GPP, *Aspects; Management and orchestration; Concepts, use cases and requirements*, Rel. 15, December 2018.

[4] TR 28.801 3GPP, *Telecommunication management; Study on management and orchestration of network slicing for next generation network*, Rel. 15, January 2018.

[5] TR 22.891 3GPP, *Feasibility Study on New Services and Markets Technology Enablers; Stage 1*, Rel. 14, December 2018.

[6] 3GPP TS 22.261, *Service requirements for the 5G system; Stage 1*, Rel. 18, January 2021.

[7] Ekram Hossain, Long Bao Le, and Dusit Niyato, *Overview of Multi-Tier Cellular Wireless Networks*, John Wiley & Sons, Ltd, 2013.

[8] Erik Dahlman, Stefan Parkvall, and Johan Skold, "Chapter 3 - LTE Radio Access: An Overview," in *4G LTE-Advanced Pro and The Road to 5G (Third Edition)*, pp. 29–53. Academic Press, third edition edition, 2016.

[9] TR 38.815 3GPP, *Technical Specification Group Radio Access Network; New frequency range for NR (24.25-29.5 GHz)*, Rel. 15, September 2021.

[10] ITU-T Y.3011, *Framework of Network Virtualization for Future Networks*, Next Generation Networks-Future Networks, January 2012.

[11] Tech. Rep. FCC ET Docket no. 03-222, *Notice of proposed rule making and order*, Federal Communications Commission, December 2003.

[12] TR 36.889 3GPP, *Feasibility Study on Licensed-Assisted Access to Unlicensed Spectrum*, Rel. 13, June 2015.

[13] TS 36.360 3GPP, *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); LTE-WLAN Aggregation Adaptation Protocol (LWAAP) specification*, Rel. 15, July 2018.

[14] Nikolaos Nomikos, Prodromos Makris, Dimitrios N. Skoutas, Demosthenes Vouyioukas, and Charalabos Skianis, "Enabling wireless prosuming in 5G networks," in *2014 International Conference on Telecommunications and Multimedia (TEMU)*, 2014, pp. 190–195.

[15] Marja Matinmikko, Miia Mustonen, Dennis Roberson, Jarkko Paavola, Marko Hoyhtya, Seppo Yrjola, and Juha Roning, "Overview and comparison of recent spectrum sharing approaches in regulation and research: From opportunistic unlicensed access towards licensed shared access," in *2014 IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN)*, 2014, pp. 92–102.

[16] China Mobile, *C-RAN - Road towards Green Radio Access Network*, C-RAN Int'l. Wksp., Beijing, April 2010.

[17] Alcatel-Lucent Technology White Paper, *Network Sharing in LTE, Opportunity & Solutions*, 2010.

[18] GSMA, *Mobile Infrastructure Sharing*, September 2012.

[19] TR 22.951 3GPP, *Technical Specification Group Services and System Aspects; Service aspects and requirements for network sharing*, Rel. 16, July 2020.

[20] 3GPP TS 32.130, *Telecommunication management; Network sharing; Concepts and requirements*, Rel. 14, December 2016.

[21] Dario Bega, Marco Gramaglia, Albert Banchs, Vincenzo Sciancalepore, Konstantinos Samdanis, and Xavier Costa-Perez, "Optimising 5G infrastructure

markets: The business of network slicing," in *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, 2017, pp. 1–9.

[22] NGMN Alliance, *Description of network slicing concept*, Public Deliverable, January 2016.

[23] 5G-PPP projects, URL: https://5g-ppp.eu/ (visited on 10/02/2016).

[24] Ekram Hossain and Monowar Hasan, "5G cellular: key enabling technologies and research challenges," *IEEE Instrumentation Measurement Magazine*, vol. 18, no. 3, pp. 11–21, 2015.

[25] Nikos Bizanis and Fernando A. Kuipers, "SDN and Virtualization Solutions for the Internet of Things: A Survey," *IEEE Access*, vol. 4, pp. 5591–5606, 2016.

[26] Ying Zhang, *Network Function Virtualization: Concepts and Applicability in 5G Networks*, Hoboken, NJ, USA: John Wiley &amp; Sons, Inc., January 2018.

[27] Van-Giang Nguyen, Truong-Xuan Do, and YoungHan Kim, "SDN and Virtualization-Based LTE Mobile Network Architectures: A Comprehensive Survey," *Wireless Personal Communications*, vol. 86, pp. 1401–1438, 2016.

[28] Peter Rost, Ignacio Berberana, Andreas Maeder, Henning Paul, Vinay Suryaprakash, Matthew Valenti, Dirk Wubben, Armin Dekorsy, and Gerhard Fettweis, "Benefits and challenges of virtualization in 5G radio access networks," *IEEE Communications Magazine*, vol. 53, no. 12, pp. 75–82, 2015.

[29] Timothy Wood, K. K. Ramakrishnan, Jinho Hwang, Grace Liu, and Wei Zhang, "Toward a software-based network: integrating software defined networking and network function virtualization," *IEEE Network*, vol. 29, no. 3, pp. 36–41, 2015.

[30] Songlin Sun, Michel Kadoch, Liang Gong, and Bo Rong, "Integrating network function virtualization with SDR and SDN for 4G/5G networks," *IEEE Network*, vol. 29, no. 3, pp. 54–59, 2015.

