Simulation Models and Advanced Management Techniques for 5G & Beyond Radio Access Networks

Politecnico di Bari

Department of Electrical and Information Engineering
ELECTRICAL AND INFORMATION ENGINEERING
Ph.D. Program
SSD: ING-INF/03–TELECOMMUNICATIONS

Final Dissertation

Simulation Models and Advanced Management Techniques for 5G & Beyond Radio Access Networks

---

by

Sergio MARTIRADONNA

Supervisor:

Prof. Gennaro BOGGIA

Coordinator of Ph.D. Program:

Prof. Mario CARPENTIERI

Course n°34, 01/11/2018-31/10/2021

Politecnico di Bari

LIBERATORIA PER L'ARCHIVIAZIONE DELLA TESI DI DOTTORATO

Al Magnifico Rettore
del Politecnico di Bari

Il sottoscritto Sergio Martiradonna nato a Grumo Appula (BA) il 09/10/1992

residente a Bari in via delle medaglie d'oro,11 e-mail sergio.martiradonna@gmail.com

iscritto al 3° anno di Corso di Dottorato di Ricerca in Ingegneria Elettrica e dell'Informazione ciclo XXXIV

ed essendo stato ammesso a sostenere l'esame finale con la prevista discussione della tesi dal titolo:

Simulation Models and Advanced Management Techniques for 5G &Beyond Radio Access Networks

DICHIARA

1) di essere consapevole che, ai sensi del D.P.R. n. 445 del 28.12.2000, le dichiarazioni mendaci, la falsità negli atti e l'uso di atti falsi sono puniti ai sensi del codice penale e delle Leggi speciali in materia, e che nel caso ricorressero dette ipotesi, decade fin dall'inizio e senza necessità di nessuna formalità dai benefici conseguenti al provvedimento emanato sulla base di tali dichiarazioni;
2) di essere iscritto al Corso di Dottorato di ricerca in Ingegneria Elettrica e dell'Informazione ciclo XXXIV, corso attivato ai sensi del "*Regolamento dei Corsi di Dottorato di ricerca del Politecnico di Bari"*, emanato con D.R. n.286 del 01.07.2013;
3) di essere pienamente a conoscenza delle disposizioni contenute nel predetto Regolamento in merito alla procedura di deposito, pubblicazione e autoarchiviazione della tesi di dottorato nell'Archivio Istituzionale ad accesso aperto alla letteratura scientifica;
4) di essere consapevole che attraverso l'autoarchiviazione delle tesi nell'Archivio Istituzionale ad accesso aperto alla letteratura scientifica del Politecnico di Bari (IRIS-POLIBA), l'Ateneo archivierà e renderà consultabile in rete (nel rispetto della Policy di Ateneo di cui al D.R. 642 del 13.11.2015) il testo completo della tesi di dottorato, fatta salva la possibilità di sottoscrizione di apposite licenze per le relative condizioni di utilizzo (di cui al sito http://www.creativecommons.it/Licenze), e fatte salve, altresì, le eventuali esigenze di "embargo", legate a strette considerazioni sulla tutelabilità e sfruttamento industriale/commerciale dei contenuti della tesi, da rappresentarsi mediante compilazione e sottoscrizione del modulo in calce (Richiesta di embargo);
5) che la tesi da depositare in IRIS-POLIBA, in formato digitale (PDF/A) sarà del tutto identica a quelle **consegnate**/inviate/da inviarsi ai componenti della commissione per l'esame finale e a qualsiasi altra copia depositata presso gli Uffici del Politecnico di Bari in forma cartacea o digitale, ovvero a quella da discutere in sede di esame finale, a quella da depositare, a cura dell'Ateneo, presso le Biblioteche Nazionali Centrali di Roma e Firenze e presso tutti gli Uffici competenti per legge al momento del deposito stesso, e che di conseguenza va esclusa qualsiasi responsabilità del Politecnico di Bari per quanto riguarda eventuali errori, imprecisioni o omissioni nei contenuti della tesi;
6) che il contenuto e l'organizzazione della tesi è opera originale realizzata dal sottoscritto e non compromette in alcun modo i diritti di terzi, ivi compresi quelli relativi alla sicurezza dei dati personali; che pertanto il Politecnico di Bari ed i suoi funzionari sono in ogni caso esenti da responsabilità di qualsivoglia natura: civile, amministrativa e penale e saranno dal sottoscritto tenuti indenni da qualsiasi richiesta o rivendicazione da parte di terzi;
7) che il contenuto della tesi non infrange in alcun modo il diritto d'Autore né gli obblighi connessi alla salvaguardia di diritti morali od economici di altri autori o di altri aventi diritto, sia per testi, immagini, foto, tabelle, o altre parti di cui la tesi è composta.

Bari, 27/12/2021 Firma _____

Il/La sottoscritto, con l'autoarchiviazione della propria tesi di dottorato nell'Archivio Istituzionale ad accesso aperto del Politecnico di Bari (POLIBA-IRIS), pur mantenendo su di essa tutti i diritti d'autore, morali ed economici, ai sensi della normativa vigente (Legge 633/1941 e ss.mm.ii.),

CONCEDE

- al Politecnico di Bari il permesso di trasferire l'opera su qualsiasi supporto e di convertirla in qualsiasi formato al fine di una corretta conservazione nel tempo. Il Politecnico di Bari garantisce che non verrà effettuata alcuna modifica al contenuto e alla struttura dell'opera.
- al Politecnico di Bari la possibilità di riprodurre l'opera in più di una copia per fini di sicurezza, back-up e conservazione.

Bari, 27/12/2021 Firma _____

Politecnico di Bari

Department of Electrical and Information Engineering
ELECTRICAL AND INFORMATION ENGINEERING
Ph.D. Program
SSD: ING-INF/03–TELECOMMUNICATIONS

Final Dissertation

Simulation Models and Advanced Management Techniques for 5G &Beyond Radio Access Networks

by

Sergio Martiradonna

Referees:

Prof. Paolo Dini

Prof. Fernando Velez

Supervisors:

Prof. Gennaro Boggia

Coordinator of Ph.D Program:

Prof. Mario Carpentieri

Course n°34, 01/11/2018-31/10/2021

To my beloved Graces, magnificent examples of Love, Strength, and Truth

Abstract

5^{th} Generation (5G) is providing a significant transformation in the mobile network landscape. It introduces flexible and heterogeneous capabilities to harmoniously blend numerous technical components since a variety of advanced services are being developed, each one entailing different requirements. For this reason, 5G does not have a single air interface, but rather a family of air interfaces to adequately address specific use cases, all plugged into a common framework. Nonetheless, the effective management of such a broad diversity is an extremely ambitious goal to accomplish. To this end, this work pursues the goal of investigating several cutting-edge management techniques and simulation models for 5G & Beyond Radio Access Networks (RANs). Specifically, this thesis presents an open-source system-level tool to model the key elements of the 5G RAN and support the performance analysis of reference scenarios. Moreover, it examines NarrowBand IoT (NB-IoT), which is usually regarded as a promising radio access technology to meet the requirements of the 5G & Beyond development for the Internet of Things (IoT). Finally, it addresses the RAN Slicing problem leveraging Edge Computing and Artificial Intelligence (AI), which promise to turn future mobile networks into service- and radio-aware infrastructures.

Contents

List of Figures

List of Tables

List of Acronyms

3GPP	3rd Generation Partnership Project
4G	4th Generation
5G	5th Generation
AI	Artificial Intelligence
AM	Acknowledged Mode
AMC	Adaptive Modulation and Coding
ANN	Artificial Neural Network
AP	Access Point
API	Application Programming Interface
ARQ	Automatic Repeat reQuest
AT	Autonomous Tram
B5G	Beyond 5G
BLE	Bluetooth low energy
BLER	BLock Error Rate
BWP	BandWidth Part
C-RAN	Cloud - Radio Access Network
CDF	Cumulative Distribution Function
CNN	Convolutional Neural Network
CP	Cyclic Prefix
CQI	Channel Quality Indicator
CSI	Channel State Information
D2D	Device-to-Device
DC	Dual-Connectivity
DCI	Downlink Control Information
DDPG	Deep Deterministic Policy Gradient
DL	Deep Learning
DQN	Deep Q-Network
DRL	Deep Reinforcement Learning
E2E	End-to-End
ECDF	Empirical Cumulative Distribution Function
EEG	ElectroEncephaloGraph