[31] Bo Han, Vijay Gopalakrishnan, Lusheng Ji, and Seungjoon Lee, "Network function virtualization: Challenges and opportunities for innovations," *IEEE Communications Magazine*, vol. 53, no. 2, pp. 90–97, 2015.

[32] Yong Li and Min Chen, “Software-Defined Network Function Virtualization: A Survey,” *IEEE Access*, vol. 3, pp. 2542–2553, 2015.

[33] Jun Wu, Zhifeng Zhang, Yu Hong, and Yonggang Wen, “Cloud radio access network (C-RAN): a primer,” *IEEE Network*, vol. 29, no. 1, pp. 35–41, 2015.

[34] Rashid Mijumbi, Joan Serrat, Juan-Luis Gorricho, Niels Bouten, Filip De Turck, and Raouf Boutaba, “Network function virtualization: State-of-the-art and research challenges,” *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 236–262, 2016.

[35] Juliver Gil Herrera and Juan Felipe Botero, “Resource Allocation in NFV: A Comprehensive Survey,” *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 518–532, 2016.

[36] Matias Richart, Javier Baliosian, Joan Serrat, and Juan-Luis Gorricho, “Resource Slicing in Virtual Wireless Networks: A Survey,” *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 462–476, 2016.

[37] ETSI Industry Specification Group (ISG) NFV, *ETSI GS NFV 006 V2.1.1: Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Architectural Framework Specification*, January 2021.

[38] TS 28.527 3GPP, *Life Cycle Management (LCM) for mobile networks that include virtualized network functions; Stage 2*, Rel. 15, June 2018.

[39] Alcardo Alex Barakabitze, Arslan Ahmad, Rashid Mijumbi, and Andrew Hines, “5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges,” *Computer Networks*, vol. 167, pp. 106984, 2020.

[40] David Lake, Ning Wang, Rahim Tafazolli, and Louis Samuel, “Softwarization of 5g networks,Äìimplications to open platforms and standardizations,” *IEEE Access*, vol. 9, pp. 88902–88930, 2021.

[41] CISCO White Paper, *Cisco global cloud index: Forecast and methodology, 2016-2021*, 2018.

[42] N. M. Mosharaf Kabir Chowdhury and Raouf Boutaba, “Network virtualization: state of the art and research challenges,” *IEEE Communications Magazine*, vol. 47, no. 7, pp. 20–26, 2009.

[43] Zvika Bronstein and Eyal Shraga, “NFV virtualisation of the home environment,” in *2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*, 2014, pp. 899–904.

[44] Hassan Hawilo, Abdallah Shami, Maysam Mirahmadi, and Rasool Asal, “Nfv: state of the art, challenges, and implementation in next generation mobile networks (vEPC),” *IEEE Network*, vol. 28, no. 6, pp. 18–26, 2014.

[45] ETSI Industry Specification Group (ISG) NFV, *ETSI GR NFV 001 V1.3.1: Network Functions Virtualisation (NFV); Use Cases*, March 2021.

[46] Windhya Rankothge, Jiefei Ma, Franck Le, Alessandra Russo, and Jorge Lobo, “Towards making network function virtualization a cloud computing service,” in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2015, pp. 89–97.

[47] Diego Kreutz, Fernando M. V. Ramos, Paulo Esteves Verissimo, Christian Esteve Rothenberg, Siamak Azodolmolky, and Steve Uhlig, “Software-Defined Networking: A Comprehensive Survey,” *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015.

[48] Nick Feamster and Hari Balakrishnan, “Detecting BGP Configuration Faults with Static Analysis,” in *in Proc. Networked Systems Design and Implementation*, 2005, pp. 43–56.

[49] Barath Raghavan, Teemu Koponen, Ali Ghodsi, Martin Casado, Sylvia Ratnasamy, and Scott Shenker, “Software-defined internet architecture: decoupling architecture from infrastructure,” in *In Proceedings of the 11th ACM Workshop on Hot Topics in Networks*. 2012, pp. 43–48, ACM.

[50] Nick Feamster, Jennifer Rexford, and Ellen Zegura, “The road to sdn: An intellectual history of programmable networks,” vol. 44, no. 2, pp. 87–98, apr 2014.

[51] Theophilus Benson, Aditya Akella, and David Maltz, “Unraveling the Complexity of Network Management,” in *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation*, USA, 2009, NSDI’09, pp. 335–348, USENIX Association.

[52] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner, "Openflow: Enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, mar 2008.

[53] P. Newman, G. Minshall, and T.L. Lyon, "IP switching-ATM under IP," *IEEE/ACM Transactions on Networking*, vol. 6, no. 2, pp. 117–129, 1998.

[54] M. Paul *et al.*, *Applying SDN Architecture to 5G Slicing*, Open Network Fundation document, April 2016.

[55] Bruno Astuto A. Nunes, Marc Mendonca, Xuan-Nam Nguyen, Katia Obraczka, and Thierry Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," *IEEE Communications Surveys Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014.

[56] Chengchao Liang, F. Richard Yu, and Xi Zhang, "Information-centric network function virtualization over 5g mobile wireless networks," *IEEE Network*, vol. 29, no. 3, pp. 68–74, 2015.

[57] Hyojoon Kim and Nick Feamster, "Improving network management with software defined networking," *IEEE Communications Magazine*, vol. 51, no. 2, pp. 114–119, 2013.

[58] Teemu Koponen, Keith Amidon, Peter Balland, Martin Casado, Anupam Chanda, Bryan Fulton, Igor Ganichev, Jesse Gross, Natasha Gude, Paul Ingram, Ethan Jackson, Andrew Lambeth, Romain Lenglet, Shih-Hao Li, Amar Padmanabhan, Justin Pettit, Ben Pfaff, Rajiv Ramanathan, Scott Shenker, Alan Shieh, Jeremy Stribling, Pankaj Thakkar, Dan Wendlandt, Alexander Yip, and Ronghua Zhang, "Network Virtualization in Multi-Tenant Datacenters," in *Proceedings of the 11th USENIX Conference on Networked Systems Design and Implementation*, USA, 2014, NSDI'14, pp. 203–216, USENIX Association.

[59] Andrew D. Ferguson, Arjun Guha, Chen Liang, Rodrigo Fonseca, and Shriram Krishnamurthi, "Hierarchical policies for software defined networks," in *Proceedings of the First Workshop on Hot Topics in Software Defined Networks*, New York, NY, USA, 2012, HotSDN '12, pp. 37–42, Association for Computing Machinery.

[60] Christopher Monsanto, Joshua Reich, Nate Foster, Jennifer Rexford, and David Walker, "Composing software-defined networks," in *Proceedings of the 10th USENIX Conference on Networked Systems Design and Implementation*, USA, 2013, nsdi'13, pp. 1–14, USENIX Association.

[61] Teemu Koponen, Martin Casado, Natasha Gude, Jeremy Stribling, Leon Poutievski, Min Zhu, Rajiv Ramanathan, Yuichiro Iwata, Hiroaki Inoue, Takayuki Hama, and Scott Shenker, "Onix: A distributed control platform for large-scale production networks," in *Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation*, USA, 2010, OSDI'10, pp. 351–364, USENIX Association.

[62] Soheil Hassas Yeganeh, Amin Tootoonchian, and Yashar Ganjali, "On scalability of software-defined networking," *IEEE Communications Magazine*, vol. 51, no. 2, pp. 136–141, 2013.

[63] Sushant Jain, Alok Kumar, Subhasree Mandal, Joon Ong, Leon Poutievski, Arjun Singh, Subbaiah Venkata, Jim Wanderer, Junlan Zhou, Min Zhu, Jon Zolla, Urs Holzle, Stephen Stuart, and Amin Vahdat, "B4: Experience with a globally-deployed software defined wan," in *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, New York, NY, USA, 2013, SIGCOMM '13, pp. 3–14, Association for Computing Machinery.

[64] A Linux Foundation Collaborative Project OpenDaylight, https://www.opendaylight.org/ (visited on 28/12/2021).

[65] Federico M. Facca, Elio Salvadori, Holger Karl, Diego R. Lopez, Pedro Andres Aranda Gutierrez, Dejan Kostic, and Roberto Riggio, "Netide: First steps towards an integrated development environment for portable network apps," in *2013 Second European Workshop on Software Defined Networks*, 2013, pp. 105–110.

[66] Martin Casado, Nate Foster, and Arjun Guha, "Abstractions for Software-Defined Networks," *Commun. ACM*, vol. 57, no. 10, pp. 86–95, sep 2014.

[67] Joshua Reich, C. Monsanto, Nate Foster, Jennifer Rexford, and D. Walker, "Modular SDN programming with pyretic," *USENIX Login*, vol. 38, pp. 128–134, 01 2013.

[68] ETSI Industry Specification Group (ISG) NFV, *ETSI GS NFV-iFA 031 V4.2.1: Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Requirements and interfaces specification for management of NFV-MANO*, June 2021.

[69] ETSI Industry Specification Group (ISG) NFV, *ETSI GS NFV-IFA 027 V4.2.1: Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Performance Measurements Specification*, May 2021.

[70] ETSI Industry Specification Group (ISG) NFV, *ETSI GS NFV 005 V1.2.1: Network Functions Virtualisation (NFV); Proofs of Concept; Framework*, May 2021.

[71] Yasir Zaki, Liang Zhao, Carmelita Goerg, and Andreas Timm-Giel, "LTE mobile network virtualization," *Mobile Networks and Applications*, vol. 16, pp. 424–432, 2011.

[72] Stefan Schneider, Manuel Peuster, Kai Hannemann, Daniel Behnke, Marcel Muller, Patrick-Benjamin Bok, and Holger Karl, "Producing Cloud-Native: Smart Manufacturing Use Cases on Kubernetes," in *2019 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, 2019, pp. 1–2.

[73] A Linux Foundation Collaborative Project Open platform for NFV (OPNFV), https://www.opnfv.org/ (visited on 28/12/2021).

[74] ETSI Industry Specification Group (ISG) NFV, *ETSI GR NFV-MAN 001 V1.2.1: Network Functions Virtualisation (NFV); Management and Orchestration; Report on Management and Orchestration Framework*, December 2021.

[75] D.L. Tennenhouse, J.M. Smith, W.D. Sincoskie, D.J. Wetherall, and G.J. Minden, "A survey of active network research," *IEEE Communications Magazine*, vol. 35, no. 1, pp. 80–86, 1997.

[76] Aurel Lazar, Koon-seng Lim, and Franco Marconcini, "Realizing a foundation for programmability of atm networks with the binding architecture," *IEEE Journal on Selected Areas in Communication*, vol. 14, pp. 1214–1227, 1996.