EI	Edge Intelligence
eMBB	enhanced Mobile BroadBand
FDD	Frequency Division Duplex
FEC	Forward Error Correction
FIFO	First-In First-Out
FWA	Fixed Wireless Access
GCS	Gateway Computing Server
GSM	Global System for Mobile communications
GTC	Generalized Tonic-Clonic
HARQ	Hybrid Automatic Repeat reQuest
HSPA	High-Speed Packet Access
IaaS	Infrastructure as a Service
IAB	Integrated Access and Backhaul
IoT	Internet of Things
IP	Infrastructure Provider
ISD	Inter-Site Distance
JSDM	Joint Spatial Division and Multiplexing
KPI	Key Performance Indicator
L2S	Link-To-System
LDPC	Low Density Parity Check
LEO	Low Earth Orbit
LPWAN	Low Power Wide Area Network
LSTM	Long Short-Term Memory
LTE	Long Term Evolution
LTE-A	LTE-Advanced
M-LWDF	Modified-Largest Weighted Delay First
M2M	Machine-to-Machine
MAC	Media Access Control
MBSFN	Multicast Broadcast Single Frequency Network
MCE	Mobile Cloud Entity
MCL	Maximum Coupling Loss
MCS	Modulation and Coding Scheme
MDP	Markov Decision Process
MEC	Multi-access Edge Computing
MIB-NB	Master Information Block - NarrowBand
MIESM	Mutual Information Effective SINR Mapping
MIMO	Multiple-Input Multiple-Output
ML	Machine Learning

mMTC	massive Machine-Type Communications
MRC	Maximum Ratio Combining
MSE	Mean Square Error
MTC	Machine-Type Communication
NB-IoT	NarrowBand IoT
NF	Network Function
NFV	Network Function Virtualization
NOMA	Non Orthogonal Multiple Access
NPBCH	Narrowband Physical Broadcast Channel
NPDCCH	Narrowband Physical Downlink Control Channel
NPDSCH	Narrowband Physical Downlink Shared Channel
NPRACH	Narrowband Physical Random Access Channel
NPSS	Narrowband Primary Synchronization Signal
NPUSCH	Narrowband Physical Uplink Shared Channel
NR	New Radio
NSSS	Narrowband Secondary Synchronization Signal
NTN	Non Terrestrial Network
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
PaaS	Platform as a Service
PDCP	Packet Data Convergence Protocol
PLR	Packet Loss Ratio
PMI	Precoding Matrix Indicator
PRI	Pairwise Reordering Improvement
QoE	Quality of Experience
QoS	Quality of Service
RACH	Random Access CHannel
RAN	Radio Access Network
RAO	Random Access Opportunity
RAR	Random Access Response
RAT	Radio Access Technology
RB	Resource Block
RBP	Resource Block Pool
ReLU	Rectified Linear Unit
RG	Residential Gateway
RI	Rank Indicator
RL	Reinforcement Learning
RLC	Radio Link Control

RMSE	Root Mean Square Error
RR	Round-Robin
RRM	Radio Resource Management
RS-ISP	RAN Slicing with Inter Slice Protection
RS-NOISP	RAN Slicing with NO Inter Slice Protection
RSU	Road Side Unit
RU	Resource Unit
RZF	Regularized Zero-Forcing
SatCom	Satellite Communication
SC-FDMA	Single-Carrier Frequency Division Multiple Access
SDAP	Service Data Adaptation Protocol
SDN	Software-Defined Networking
SINR	Signal to Interference plus Noise Ratio
SLA	Service Level Agreement
SNR	Signal to Noise Ratio
SRTA-PI	Separate Receive and Training Antennas with Polynomial Interpolation
STA	STAtion
SUDEP	Sudden Unexpected Death in EPilepsy
TBS	Transport Block Size
TDD	Time Division Duplex
TM	Transparent Mode
TM	Transmission Mode
TNT	Tenant
TTI	Transmission Time Interval
UAV	Unmanned Aerial Vehicle
UE	User Equipment
UM	Uncknowledged Mode
URLLC	Ultra-Reliable Low-Latency Communications
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VoIP	Voice over IP
WBAN	Wireless Body Area Network
WLAN	Wireless Local-Area Network

Personal Scientific Contributions

The scientific contributions published during the PhD course are listed in the following. Additionally, a manuscripts currently in the process of submission is included.

International Journals

- S. Martiradonna, G. Piro, and G. Boggia, "On the Evaluation of the NB-IoT Random Access Procedure in Monitoring Infrastructures," *Sensors*, vol. 19, no. 14, p. 3237, 2019
- S. Martiradonna, A. Grassi, G. Piro, *et al.*, "5g-air-simulator: An open-source tool modeling the 5g air interface," *Computer Networks*, vol. 173, p. 107 151, 2020, ISSN: 1389-1286. DOI: https://doi.org/10.1016/j.comnet.2020.107151. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128619317359
- S. Martiradonna, A. Grassi, G. Piro, *et al.*, "Understanding the 5G-air-simulator: A tutorial on design criteria, technical components, and reference use cases," *Computer Networks*, vol. 177, p. 107 314, 2020
- S. Martiradonna, G. Cisotto, G. Boggia, *et al.*, "Cascaded WLAN-FWA Networking and Computing Architecture for Pervasive In-Home Healthcare," *IEEE Wireless Communications*, 2021
- D. Tamang, S. Martiradonna, A. Abrardo, *et al.*, "Architecting 5G RAN Slicing for Location Aware Vehicle to Infrastructure Communications: The Autonomous Tram Use Case," *Computer Networks*, 2021
- S. Martiradonna, A. Abrardo, M. Moretti, *et al.*, "Deep Reinforcement Learning-Aided RAN Slicing Enforcement Supporting Latency Sensitive Services in B5G networks," *Internet Technology Letters*, 2021

International Conferences

- S. Martiradonna, A. Grassi, G. Piro, *et al.*, "An Open Source Platform for Exploring NB-IoT System Performance," in *European Wireless 2018; 24th European Wireless Conference*, VDE, 2018, pp. 1–6

- S. Martiradonna, A. Abrardo, M. Moretti, *et al.*, "Architecting RAN Slicing for URLLC: Design Decisions and Open Issues," in *2019 IEEE/ACM 23rd International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, IEEE, 2019, pp. 1–4

- A. Petrosino, G. Sciddurlo, S. Martiradonna, *et al.*, "An Open-Source Tool for Evaluating System-Level Performance of NB-IoT Non-Terrestrial Network," in *Proc. of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Pisa, Italy, 2021

Manuscript in the Process of Submission

- A. Rago, S. Martiradonna, A. Abrardo, *et al.*, "On the use of pervasive intelligence in tenant-driven radio access network slicing for 6g," *Computer Communications*,

Introduction

Legacy mobile technologies exploited one-size-fits-all approaches for the RANs since data traffic was mostly human-driven and with similar key performance metrics. However, the tremendous growth of mobile traffic, as well as the emergence of a huge variety of extremely innovative use cases, impose a wider range of performance requirements, thus requiring a significant level of flexibility and scalability.

In this context, 5G mobile networks emerged to jointly support a large variety of envisioned usage scenarios and support a number of new use cases from vertical industries. For this reason, 5G does not have a single air interface, but rather a family of air interfaces to adequately address specific use cases, all plugged into a common framework. Moreover, to handle future and unanticipated use cases, 5G is being designed with flexibility and extensibility at its core. As a consequence, the integration of different components in the 5G air interface is an extremely ambitious goal to accomplish. Hence, there exists a massive number of research, standardization, and deployment challenges. Among these, this Ph.D. thesis essentially deals with three main topics.

First, it explores system-level simulation tools aiming at guaranteeing long-lasting exploitation of scientific results and offering valid scientific and technological support for the development and the diffusion of modern service platforms built on top of 5G & Beyond communication infrastructures. To this purpose, the research group belonging to the Telematics Lab of the Politecnico di Bari developed an open-source simulation framework for the 5G air interface, namely 5G-air-simulator, which is a valid instrument to study a number of technical components already standardized by the 3rd Generation Partnership Project (3GPP), under investigation by other standardization entities, or recently discussed in the scientific literature. Indeed, the simulator already proved to be a valuable tool for different research activities; therefore, a series of simulation campaigns for typical 5G scenarios are also explored. For each of them, a theoretical description of the related technical components is provided, along with main implementation details, the syntax used

to perform the simulations, the steps to extract major Key Performance Indicators (KPIs), as well as the description of example use cases and related reference results.

Second, this thesis investigates one of the first substantial differences with previous cellular technologies through the study of a technology tailored to the emerging IoT scenario, which truly represents a rupture with the past of the traditional scenario of human-driven mobile traffic. In particular, the NB-IoT radio access technology is a promising standard to meet the requirements of the future 5G & Beyond development for the IoT while reusing the existing mobile infrastructure. For these reasons, it has been successfully implemented in 5G-air-simulator to analyze its performance as well as investigate even more advanced use cases such as Non Terrestrial Networks (NTNs).

Third, and last, the present work seeks to deepen the study and lay the foundations on those technologies that are not yet de-facto established in current network deployments but which represent a very important starting point for the evolution of 5G & Beyond networks, especially following the increase of use cases and services offered through third-party verticals. Specifically, Network Slicing in the RAN is emerging as a valid key enabler to support customized services on the top of shared infrastructure, but not without difficulties due to the unpredictable nature of wireless resources. Based on this premises, novel architectures to realize RAN Slicing are presented, also leveraging both the Multi-access Edge Computing (MEC) paradigm and recent AI techniques.

The remainder of this thesis is structured as follows. Chapter 1 introduces 5G & Beyond Radio Access Networks while focusing on the services and use cases as well as the enabling technologies. Chapter 2 presents the 5G-air-simulator, Chapter 3 thoroughly describes the NB-IoT radio access technology, while Chapter 4 focuses on RAN Slicing. Finally, the future research directions are presented in the Conclusions.

Chapter 1

Introduction to 5G & Beyond Radio Access Networks

The first generation of mobile communication was born around the 1980s, when analog transmissions were the effective means to offer voice services and mobile equipments were large and extremely power-hungry [1]. To make matter worse, roaming was impossible and security simply did not exist. The 1990s saw the introduction of digital transmission technologies with the second generation of mobile communication, whose Global System for Mobile communications (GSM) just scratched the surface. 2G technologies were entirely digital, hence the voice signal was digitized, compressed, and encrypted before the actual transmission. Still, it was only with the introduction of the General Packet Radio System (GPRS) that a packet-switched data connection was introduced. Then, the Enhanced Data rates for GSM Evolution (EDGE) technology further improved the GPRS with higher throughputs. Nonetheless, data and voice services were mutually exclusive. As a consequence, the third generation of mobile communication, also called 3G, was introduced in the early 2000s with the High-Speed Packet Access (HSPA) representing the leap forward to high-quality mobile broadband for experiencing fast internet access. The 4^{th} Generation (4G) era of mobile communication, represented by the Long Term Evolution (LTE) Radio Access Technology (RAT), has followed in the steps of HSPA, providing higher efficiency and further enhanced mobile-broadband experience in terms of higher achievable end-users data rates. Despite their impressive capabilities, LTE and its evolutions (i.e., LTE-Advanced and LTE-Advanced Pro) eventually tend to prioritize one or few KPIs above all the others, hence some emerging use cases cannot be adequately addressed. This led to the urgent necessity of a new generation of mobile communication: 5G.

In the following, Section 1.1 describes a wide range of advanced 5G services and use cases. Then, Section 1.2 provides several details on the new

radio air interface, standardized by 3GPP and developed for adding novel features. Section 1.3 describes several technologies that drove architectural and component disruptive design changes. The key ideas for each technology are described, along with their impact on 5G. Finally, in Section 1.4, the research challenges deeply analyzed in the rest of this thesis are briefly presented.

1.1 An Overview of 5G Services

Probably, the most important difference between 4G and 5G lies in the expected set of use cases, which motivates the corresponding design criteria and performance requirements. Figure 1.1 illustrates some examples of envisioned usage scenarios for 5G and beyond.

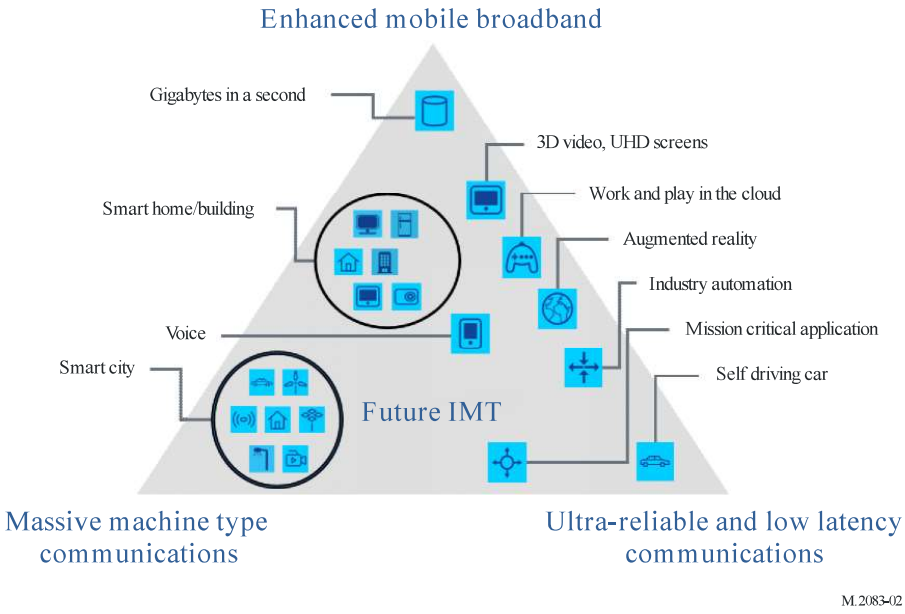


FIGURE 1.1: Examples of typical 5G usage scenarios [2].

In the case of 4G, data traffic was expected to be mostly human-driven [3], with some degree of differentiation among flows, e.g. streaming, file transfer, Voice over IP (VoIP). However, the key performance metrics were similar: throughput, delay, spectral efficiency. This fundamental similarity inspired a one-size-fits-all approach for the radio access network, which is designed to support as many of these requirements as possible.

Conversely, the 5G also deals with many data flows that do not involve any direct human intervention, such as IoT devices and connected vehicles [4]. These may have very different requirements from one another, and also

with respect to 4G. Figure 1.2 illustrates the key capabilities of 5G (IMT-2020), compared with those of 4G (IMT Advanced).

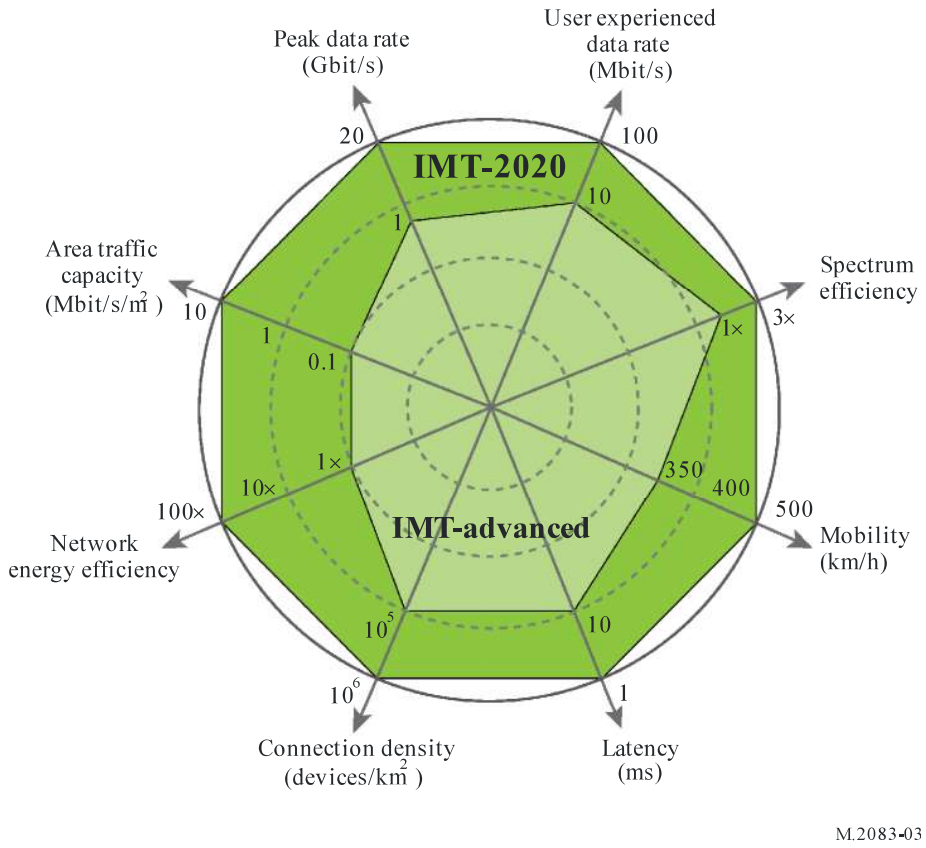


FIGURE 1.2: Enhancement of key capabilities from 4G to 5G [2].

Moreover, even human-based data exchange could have radically different requirements compared to LTE. This is the case, for instance, with virtual reality, where extremely low latency and reduced jitter are definitely more important than maximizing the throughput [5]. Due to this extreme variation of the requirements, it is not possible anymore to support everything with a single solution. For this reason, 5G does not have a single air interface, but rather a family of air interfaces dedicated to specific use cases, all plugged into a common framework [6]. Moreover, to handle future and unanticipated use cases, it is being designed with flexibility and extensibility at its core. In order to give a general idea of the aforementioned requirements' heterogeneity, the rest of this Chapter will first describe some of the main use cases and the corresponding KPIs, and then it will briefly explain the essential 5G air interface features as standardized by 3GPP.