[77] ONF Open Networking Foundation (ONF), https://opennetworking.org/ (visited on 28/12/2021).

[78] ETSI Industry Specification Group (ISG) NFV, *ETSI GS NFV-EVE 005 V1.1.1: Network Functions Virtualisation (NFV); Ecosystem; Report on SDN Usage in NFV Architectural Framework*, December 2015.

[79] Broadband Forum Cloud project, URL: https://www.broadband-forum.org/projects/cloud (visited on 28/12/2021).

[80] Hewlett Packard Enterprise (HPE) FLEXFABRIC 12900E series switches, URL: https://www.hpe.com/psnow/doc/PSN5443167USEN.pdf (visited on 28/12/2021).

[81] NEC QX series switches, URL: https://www.necam.com/sdn/ (visited on 28/12/2021).

[82] Open vSwitch, URL: https://www.openvswitch.org/ (visited on 28/12/2021).

[83] ONF Stratum, URL: https://opennetworking.org/stratum/ (visited on 28/12/2021).

[84] Pica8 PicOS software switches, URL: https://www.pica8.com/product/ (visited on 28/12/2021).

[85] Miguel Garcia, Alysson Bessani, Ilir Gashi, Nuno Neves, and Rafael Obelheiro, "Analysis of operating system diversity for intrusion tolerance," *Software: Practice and Experience*, vol. 44, no. 6, pp. 735–770, 2014.

[86] TS-016 ONF, *OpenFlow Management and Configuration Protocol*, OF-CONFIG 1.2, 2014.

[87] R. Enns, M. Bjorklund, J. Schoenwaelder, and A. Bierman, *Network configuration protocol (NETCONF)*, Internet Engineering Task Force, RFC 6241 (Proposed Standard), June 2011.

[88] B. Pfaff and B. Davie, *The Open vSwitch database management protocol*, Internet Engineering Task Force, RFC 7047 (Informational), December 2013.

[89] M. Smith *et al.*, *OpFlex control protocol*, Internet Engineering Task Force, Internet Draft, April 2016.

[90] H. Yin, H. Xie, T. Tsou, D. Lopez, P. Aranda, and R. Sidi, *SDNi: A message exchange protocol for software defined networks (SDNS) across multiple domains*, Internet Engineering Task Force, Internet Draft, June 2012.

[91] Minlan Yu, Andreas Wundsam, and Muruganantham Raju, "NOSIX: A Lightweight Portability Layer for the SDN OS," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 2, pp. 28–35, apr 2014.

[92] Andrew D. Ferguson, Arjun Guha, Chen Liang, Rodrigo Fonseca, and Shriram Krishnamurthi, "Participatory networking: An api for application control of sdns," *SIGCOMM Comput. Commun. Rev.*, aug 2013.

[93] ONF Open Network Operating System (ONOS), https://opennetworking.org/onos/ (visited on 28/12/2021).

[94] Frenetic, URL: http://frenetic-lang.org/ (visited on 28/12/2021).

[95] Andreas Voellmy, Hyojoon Kim, and Nick Feamster, "Procera: A language for high-level reactive network controls," in *HotSDN'12*, 2012.

[96] P4, URL: https://p4.org/ (visited on 28/12/2021).

[97] Ola Salman, Imad H. Elhajj, Ayman Kayssi, and Ali Chehab, "Sdn controllers: A comparative study," in *2016 18th Mediterranean Electrotechnical Conference (MELECON)*, 2016, pp. 1–6.

[98] Othmane Blial, Mouad Ben Mamoun, and Redouane Benaini, "An Overview on SDN Architectures with Multiple Controllers," *Journal of Computer Networks and Communications*, vol. 2016, 2016.

[99] NEC Network Operation Engine (NOE), URL: https://www.necam.com/SDN/Software/NOE/ (visited on 28/12/2021).

[100] Juniper Contrail Controller, URL: https://www.juniper.net/us/en/products/sdn-and-orchestration/contrail/contrail-networking.html (visited on 28/12/2021).

[101] VMware Software-Defined Wide Area Network (SD-WAN), https://www.vmware.com/content/dam/digitalmarketing/velocloud/en/documents/208805aq-so-vcloud-sd-wan-simplfd-uslet.pdf (visited on 28/12/2021).

[102] Mohammad Banikazemi, David Olshefski, Anees Shaikh, John Tracey, and Guohui Wang, "Meridian: an sdn platform for cloud network services," *IEEE Communications Magazine*, vol. 51, no. 2, pp. 120–127, 2013.

[103] FloodLight controller, URL: https://floodlight.atlassian.net/wiki/spaces/floodlightcontroller/overview (visited on 28/12/2021).

[104] ONF Open Network Operating System (ONOS) repository, https://github.com/opennetworkinglab/onos (visited on 28/12/2021).

[105] Yustus Eko Oktian, SangGon Lee, HoonJae Lee, and JunHuy Lam, "Distributed SDN controller system: A survey on design choice," *Computer Networks*, vol. 121, pp. 100–111, 2017.

[106] Chih-Lin I, Jinri Huang, Ran Duan, Chunfeng Cui, Jesse Jiang, and Lei Li, "Recent Progress on C-RAN Centralization and Cloudification," *IEEE Access*, vol. 2, pp. 1030–1039, 2014.