1.1.1 enhanced Mobile Broadband

eMBB generally refers to use cases related to human-based Internet activity that are already common in 4G networks. It embodies practices such as streaming videos, downloading files, browsing the web, performing VoIP or video calls [7]. In most cases, there is a need to download large amounts of data to perform a task. Sometimes, instead, large files need to be uploaded online, e.g., for cloud storage. Either way, the most critical performance metric for the end-user is the perceived connection speed because if it is too low, any operation will take a long time, hence reducing the user experience. Therefore, the connection speed should be as high as possible, and should also remain sufficiently high for a long time and in extended areas, to give a sense of reliability. Besides, from the operator's perspective, it is crucial to offer the service to as many users as possible. This means that the aggregated throughput supported in a given area should be notably high. The most effective way to reach this goal is to make efficient use of the available resources, e.g., by maximizing spectral efficiency. According to some estimates for the coming years, the user data rates required for satisfactory user experiences are in the range 50-100 Mbps, which translates to a traffic density of tens of Gbps/km^2 and a spectral efficiency of hundreds of bps/Hz [8].

1.1.2 massive Machine Type Communications

The IoT is growing steadily, approaching 11 billion devices [9]. Communications involving these connected objects are usually referred to as Machine-Type Communication (MTC), but since their number is so high (and growing), this use case is often called massive Machine-Type Communications (mMTC). Using cellular networks is an attractive option to provide IoT devices with Internet connectivity, because of their widespread deployment and reliable operation. However, current networks are not entirely suited for this use, as they mainly focus on throughput and spectral efficiency. Instead, connected things have different needs.

The majority of these devices are sensors that periodically make measurements and send data to a server, so transmission is largely uplink-dominated. They also need to operate from a battery for a very long time, up to 10 years, because replacing batteries frequently for so many devices is expensive (especially in terms of human work) and inconvenient. Coverage is also an issue: IoT devices may be placed in hard-to-reach places, such as basements or

underground pipes, where cellular signal is weak or missing. To this end, the communication technologies oriented towards IoT applications may need a coverage enhancement, in terms of Maximum Coupling Loss, ranging from 10 to 20 dB, when compared to current networks [10]. Finally, supporting a very large number of devices connected to the same cell may prove difficult, as they may be orders of magnitude more numerous than human users [11].

Chapter 3 will thoroughly present the NB-IoT technology, which has been conceived to addresses the requirements of MTC devices.

1.1.3 Ultra Reliable and Low Latency Communications

In the past, communication networks have been engineered focusing on improving network capacity while overlooking latency or reliability. Achieving Ultra-Reliable Low-Latency Communications (URLLC) represents one of the major challenges facing 5G networks, targeting milliseconds latency (or even lower in Beyond 5G) [12]. The stringent requirements call for a paradigm shift from reactive and centralized networks to massive, low-latency, ultra-reliable, and proactive 5G networks. In particular, a number of enabling technologies (and above all their management and orchestration) are needed to fully realize URLLC: short frame structure (hence short Transmission Time Intervals (TTIs)), Hybrid Automatic Repeat reQuest (HARQ), flexible resource allocation, multi-connectivity, robust channel coding, data replication, edge caching and computing are just some cases in point.

Moreover, this scenario is intended to enable many new applications, while covering both human- and machine-centric communications. For instance, URLLC enables haptic feedback and real-time sensors for remote surgery, Tactile Internet, and wireless control of industrial equipment in Industry 4.0, as well as XR (augmented, virtual and immersive reality). Other examples may include Vehicle-to-Vehicle (V2V) communication involving safety.

1.1.4 Vehicular to Everything

Vehicle-to-Everything (V2X) refers to scenarios where moving vehicles are involved [13]. In this field, numerous exciting applications are possible. One of them is infotainment, where an Internet connection is utilized to provide both entertainment content for passengers and valuable information

for the driver, such as traffic and weather forecasts [14]. The technical requirements would be similar to the eMBB use cases, with the added complication of mobility and possible signal drops. Assisted and autonomous driving, where the vehicle can autonomously perform some maneuvers or adjustments, based on available information about the immediate surrounding, is a different application [15]. In this case, the fundamental features of the underlying communication system are more related to URLLC, to enable timely responses from the assisted driving unit [16]. Very high-speed trains is yet another V2X application, where broadband performance is impaired and shows the use of predictor antennas as a possible solution.

1.1.5 Broadcast/Multicast Services

Even though communications are increasingly becoming personalized, there are additional cases where either a one-to-many or a one-to-all distribution model is appropriate. Moreover, the wireless medium is naturally a shared one, and multicast/broadcast schemes are the best way to take advantage of this feature [17]. Notable applications of this idea include video broadcasting in social events such as concerts and sports matches, local emergency warnings for dangerous weather conditions, and large-scale firmware updates for IoT or automotive devices [18].

The main objective for these scenarios is to achieve extensive and consistent coverage. However, this has to be balanced with throughput, since a higher transmission rate (achieved via higher modulation orders and code rates) makes the reception more challenging for cell-edge users. When multiple cells are involved, they need to be tightly synchronized, and managing multiple partially overlapping broadcast areas may become difficult. Moreover, the network should have means to detect the presence of users interested in the same content, to decide whether a multicast transmission is appropriate, and to establish it on the fly. This is clearly reflected in the fact that LTE has supported a usable broadcasting feature since Release-9, that is the MBSFN [19]. Although 3GPP Release 15 only supports unicast delivery, work is ongoing in Release 16 and the introduction of multicast/broadcast in 5G NR is expected to start in Release 17 for offering new opportunities beyond the capabilities of the previous generation mobile broadcasting [20].

1.1.6 Non Terrestrial Networks

NTNs are expected to have a primary role in 5G & Beyond networks [21]. Thanks to its ubiquity capabilities and the robustness against natural disasters, Satellite Communication (SatCom) fosters network spread in a cost-effective way, by delivering connectivity where telecommunication infrastructures are lacking (i.e., oceans, forests, and deserts). Such a cutting-edge connectivity model can naturally provide backup links in case of network failures. Moreover, it offers additional connections to offload terrestrial networks, while preserving the performance of specific loss or delay-sensitive applications. At the same time, it strongly promotes the scalability of mobile networks, since satellites easing allows possible future further expansions of current 5G deployments. For these reasons, SatCom results particularly effective for MTC scenarios, especially when a huge number of low-cost devices need connectivity in large areas not covered by terrestrial networks. Here, the main challenge is to allow connectivity to a massive number of devices that can have some design constraints or conflicting KPIs, including extended battery lifetime and long transmission range. Interesting deployment for NTNs are expected to use Low Earth Orbit (LEO) satellites because of their reduced cost and experienced round trip time with respect to other kinds of satellites (e.g., GEO) [22].

1.2 3GPP New Radio

The 3GPP is a global standard development organization and has been developing 5G New Radio (NR) over the past few years. The expectation is that 5G NR is a totally new air interface that can operate alongside 4G LTE. However, differently from previous generations, the essential enhancement of 5G with respect to 4G is not only the ability to handle much faster data rates and to provide higher capacity for users. In fact, key NR features include advanced antenna technologies, spectrum flexibility, operation in high-frequency bands, dynamic Time Division Duplex (TDD), and support for low latency. In addition, achieving the 5G expectation, it is essential that 5G NR must be able to deliver numerous and varied services across a different set of devices with different performance and latency needs.

1.2.1 Release 15

Release 15 is the first-ever 5G-compliant standardization work conducted to produce the initial NR specifications. In this initial NR specification, drafted from the beginning of 2017 to 2019, the target objectives were set to specify the functionalities for eMBB and to lay the cornerstones for providing Ultra Reliable and Low Latency Communications.

NR is the first mobile radio technology that is designed to operate on any frequency band between 450 MHz and 52.6 GHz. The lower bands are needed for coverage, while the higher bands will provide high data rates and capacity. Specifically, 3GPP defines two frequency ranges: the first one (Frequency Range 1) covers the frequencies between 450 MHz and 6 GHz range, whereas the second one (Frequency Range 2) refers to the frequencies within the 24.250–52.600 GHz interval. These frequency ranges are commonly referred to as sub-6 GHz and millimeter-wave, respectively. According to the specifications [23], the initial 5G deployments use TDD between 2.5 and 5.0 GHz, Frequency Division Duplex (FDD) below 2.7 GHz, and TDD at millimeter wave at 24–39 GHz. As it happened with the LTE development, it can be expected that there will be several new 5G operating bands and channel bandwidths in forthcoming 3GPP releases.

After the evaluation of new candidates to 5G waveforms, the Orthogonal Frequency Division Multiplexing (OFDM) was chosen, as its performance has been proven in LTE over the last years. However, it has been further optimized to tackle the strict 5G requirements and enabling lower latency compared to the 4G version. In LTE, OFDM subcarriers have a fixed spacing of 15 kHz, and 12 subcarriers in the frequency domain define the basic radio resource, namely the Resource Block (RB). Although also in 5G a RB has 12 subcarriers, 3GPP introduces in the NR standard the idea of flexible numerology, characterized by a set of supported subcarrier spacings and cyclic prefixes [24]. Specifically, Release 15 supports spacing equal to 15 (as in LTE), 30, 60, 120, and 240 kHz, i.e., RBs of 180, 360, 720, 1440, and 2880 kHz width, respectively. While all these spacings support the normal cyclic prefix length, only 30 kHz spacing also supports the extended one, thus accounting for a total of 6 different supported numerologies. It is worth mentioning that 240 kHz subcarrier spacing is only used for synch and it does not support data transmission. [25]

In the dime domain, NR tries to maintain certain backward compatibility with LTE. As a consequence, similarly to LTE, the NR frame is 10 ms long, and it is composed of ten subframes of 1 ms each. Nonetheless, according

to the chosen numerology, each subframe is split into a variable number of slots, which increases with the subcarrier spacing. In accordance, the slot length is smaller for higher spacings. Each slot then contains a fixed number of OFDM symbols: 14 symbols for the normal cyclic prefix length and 12 for the extended one. This architecture enables a flexible NR frame structure, allowing different number of slots per subframe, as well as varying OFDM symbol and slot lengths, as shown in Figure 1.3.

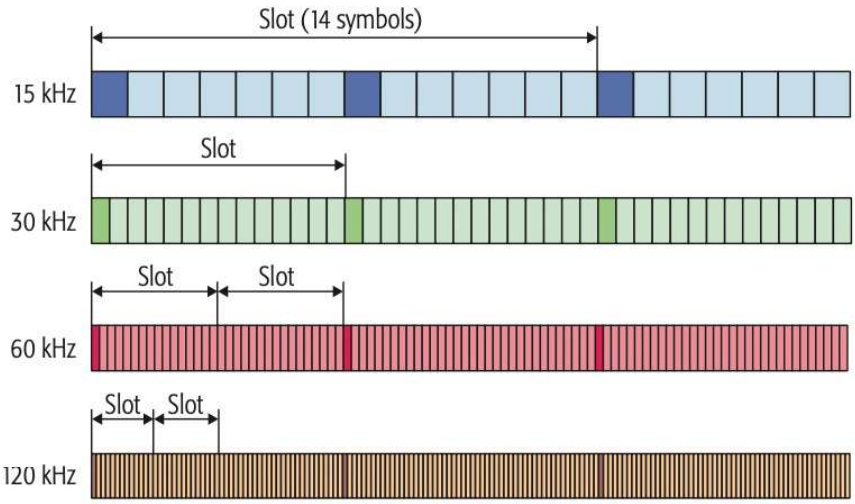


FIGURE 1.3: Scalable OFDM slots ensuring symbol-wise and slot-wise alignment in time domain [26].

To address scenarios characterized by rapid per-cell traffic variations, NR defines dynamic TDD, that is the possibility for dynamic assignment of resources between the downlink and the uplink transmission directions. In other words, the number of uplink and downlink slots in a frame may be changed according to the traffic demands of downlink and uplink directions. In addition, the resource scheduler, which is in charge of conduct this dynamic assignment, works on a per-slot basis, instead of the per-subframe basis typical of LTE, hence with a finer grain.

An additional level of flexibility in NR is achieved with the concept of BWP (see Figure 1.4), which is a subset of the total bandwidth of a cell. In particular, a user can be configured to support one or multiple BWPs, even though only one can be active.

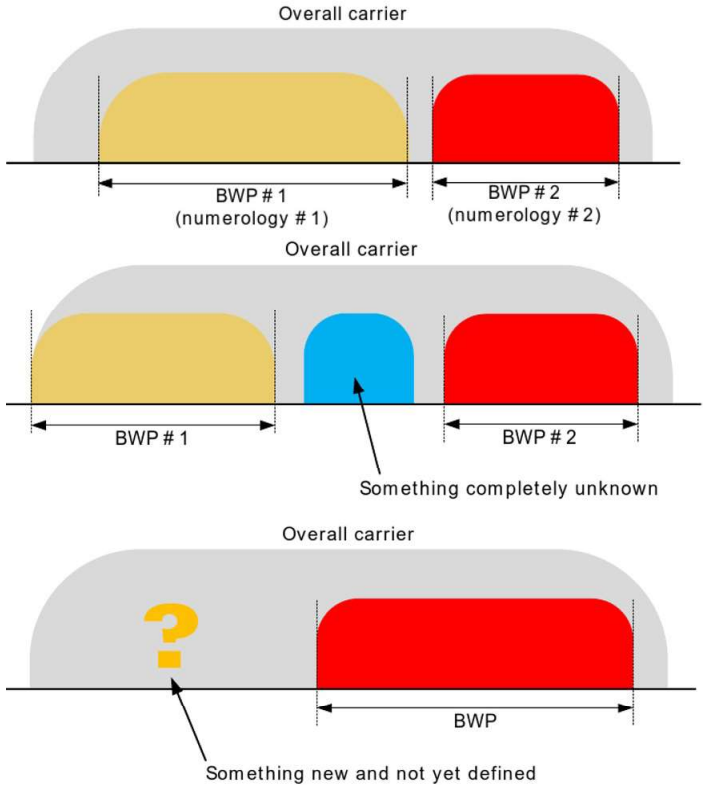


FIGURE 1.4: Use of BWP to enhance 5G flexibility [27].

This is done essentially for three main reasons. First, in order to maintain the hardware of the user devices at a reasonable level of complexity, since operating bandwidth of NR is much higher compared to LTE (up to 100 MHz and 400 MHz for sub-6 GHz and millimeter-wave, respectively). Second, for multiplexing different numerologies in the frequency domain, in order to support various traffic types with different requirements. Third, visionarily, to permit the coexistence of different, unknown, and/or still unspecified technologies.

Thanks to many of the features discussed above, Release 15 also supports and assists usage scenarios for mission-critical services that require extremely low latency and high reliability. As mentioned earlier, low latency is implemented by using a wide subcarrier spacing and reducing the number of OFDM symbols used for data assignment, e.g., a mini-slot can be used to support these services. With reference to the latter example, NR also defined procedures to enable what is called punctured scheduling in the downlink direction [28]. Specifically, low latency may be achieved by puncturing the resources already assigned to other traffics while informing the affected users, in case of a sudden need for resources by a prioritized flow.

On the other hand, to implement high reliability, new modulation and coding schemes are specified to support even lower signal ratios. As a matter of fact, while LTE uses Turbo and convolutional coding, NR adopts Polar Codes for control channels and Low Density Parity Check (LDPC) coding for data channels, which can offer lower complexity, especially at higher code rates, and better performance for small packet sizes. In addition, while LTE can utilize QPSK, 16QAM, and 64QAM modulation schemes, NR may use up to 256QAM, hence increasing throughput and spectral efficiency.

Massive MIMO is another of the key enabling technologies for 5G and it has been part of NR specifications and deployments from the beginning. Differently from the MIMO systems in current 4G standards, Massive MIMO is based on 2D active antenna arrays with a large number of antennas at base stations. This bidimensional structure implies that the radio signal on both vertical and horizontal planes can be controlled simultaneously through a mechanism called 3D beamforming, which can increase spectral efficiency and network coverage substantially. These advancements allow using coding techniques for significantly mitigating the interference between nodes. Nevertheless, such benefits can only be guaranteed if perfect channel knowledge is available at the base station. For this reason, several design challenges need to be considered to implement Massive MIMO in practical systems, as deeply discussed in Chapter 2. Release 15 supports up to 256 antenna elements on base stations and up to 32 antenna elements on terminals. With this configuration, the downlink supports single user MIMO transmission with up to 8 layers and multi-user MIMO transmission with up to 12 layers, whereas the uplink supports single-user MIMO transmission with up to 4 layers. It is important to note that Massive MIMO and beamforming are assumed throughout the specifications not only for data transmission but also for several other aspects, e.g., reference signal structure, beam management, initial access, scheduling, and HARQ retransmission.

1.2.2 Release 16

A big part of the focus of Release 16 is addressing more vertical segments with respect to the use case expected by Release 15, e.g., transportation industry, factory automation, and power distribution, hence covering V2X and mMTC. First of all, Release 16 provided several enhancements to already standardized features, like MIMO with Multiple Transmission Point(Multi-TRP) for the provision of high reliability and robustness of connection thanks

to the increased diversity, and a two-step RACH, for reducing the delay of the traditional four-message random access operation. Second, it introduced completely new topics, i.e., Integrated Access and Backhaul (IAB) [29] and NR-Unlicensed (NR-U) [30]. IAB allows part of the radio access spectrum resources to be used for backhaul transmission, hence enabling a cost-effective deployment option, especially in contexts where a fiber infrastructure is lacking. Instead, the possibility offered by enabling 5G operations in unlicensed spectrum with NR-U, allows to achieve coverage extension as well as to support higher bandwidth operation, hence boosting the performance. At the same time, fair coexistence can be ensured with the wireless technologies which have been already deployed in the 5 GHz band, e.g., Wi-Fi and LTE-based Licensed Assisted Access.