[107] Li Erran Li, Z. Morley Mao, and Jennifer Rexford, "Toward software-defined cellular networks," in *2012 European Workshop on Software Defined Networking*, 2012, pp. 7–12.

[108] Jonathan Vestin, Peter Dely, Andreas Kassler, Nico Bayer, Hans Einsiedler, and Christoph Peylo, "Cloudmac: Towards software defined wlans," vol. 16, no. 4, pp. 42–45, feb 2013.

[109] Kok-Kiong Yap, Rob Sherwood, Masayoshi Kobayashi, Te-Yuan Huang, Michael Chan, Nikhil Handigol, Nick McKeown, and Guru Parulkar, "Blueprint for introducing innovation into wireless mobile networks," in *Proceedings of the Second ACM SIGCOMM Workshop on Virtualized Infrastructure Systems and Architectures*, New York, NY, USA, 2010, VISA '10, pp. 25–32, Association for Computing Machinery.

[110] Hassan Ali-Ahmad, Claudio Cicconetti, Antonio de la Oliva, Martin Draxler, Rohit Gupta, Vincenzo Mancuso, Laurent Roullet, and Vincenzo Sciancalepore, "Crowd: An sdn approach for densenets," in *2013 Second European Workshop on Software Defined Networks*, 2013, pp. 25–31.

[111] Federico Tonini, Bahare Masood Khorsandi, Steinar Bjornstad, Raimena Veisllari, and Carla Raffaelli, "C-ran traffic aggregation on latency-controlled ethernet links," *Applied Sciences*, vol. 8, no. 11, 2018.

[112] Chih-lin I, Yannan Yuan, Jinri Huang, Shijia Ma, Chunfeng Cui, and Ran Duan, "Rethink fronthaul for soft RAN," *IEEE Communications Magazine*, vol. 53, no. 9, pp. 82–88, 2015.

[113] Islam Alyafawi, Eryk Schiller, Torsten Braun, Desislava Dimitrova, Andre Gomes, and Navid Nikaein, "Critical issues of centralized and cloudified LTE-FDD Radio Access Networks," in *2015 IEEE International Conference on Communications (ICC)*, 2015, pp. 5523–5528.

[114] Zhen Kong, Jiayu Gong, Cheng-Zhong Xu, Kun Wang, and Jia Rao, "eBase: A baseband unit cluster testbed to improve energy-efficiency for cloud radio access network," in *2013 IEEE International Conference on Communications (ICC)*, 2013, pp. 4222–4227.

[115] Tshiamo Sigwele, Atm Shafiul Alam, Prashant Pillai, and Y. Fun Hu, "Evaluating Energy-Efficient Cloud Radio Access Networks for 5G," in *2015 IEEE International Conference on Data Science and Data Intensive Systems*, 2015, pp. 362–367.

[116] Karthikeyan Sundaresan, Mustafa Y. Arslan, Shailendra Singh, Sampath Rangarajan, and Srikanth V. Krishnamurthy, "FluidNet: A Flexible Cloud-Based Radio Access Network for Small Cells," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 915–928, 2016.

[117] Mukundan Madhavan, Parul Gupta, and Malolan Chetlur, "Quantifying multiplexing gains in a Wireless Network Cloud," in *2012 IEEE International Conference on Communications (ICC)*, 2012, pp. 3212–3216.

[118] Pablo Caballero, Albert Banchs, Gustavo de Veciana, and Xavier Costa-Perez, "Multi-Tenant Radio Access Network Slicing: Statistical Multiplexing of Spatial Loads," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 3044–3058, 2017.

[119] Mugen Peng, Yuan Li, Jiamo Jiang, Jian Li, and Chonggang Wang, "Heterogeneous cloud radio access networks: a new perspective for enhancing spectral and energy efficiencies," *IEEE Wireless Communications*, vol. 21, no. 6, pp. 126–135, 2014.

[120] Manoj Muniswamaiah, Tilak Agerwala, and Charles C. Tappert, "A survey on cloudlets, mobile edge, and fog computing," in *2021 8th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2021 7th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, 2021, pp. 139–142.

[121] Tuyen X. Tran, Abolfazl Hajisami, Parul Pandey, and Dario Pompili, "Collaborative mobile edge computing in 5g networks: New paradigms, scenarios, and challenges," *IEEE Communications Magazine*, vol. 55, no. 4, pp. 54–61, 2017.

[122] Maurantonio Caprolu, Roberto Di Pietro, Flavio Lombardi, and Simone Raponi, "Edge computing perspectives: Architectures, technologies, and open security issues," in *2019 IEEE International Conference on Edge Computing (EDGE)*, 2019, pp. 116–123.

[123] Mugen Peng, Shi Yan, Kecheng Zhang, and Chonggang Wang, "Fog-computing-based radio access networks: issues and challenges," *IEEE Network*, vol. 30, no. 4, pp. 46–53, 2016.

[124] Koustabh Dolui and Soumya Kanti Datta, "Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing," in *2017 Global Internet of Things Summit (GIoTS)*, 2017, pp. 1–6.

[125] Yuan-Yao Shih, Wei-Ho Chung, Ai-Chun Pang, Te-Chuan Chiu, and Hung-Yu Wei, "Enabling Low-Latency Applications in Fog-Radio Access Networks," *IEEE Network*, vol. 31, no. 1, pp. 52–58, 2017.