1.2.3 Release 17

As for the key directions for Release 17, they are mainly the work on frequency bands higher than 52.6 GHz and up to 114 GHz, as well as Coverage Enhancement [31]. Moreover, work items include solutions for NR to support NTNs and Unmanned Aerial Vehicles (UAVs) [22]. In addition, several other themes include broadcast and multicast services, enhancements to support edge computing, sidelink relay, support for Multi-Subscriber Identity Module devices, and enhancement for private networks and Quality of Experience (QoE).

It is worth noting that the timeline of this new release has been affected by the COVID-19 pandemic, which resulted in a three-month shift of schedule [32].

1.3 5G & Beyond Enabling Technologies

1.3.1 NFV and SDN

Legacy networks mostly rely on proprietary appliances as well as various network devices that are usually purpose-built. This led to the network ossification problem, which prevents the operation of service additions and network upgrades [33]. In order to cope with this issue (and also reduce CAPEX and OPEX), virtualization has emerged as an approach to decouple the software networking processing and applications from their supported hardware, hence allowing network services to be softwarized. Leveraging

virtualization technologies, ETSI proposed Network Function Virtualization (NFV) to virtualize the Network Functions (NFs) that were previously implemented in proprietary dedicated hardware [34]. In other words, NFV is the relocation and management of NFs (e.g., firewall, NAS, DHCP server, proxy, gateway) in general-purpose devices.

In parallel with NFV rose the Software-Defined Networking (SDN), which is a recent trend in communications networking whereby the behavior of network equipments is controlled by a logically centralized controller. Note that the SDN controllers can retrieve useful information from network elements through standardized protocols (i.e., OpenFlow, RESTCONF, etc.) [35]. In essence, SDN decouples the data and control planes by using software components responsible for managing the control plane, therefore reducing hardware constraints. As a consequence, it allows a split between control and data planes, hence introducing swiftness and flexibility in 5G networks in a way that would have been unthinkable before [36].

1.3.2 Multi-Access Edge Computing

The traditional centralized network architecture cannot support the exponentially growing traffic due to the heavy burden on the backhaul links and long latency. Furthermore, mobile users mostly have limited storage and processing capacity, hence running compute-intensive applications on resource-constrained users is still an important issue [37]. As a consequence, the MEC paradigm emerged as a promising solution to provide cloud computing and caching capabilities at the edge of cellular networks. Particularly, MEC servers are deployed at the network edge to offer intensive computing and memory capabilities in the proximity of end-users, while guaranteeing reliable and low-latency communication to the new heavy demanding and real-time services [38]. According to ETSI-MEC specifications [39], MEC servers can be directly colocated with the Base Stations, or deployed at aggregation points and/or at the edge of the core network. MEC servers, in addition, limit network congestion by processing data directly at the edge, instead of forwarding a big amount of data to the cloud. This particularly applies to MEC servers co-located with 5G Base Stations, which can provide computational capabilities as close as possible to end-users and capture information for further purposes (e.g., data analytics and big data processing). The servers are monitored, configured, and orchestrated by the Multi-access Edge Orchestrator, which represents a fundamental entity of the ETSI-MEC

reference architecture, included in the MEC system-level management [40]. To this end, SDN controllers continuously interact with the orchestrator (and with the rest of the network) for monitoring several parameters, e.g., the computational resources the users' request, the number of resources exposed and/or available in each MEC server.

1.3.3 Network Slicing

5G mobile networks promise to jointly support a large variety of applications that present different QoS requirements, traffic patterns, and radio resource usage. In this context of effective service differentiation, Network slicing emerged as an effective design paradigm, fostered by SDN and NFV. According to 3GPP specifications [41], a slice instance represents a set of network functions and related resources that are arranged and configured in a logical network to meet certain network characteristics. As a consequence, Network Slicing allows the creation of network segments, dedicated to the provisioning of specific services with their own Service Level Agreement (SLA) and QoS requirements, while enabling data and control plane functionalities to be programmable and auto-configurable [42]. The design of each slice is service-based, as it is steered by the requirements of a particular service [43]. In other words, the slice appears as a virtualized and independent portion of the overall network, configurable through a service-based approach.

Moreover, Network Slicing is emerging as a valid key enabler to support customized network services on-demand, permitting multiple vertical industries to execute their solutions on the top of shared infrastructure and accommodating heterogeneous services [44]–[47]. Typically, the slice Tenants (TNTs), i.e., the customers from vertical industries, have a vision of the underlying infrastructure as a virtualized entity of which they have, at least partially, control and which they can configure and operate independently [42]. While the Infrastructure Provider (IP) still represents the owner of the resources employed for each slice, a slice TNT is allowed to use those resources, install its own applications, hold its own data, and enable its preferred security policies. To this end, a TNT declares some communication service requirements to the IP. In turn, the IP configures the corresponding network slice instance, whose preparation phase includes the on-boarding and verification of network function products and the necessary network environment. From this moment on, the TNT can dynamically allocate the resources belonging to the aforementioned slice to the served mobile users (i.e.,

the task offloading within a specific slice). Moreover, TNTs should have the ability to adapt their slice requests to their users' requirements in real-time, dodging additional expenses due to the problem of resource overbuying. Thus, the slice request generation, i.e., when each TNT declares its desired slice configuration to the IP, clearly becomes crucial [48]. Note that in complex deployments, where heterogeneous services are offered through different slices, the proposed approach can be replicated for each slice. Indeed, the most common slicing scenario includes a single IP and several independent TNTs that provides advanced network services [49].

Overall, Network Slicing promises to open new business models for all the interested stakeholders, while intensifying the collaboration among all the involved parties and keeping their requirements distinct [50], [51]. On the one hand, in fact, the IP should manage and accept resource requests issued by TNTs, without having access to their most significant data. On the other hand, TNTs should be able to submit their requests, without having complete comprehension of the network itself.

1.3.4 Network Intelligence

As it should be clear by now, 5G mobile networks are definitely characterized by a large increase in the heterogeneity of the supported services and explosive growth of communication traffic. This expanding complexity made the management and the monitoring of the multitude of network elements almost intractable [52], [53]. For this reason, Machine Learning (ML) has been recognized as essential for solving complex problems in current and future generations of mobile systems [54], [55]. In particular, ML is the branch of AI that investigates algorithms able to learn and improve their experience and performance over time directly from data examples, without being explicitly programmed. Thanks to ML, a system can scrutinize data and deduce knowledge. In other words, hidden patterns in the training data are identified and used to analyze unknown information and drive the execution of a given task. Typically, these tasks include classification, prediction, and/or clustering [56]. ML is conventionally divided into three categories, based on both the type of available data and the problem goals. *Supervised learning* is a machine learning task that aims at learning to build a statistical model for predicting or estimating an output based on one or more inputs by using labeled data. *Unsupervised learning* aims to learn a function to describe a hidden structure from unlabeled data or an undefined and unspecific output

Reinforcement Learning (RL), i.e., the agent aims to optimize a long term objective by interacting with the environment based on a trial and error process and learning from past experience.

Deep Learning (DL) further improve ML capabilities employing neural networks, that mimic biological nervous systems (hence their name). In addition, the combination of DL and RL techniques produce *Deep Reinforcement Learning (DRL)* algorithms: the agents exploit neural networks to obtain the optimal policy [54], [57].

Overall, the motivations to adopt ML techniques in mobile networks are numerous:

- developing low-complexity algorithms for resource allocation;
- overcoming the lack of network information/knowledge;
- facilitating self-organization capabilities to reduce CAPEX and OPEX
- reducing signaling overhead;
- learning robust patterns and avoiding unsatisfying heuristics.

Furthermore, ML may be applied to several themes to improve traditional performance, including, but not limited to, Networking (Routing, Switching, Clustering), mobility management, Localization, power control, beamforming, management of spectrum, backhaul, cache, and computation resources [55].

1.4 Research Directions

5G mobile networks jointly support a large variety of applications, while supporting various new use cases from vertical industries. This imposes a wide range of performance and requirements and requires the network a tremendously high level of flexibility and scalability. Accordingly, the number of research, standardization, and deployment challenges is massive [18], [21], [34], [43], [46], [58]–[73]. To make matter worse, the integration of different components in the 5G air interface, i.e. NR, makes the overall system configuration a very challenging goal to accomplish. This thesis will essentially explore three different aspects: system-level simulations, NB-IoT, and RAN slicing.