[126] TS 23.251 3GPP, *Universal Mobile Telecommunications System (UMTS); Network sharing; Architecture and functional description*, Rel. 6, March 2006.

[127] TR 22.852 3GPP, *Study on Radio Access Network (RAN) sharing enhancements*, Rel. 13, September 2014.

[128] 3GPP TS 23.682, *Architecture enhancements to facilitate communications with packet data networks and applications*, Rel. 14, December 2016.

[129] TR 23.799 3GPP, *Study on Architecture for Next Generation System*, Rel. 14, December 2016.

[130] ITU-T FG-Cloud-TR, *Focus Group Technical Report on Cloud Computing*, February 2012.

[131] Joao Soares, Joao Aparicio, and Susana Sargento, "Dynamic strategies for the optimal embedding of virtual infrastructures," in *2017 IEEE Symposium on Computers and Communications (ISCC)*, 2017, pp. 272–277.

[132] Yasuhiro Yamasaki and Masayoshi Aritsugi, "A Case Study of IaaS and SaaS in a Public Cloud," in *2015 IEEE International Conference on Cloud Engineering*, 2015, pp. 434–439.

[133] Gang Wang, Gang Feng, Wei Tan, Shuang Qin, Ruihan Wen, and SanShan Sun, "Resource Allocation for Network Slices in 5G with Network Resource Pricing," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, 2017, pp. 1–6.

[134] Mathieu Leconte, Georgios S. Paschos, Panayotis Mertikopoulos, and Ulas C. Kozat, "A Resource Allocation Framework for Network Slicing," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, 2018, pp. 2177–2185.

[135] Dario Bega, Marco Gramaglia, Albert Banchs, Vincenzo Sciancalepore, and Xavier Costa-Perez, "A Machine Learning Approach to 5G Infrastructure Market Optimization," *IEEE Transactions on Mobile Computing*, vol. 19, no. 3, pp. 498–512, 2020.

[136] TS 29.522 3GPP, *Technical Specification Group Core Network and Terminals; 5G System; Network Exposure Function Northbound APIs; Stage 3*, Rel. 17, September 2021.

[137] OpenAPI: "OpenAPI Specification Version 3.0.0", https://spec.openapis.org/oas/v3.0.0. (visited on 13/11/2021).

[138] TS 23.501 3GPP, *Technical Specification Group Services and System Aspects; System architecture for the 5G System (5GS); Stage 2*, Rel. 17, December 2021.

[139] Konstantinos Samdanis, Xavier Costa-Perez, and Vincenzo Sciancalepore, "From network sharing to multi-tenancy: The 5G network slice broker," *IEEE Communications Magazine*, vol. 54, no. 7, pp. 32–39, 2016.

[140] Di Zhang, Shahid Mumtaz, Zhenyu Zhou, and Takuro Sato, "Integrating Energy Efficiency mechanism with components selection for massive MIMO based C-RAN," in *2016 IEEE International Conference on Communications Workshops (ICC)*, 2016, pp. 74–79.

[141] TELEFONICA, https://www.telefonica.com/es/web/press-office/-/telefonica-creates-telxius-a-global- telecommunications-infrastructure-company. (visited on 10/02/2016).

[142] Shao-Yu Lien, Shao-Chou Hung, Kwang-Cheng Chen, and Ying-Chang Liang, "Ultra-low-latency ubiquitous connections in heterogeneous cloud radio access networks," *IEEE Wireless Communications*, vol. 22, no. 3, pp. 22–31, 2015.

[143] TS 36.321 3GPP, *Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification*, Rel. 16, September 2021.

[144] Syed Danial Ali Shah, Mark A. Gregory, and Shuo Li, "Cloud-native network slicing using software defined networking based multi-access edge computing: A survey," *IEEE Access*, vol. 9, pp. 10903–10924, 2021.

[145] TS 38.401 3GPP, *Technical Specification Group Radio Access Network; NG-RAN; Architecture description*, Rel. 16, December 2021.

[146] Gunther Auer, Vito Giannini, Claude Desset, Istvan Godor, Per Skillermark, Magnus Olsson, Muhammad Ali Imran, Dario Sabella, Manuel J. Gonzalez, Oliver Blume, and Albrecht Fehske, "How much energy is needed to run a wireless network?," *IEEE Wireless Communications*, vol. 18, no. 5, pp. 40–49, 2011.

[147] Eunsung Oh, Kyuho Son, and Bhaskar Krishnamachari, "Dynamic base station switching-on/off strategies for green cellular networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 5, pp. 2126–2136, 2013.

[148] David Sesto-Castilla, Eduard Garcia-Villegas, George Lyberopoulos, and Eleni Theodoropoulou, "Use of machine learning for energy efficiency in present and future mobile networks," in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, 2019, pp. 1–6.

[149] Mohammed Yazid Lyazidi, Nadjib Aitsaadi, and Rami Langar, "Dynamic resource allocation for Cloud-RAN in LTE with real-time BBU/RRH assignment," in *2016 IEEE International Conference on Communications (ICC)*, 2016, pp. 1–6.

[150] M. Khan, R.S. Alhumaima, and H.S. Al-Raweshidy, "Reducing energy consumption by dynamic resource allocation in C-RAN," in *2015 European Conference on Networks and Communications (EuCNC)*, 2015, pp. 169–174.