System-level simulation always supported both the design and the evaluation, as well as the optimization of emerging technologies, while guaranteeing faster and cheaper investigations than real-world prototypes. At the time of this writing, there exist indeed many interesting simulation tools modeling the 5G air interface [58], [74]–[82]. However, they only implement specific subsets of technical components. Some of them also come with a commercial license, which generally restricts their adoption in many research teams. For these reasons, Chapter 2 presents 5G-air-simulator, an open-source tool offering a valid scientific and technological support for the development and the diffusion of modern service platforms built on top of 5G communication infrastructures.

In parallel, new solutions are also needed to provide appropriate support for MTC and the growing IoT in general. Here, the focus is on coverage, computational complexity, as well as energy constraints, and cellular-based solutions, e.g., NB-IoT, emerged to offer superior performance and easier management. Even in this case, the need for suitable simulation tools increases as well, as both academia and industry are increasingly involved in the development of NB-IoT. Still, only preliminary NB-IoT implementations have been proposed, which are largely incomplete or not freely available for the research community [83], [84]. Starting from this premises, Chapter 3 presents the promising NB-IoT radio interface with a particular focus on the challenges related to the Random Access Procedure, as well as its implementation in 5G-air-simulator.

Finally, the idea to support orthogonal logical segments also at the radio interface of 5G and Beyond 5G (B5G) deployments recently gained momentum. Differently from the conventional network slicing concept, RAN slicing is less mature and more challenging, because of the intrinsically shared and unpredictable nature of wireless resources. Indeed, the integration of the Network Slicing paradigm in the RAN is a complex task, which requires the definition of novel Radio Resource Management (RRM) functionalities. e.g., spectrum planning, interference coordination, packet scheduling, and admission control [69], [85]. To bridge this gap, Chapter 4 proposes a novel architecture to realize TNT-driven RAN slicing for Latency Sensitive Services and presents the concept of applying RAN slicing also to Wireless Local-Area Networks (WLANs) in order support indoor healthcare monitoring.

Chapter 2

An Open-Source Platform Exploring the 5G Air Interface

5G-air-simulator is an open-source and event-driven tool modeling the key elements of the 5G air interface from a system-level perspective. The proposed software has been designed with flexibility and extensibility at its core, therefore, it can be adopted to effectively pursue, with a limited additional effort, new research questions arising from new applications/services and features. Moreover, the simulator already proved to be a valuable tool for different research activities and scientific contributions. This tool aims at guaranteeing long-lasting exploitation of scientific results and offering valid scientific and technological support for the development and the diffusion of modern service platforms built on top of 5G communication infrastructures.

The rest of this Chapter is organized as follows: Section 2.1 presents a comparison of the 5G system-level simulation tools. The structure and the general-purpose features of the new 5G-air-simulator are described from Section 2.2 to Section 2.5. The remaining sections explore more advanced features specifically developed for some challenging 5G scenarios. In particular, Section 2.6 describes how the Massive MIMO technology can be used to provide high-bandwidth internet connection in a variety of environments, from rural to urban areas. Section 2.7 shows how extended multicast and broadcast techniques can be used to realize video streaming for a large number of users in a highly bandwidth-efficient way. In Section 2.8, the problem of degraded performance on very high-speed trains is presented, and the predictor antenna concept is used to improve the issue to a large extent. Finally, Section 2.9 explains the implementation of an enhanced random access procedure, which is important to improve the performance of massive IoT deployments on cellular networks. Please note that Thesis' Conclusions will outline the future of the simulator and suggest possible research fields that could take advantage of it.

2.1 State of the Art on 5G System-Level Simulators

This Chapter is oriented toward a specific type of simulation, i.e., system-level simulation modeling complete networks with multiple base stations and a large number of mobile users for the evaluation of procedures related to mobility, application, physical transmissions, scheduling, frequency reuse, and so on. To limit complexity to an acceptable level, system-level simulators employ various simplifications. The opposite category is that of link-level simulation, where the models go into great levels of detail, but simulation is usually limited to a single link, hence the name. Link-level simulations are used to investigate topics that can be limited to a single communication link, and the results can be used to construct simpler and faster models to realize system-level tools.

At the time of this writing, some simulators are known to be available or in development for the 5G. A comparison of the main qualities available in them is shown in Table 2.1, separated into features and general information.

TABLE 2.1: Comparison of the features of various 5G system-level simulators.

Feature	[58]	[74]	[75]	[76]	[77]	[78]	Vienna [79]	5G K-SymSys [80]	NetSim [81]	WiSE [82]	**5G-air-simulator**
Massive MIMO			✓		✓			✓		✓	✓
Multicast/Broadcast											✓
Predictor Antennas											✓
Random Access Procedure						✓		✓			✓
NB-IoT			✓								✓
MIMO				✓	✓	(✓)	✓			✓	✓
HetNet			✓		✓	✓	✓			✓	✓
Calibrated channel models	✓					(✓)		✓		✓	✓
Flexible Numerology					✓	✓	✓	✓	✓	✓	(✓)
mmWaves					✓		✓	✓	✓		
Dual/Multi connectivity				✓	✓	(✓)			✓		
Device-to-Device (D2D)							✓				
Integrated system and link-level		✓					✓				
Integrated network simulator					✓	✓		✓			
General information											
Programming language	N	N	N	U	C++	C++	Matlab	C++	C	C++	C++
License	N	N	N	U	GNU GPL	GNU GPL	Acad.	Acad.	Comm.	Comm.	GNU GPL
Open-source	N	N	N	U	✓	✓	✓	✓			✓
Usable implementation available				✓	✓	✓	✓	✓	✓	✓	✓

Note: * N = N/A, U = Unknown, (✓) = partial implementation of the reference feature.

It is important to underline that some of the missing features of the 5G-air-simulator are either part of the future work or out-of-scope with respect to the main goals of the tool presented herein.

The work in [58] describes a two-level simulator, including the core network and the access network as well as their interactions, which also takes advantage of cloud resources to speed up the simulation. However, this architecture is only a high-level proposal and there is no actual implementation yet. Similarly, the authors of [74], [75] outline an architecture for 5G simulators, but the simulators themselves are still not complete and only some features are presented in the papers. The simulator presented in [76] seems to be an actual product with various features but is not stated whether it is open source or otherwise available to the public. Moreover, the only relevant feature that falls into the 5G realm is the aggregation of cellular and Wi-Fi traffic, i.e. dual connectivity.

The work [77] presents an open-source simulation tool for LTE-like 5G mmWave cellular networks integrated into the widely used open-source ns–3 simulator [86]. Starting from both this module and the LTE module (LENA) [87], the authors of 5G-LENA [78] conducts a comprehensive and intensive work to align both modules to the latest standard published by 3GPP and build a NR simulator. However, despite its compliance with the latest standards, the number of offered features is rather limited: it lacks spatial user multiplexing, MIMO and Massive MIMO, and FDD.

On the other hand, Vienna 5G system-level simulator [79], which is a direct evolution of the pre-existing LTE-Advanced (LTE-A) system-level simulator, adds many 5G-related capabilities such as new propagation models, heterogeneous networks, D2D operation, relays, and IoT scenarios. Also, the 5G K-SimSys simulator [80], which is part of the 5G K-Simulator platform [88] integrating link, system and network-level simulators, offers several 5G capabilities. While this feature set is remarkable compared to other available simulators, the comparison with 5G-air-simulator is not this immediate. In fact, the simulators share some common features, whereas some functionalities are only available in the Vienna and 5G K-SimSys simulators (e.g. D2D, mmWaves) and some are exclusive to 5G-air-simulator (e.g. Massive MIMO, broadcasting). As for licensing, the Vienna and 5G K-SimSys simulators are freely available for academic purposes, but require the purchase of a license for commercial use, while 5G-air-simulator is under GPL license, thus it is free to use for everyone and the source is always available.

It is worth noting that [77], [78], [80] are also network simulators, hence

allowing not only the analysis of the application and the radio interface layers of but also of E2E scenarios with a full protocol stack.

Other simulators can be purchased using a commercial license, therefore their source codes are not publicly available. Specifically, there is the NetSim's 5G library for mmWave networks [81] and the WiSE simulator [82].

In conclusion, 5G-air-simulator offers many technical components enabling the 5G air interface and calibrated channel models. Moreover, since it is under GPL license, it allows for a simple and fast utilization, as well as the possibility to investigate new protocols and technologies by extending the available code.

2.2 The Core of 5G-air-simulator

5G-air-simulator is written in the C++ language with an object-oriented paradigm and extends the popular LTE-sim network tool [89]. The source code is readily available at [90]. To help the reader understand its inner workings and how some features and properties are achieved, this Section encompasses some general properties of the 5G-air-simulator. Figure 2.1 depicts the main building blocks of the developed tool.