[151] Feng Hu, Bing Chen, and Kun Zhu, "Full spectrum sharing in cognitive radio networks toward 5g: A survey," *IEEE Access*, vol. 6, pp. 15754–15776, 2018.

[152] W. S. H. M. W. Ahmad, N. A. M. Radzi, F. S. Samidi, A. Ismail, F. Abdullah, M. Z. Jamaludin, and M. N. Zakaria, "5g technology: Towards dynamic spectrum sharing using cognitive radio networks," *IEEE Access*, vol. 8, pp. 14460–14488, 2020.

[153] Lingjie Duan, Lin Gao, and Jianwei Huang, "Cooperative spectrum sharing: A contract-based approach," *IEEE Transactions on Mobile Computing*, vol. 13, no. 1, pp. 174–187, 2014.

[154] Lorela Cano, Antonio Capone, Giuliana Carello, Matteo Cesana, and Mauro Passacantando, "Cooperative Infrastructure and Spectrum Sharing in Heterogeneous Mobile Networks," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 10, pp. 2617–2629, 2016.

[155] 3GPP TS 36.213, *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures*, Rel. 12, February 2015.

[156] Jiancun Fan, Qinye Yin, Geoffrey Ye Li, Bingguang Peng, and Xiaolong Zhu, "MCS Selection for Throughput Improvement in Downlink LTE Systems," in *2011 Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN)*, 2011, pp. 1–5.

[157] Dario Sabella, Antonio de Domenico, Efstathios Katranaras, Muhammad Ali Imran, Marco di Girolamo, Umer Salim, Massinissa Lalam, Konstantinos Samdanis, and Andreas Maeder, "Energy Efficiency Benefits of RAN-as-a-Service Concept for a Cloud-Based 5G Mobile Network Infrastructure," *IEEE Access*, vol. 2, pp. 1586–1597, 2014.

[158] iJoin project, URL: https://www.ict-ijoin.eu/ (visited on 10/02/2016).

[159] Z. Han, D. Niyato, W. Saad, Tamer Baar, and Are Hjrungnes, *Game Theory in Wireless and Communication Networks: Theory, Models and Applications*, Cambridge, U.K.: Cambridge Univ. Press, October 2011.

[160] Antonio de la Oliva, Jose Alberto Hernandez, David Larrabeiti, and Arturo Azcorra, "An overview of the CPRI specification and its application to C-RAN-based LTE scenarios," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 152–159, 2016.

[161] Mikhail Gerasimenko, Dmitri Moltchanov, Roman Florea, Sergey Andreev, Yevgeni Koucheryavy, Nageen Himayat, Shu-Ping Yeh, and Shilpa Talwar, "Cooperative Radio Resource Management in Heterogeneous Cloud Radio Access Networks," *IEEE Access*, vol. 3, pp. 397–406, 2015.

[162] EUROSTAT, https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Electricity_price_statistics. (visited on 20/10/2016).

[163] TS 36.101 3GPP, *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception*, Rel. 8, July 2010.

[164] Wikipedia, https://en.wikipedia.org/wiki/Low-definition_television. (visited on 13/11/2021).

[165] Google, https://support.google.com/youtube/answer/2853702?hl=en#zippy=%2Cp. (visited on 13/11/2021).

[166] TR 21.915 3GPP, *Release 15 Description; Summary of Rel-15 Work Items*, Rel. 15, September 2019.

[167] Ummy Habiba and Ekram Hossain, "Auction Mechanisms for Virtualization in 5G Cellular Networks: Basics, Trends, and Open Challenges," *IEEE Communications Surveys Tutorials*, vol. 20, no. 3, pp. 2264–2293, 2018.

[168] Bin Han, Ji Lianghai, and Hans D. Schotten, "Slice as an Evolutionary Service: Genetic Optimization for Inter-Slice Resource Management in 5G Networks," *IEEE Access*, vol. 6, pp. 33137–33147, 2018.

[169] Gaofei Sun, Xinxin Feng, Xiaohua Tian, Xiaoying Gan, Youyun Xu, Xinbing Wang, and Mohsen Guizani, "Coalitional Double Auction for Spatial Spectrum Allocation in Cognitive Radio Networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 6, pp. 3196–3206, 2014.

[170] Peng Lin, Xiaojun Feng, Qian Zhang, and Mounir Hamdi, "Groupon in the Air: A three-stage auction framework for Spectrum Group-buying," in *2013 Proceedings IEEE INFOCOM*, 2013, pp. 2013–2021.

[171] Yingxiao Zhang, Suzhi Bi, and Ying Jun Zhang, "A two-stage spectrum leasing optimization framework for virtual mobile network operators," in *2016 IEEE International Conference on Communication Systems (ICCS)*, 2016, pp. 1–6.

[172] Pablo Caballero, Albert Banchs, Gustavo De Veciana, and Xavier Costa-Perez, "Network Slicing Games: Enabling Customization in Multi-Tenant Mobile Networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 2, pp. 662–675, 2019.

[173] Mira Morcos, Tijani Chahed, Lin Chen, Jocelyne Elias, and Fabio Martignon, "A two-level auction for C-RAN resource allocation," in *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2017, pp. 516–521.

[174] Abdallah Jarray and Ahmed Karmouch, "Decomposition Approaches for Virtual Network Embedding With One-Shot Node and Link Mapping," *IEEE/ACM Transactions on Networking*, vol. 23, no. 3, pp. 1012–1025, 2015.

[175] Flavio Esposito, Donato Di Paola, and Ibrahim Matta, "On Distributed Virtual Network Embedding With Guarantees," *IEEE/ACM Transactions on Networking*, vol. 24, no. 1, pp. 569–582, 2016.

[176] Bin Han, Di Feng, and Hans D. Schotten, "A Markov Model of Slice Admission Control," *IEEE Networking Letters*, vol. 1, no. 1, pp. 2–5, 2019.

[177] Sihem Bakri, Bouziane Brik, and Adlen Ksentini, "On using reinforcement learning for network slice admission control in 5G: Offline vs. online," *International Journal of Communication Systems*, vol. 34, no. 7, pp. e4757, 2021.

[178] Jordi Pérez-Romero, Juan Sánchez-González, Oriol Sallent, and Ramon Agustí, "On Learning and Exploiting Time Domain Traffic Patterns in Cellular Radio Access Networks," in *Machine Learning and Data Mining in Pattern Recognition*, Petra Perner, Ed., Cham, 2016, pp. 501–515, Springer International Publishing.

[179] Utpal Paul, Luis Ortiz, Samir R. Das, Giordano Fusco, and Milind Madhav Buddhikot, "Learning probabilistic models of cellular network traffic with applications to resource management," in *2014 IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN)*, 2014, pp. 82–91.

[180] Yunjuan Zang, Feixiang Ni, Zhiyong Feng, Shuguang Cui, and Zhi Ding, "Wavelet transform processing for cellular traffic prediction in machine learning

networks," in *2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, 2015, pp. 458–462.

[181] METIS-II, *Performance evaluation framework*, Deliverable D2.1, p. 10, January 2016.

[182] Albert Banchs, Gustavo de Veciana, Vincenzo Sciancalepore, and Xavier Costa-Perez, "Resource Allocation for Network Slicing in Mobile Networks," *IEEE Access*, vol. 8, pp. 214696–214706, 2020.

[183] Cristina Marquez, Marco Gramaglia, Marco Fiore, Albert Banchs, and Xavier Costa-Perez, "How Should I Slice My Network? A Multi-Service Empirical Evaluation of Resource Sharing Efficiency," New York, NY, USA, 2018, pp. 191–206, Association for Computing Machinery.

[184] Anteneh A. Gebremariam, Mainak Chowdhury, Andrea Goldsmith, and Fabrizio Granelli, "Resource pooling via dynamic spectrum-level slicing across heterogeneous networks," in *2017 14th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2017, pp. 818–823.

[185] Josep Xavier Salvat, Lanfranco Zanzi, Andres Garcia-Saavedra, Vincenzo Sciancalepore, and Xavier Costa-Perez, "Overbooking Network Slices through Yield-Driven End-to-End Orchestration," in *Proceedings of the 14th International Conference on Emerging Networking EXperiments and Technologies*, New York, NY, USA, 2018, CoNEXT '18, pp. 353–365, Association for Computing Machinery.

[186] Irene Vila, Oriol Sallent, Anna Umbert, and Jordi Perez-Romero, "An Analytical Model for Multi-Tenant Radio Access Networks Supporting Guaranteed Bit Rate Services," *IEEE Access*, vol. 7, pp. 57651–57662, 2019.

[187] Dario Bega, Marco Gramaglia, Marco Fiore, Albert Banchs, and Xavier Costa-Perez, "DeepCog: Cognitive Network Management in Sliced 5G Networks with Deep Learning," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, 2019, pp. 280–288.

[188] J. Macqueen, "Some methods for classification and analysis of multivariate observations," in *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, 1967, pp. 281–297.

[189] Simon Haykin, *Neural Networks and Learning Machines - 3rd ed.*, Prentice Hall, 2008.

[190] Stephen Bates, Trevor Hastie, and Robert Tibshirani, "Cross-validation: what does it estimate and how well does it do it?," 2021.

[191] Kaifeng Bu, Yaobo Zhang, and Qingxian Luo, "Depth-Width Trade-offs for Neural Networks via Topological Entropy," 2020.

[192] August Betzler, Daniel Camps-Mur, Eduard Garcia-Villegas, Ilker Demirkol, and Joan Josep Aleixendri, "SODALITE: SDN Wireless Backhauling for Dense 4G/5G Small Cell Networks," *IEEE Transactions on Network and Service Management*, vol. 16, no. 4, pp. 1709–1723, 2019.

[193] Mourice O. Ojijo and Olabisi E. Falowo, "A survey on slice admission control strategies and optimization schemes in 5g network," *IEEE Access*, vol. 8, pp. 14977–14990, 2020.

[194] Mohammed Amine Togou, Ting Bi, Kapal Dev, Kevin McDonnell, Aleksandar Milenovic, Hitesh Tewari, and Gabriel-Miro Muntean, "Dbns: A distributed blockchain-enabled network slicing framework for 5g networks," *IEEE Communications Magazine*, vol. 58, no. 11, pp. 90–96, 2020.

[195] Dario Bega, Marco Gramaglia, Andres Garcia-Saavedra, Marco Fiore, Albert Banchs, and Xavier Costa-Perez, "Network slicing meets artificial intelligence: An ai-based framework for slice management," *IEEE Communications Magazine*, vol. 58, no. 6, pp. 32–38, 2020.