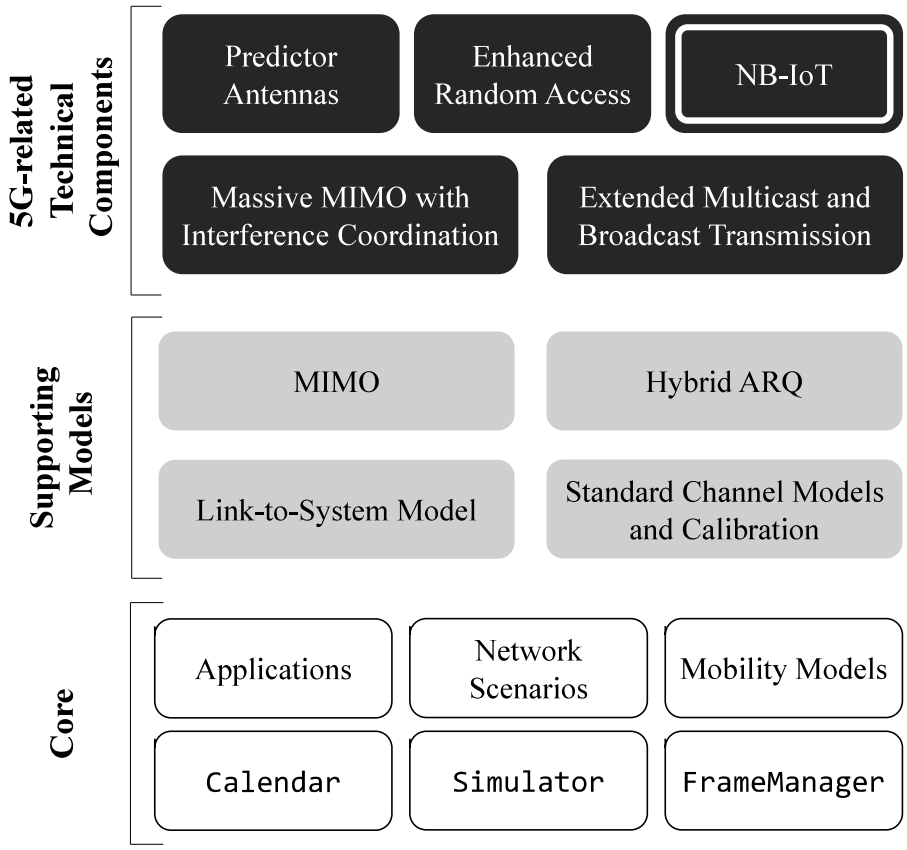


FIGURE 2.1: Building blocks of the 5G-air-simulator. NB-IoT is an independent component built directly on the Core and it will be described in Chapter 3.

Simulator's core represents the first key building block, providing all the procedures useful to implement object-oriented and event-driven paradigms, as well as to manage nodes, protocol stack, mobility, and applications. It mainly inherits from the well-known LTE-Sim tool [89]. On top of the simulator's core, the 5G-air-simulator integrates many other supporting models that significantly extend the features initially offered by LTE-Sim. They include the calibrated and standard compliant Link-To-System (L2S) model, MIMO features, and HARQ. These latter models offer a suitable substrate for the development of 5G technical components, like Massive MIMO, extended multicast/broadcast transmissions, predictor antennas, enhanced random access procedure, and NB-IoT. Although reference scenarios have been developed to conduct a performance assessment of each 5G technical component almost independently, it is worth mentioning that a simulator's user can realize completely new scenarios (as explained in Section 2.5) by

combining multiple components. As a matter of fact, all the presented building blocks may be used concurrently depending on the users' needs. At the time of this writing, however, the NB-IoT component only works standalone. This aspect will be tackled in the future when it will be clearer the inclusion of NB-IoT in 5G NR specifications.

The simulator's core and its supporting models are presented in this Section, while the implemented 5G technical components will be deeply discussed later on.

5G-air-simulator is structured as an event-driven application: the `Calendar` class holds a list of events to be executed, with each item containing the required time of execution, the method to execute, the object on which the method should be called, and possibly some parameters. Other important classes are `Simulator` and `Framemanager`. `Simulator` is a singleton class performing global actions, such as initiating and halting the simulation, adding events to the `Calendar`, and running them. The `FrameManager` tracks the flow of time, increases the counters related to frames and sub-frames, and, in some cases, it marks sub-frames dedicated to different functions, e.g., downlink versus uplink sub-frames in TDD mode [91]. Practically, it is in charge of scheduling the events related to the start and the end of frames and sub-frames according to a fixed frame structure.

At the beginning of the program's flow, one of the available scenarios is selected. The scenario is in charge of creating and initializing many important elements of the simulation environment, such as the base stations, mobile terminals, and channel realizations. Most scenarios also accept several parameters that affect the specific details of the objects that are created (e.g. the scheduling algorithm at the base stations [92] or the periodicity channel state reporting) or even the simulation as a whole (e.g. the total duration or the size of the environment). In some cases, the initialization of some objects defines some events that are inserted into the `Calendar` class, to be executed at a later time. This includes, for instance, the generation of data packets at the application layer and the movement of the devices.

After all the initial setup, the actual simulation is started by calling the `Simulator::Start()` method. At this point, the `Calendar` class starts executing the registered events in chronological order. Each event can result in the generation of new events that are put in the calendar, resulting in a sustained supply of events to process until the end of the simulation. In particular, some kinds of events re-schedule themselves just at the end of their execution, thus repeating periodically for the entire simulation time.

These include the allocation of radio resources and the reception procedures of each device. The generation and execution of events go on until a call is made to the `Simulator::Stop()` method, which causes the calendar to stop and discard all the events that may still be pending, and finally terminate the program. Usually, the end time of the simulation is set in advance via a call to `Simulator::SetStop()` in the scenario, right before calling `Simulator::Start()`.

2.2.1 Application Layer and the Protocol Stack

In 5G-air-simulator, application models are in charge of generating data packets that are then forwarded through the protocol stack, transmitted, and then processed at the receiver's protocol stack. Different models are available to cover varying situations. They are all derived from the same `Application` class, thus introducing new models is as simple as writing a new class derived from it.

A straightforward model is the `InfiniteBuffer`. A transmitting node using this model acts as an infinite supply of data: at every occasion for communication, it generates as much data as can be transmitted, for the entire simulation. This model is intended to put as much stress as possible on the network and measure its maximum capacity.

The `TraceBased` model is intended to emulate the traffic generated from video streaming. The model contains many traces created from an actual video file [93], containing the size and timestamp of each frame. These are then used to generate data packets at the appropriate times. If the end of the trace is reached, it is restarted from the beginning.

For voice traffic, there is a `VoIP` model: it follows the G.729 model [94], thus generating packets of constant rate and size, but only during the so-called active state. Instead, in the inactive state, no packets are created. At any given time, there is a given probability of going from active to inactive state or vice-versa, thus reflecting the intermittent nature of human speech.

Web browsing is described with the `FTP2` model [19], where packets of a given size (representing a web page) are generated at random intervals (representing the inactive time where the user is reading). The average duration of the interval is modeled with an exponential probability, where the mean value is given as a parameter, together with the packet size.

Finally, the simplest model is constant bit-rate or `CBR`. It generates packets of a fixed size at fixed regular intervals, which are both input parameters.

This model is intended to represent applications that transmit data with a strict recurring schedule, such as remote sensors producing periodic reports.

5G-air-simulator also supports several other features of both user- and control-plane protocol stacks. To this aim, each device implements an instance of the `ProtocolStack` class, which in turn contains Media Access Control (MAC), Radio Link Control (RLC), Packet Data Convergence Protocol (PDCP), and Application entities. The Application entity associates the sources with the destinations of each application flow, which is generated as mentioned above. The PDCP Entity mainly handles the header compression of all the packets coming from the upper layer and enqueueing into the MAC entity. The RLC Entity models the three data transfer modes, namely Transparent Mode (TM), Uncknowledged Mode (UM), and Acknowledged Mode (AM), and handles the buffering, segmentation/reassembly, and retransmission of service data units. Each dedicated radio bearer has its own RLC entity. The MAC Entity provides, for both users and gNBs, an interface between the device and the PHY layer, for delivering packets coming from the upper layers to the PHY one and vice versa. Furthermore, the Adaptive Modulation and Coding (AMC) module and the Packet Schedulers also belong to the gNB's MAC entity.

At the bottom of the protocol stack, the `Phy` class allows to customize physical characteristics, by setting the transmission power, the number of transmitting and receiving antennas, and the noise figure for both users and base stations. Furthermore, for base stations only, it is also possible to choose between omnidirectional or tri-sector antennas. In the latter case, several parameters may be customized, e.g., the antenna's bearing and its gain, e-tilt, horizontal and vertical beamwidth at 3 dB, feeder loss, and maximum horizontal and vertical attenuation.

2.2.2 Network Deployments and Mobility Models

5G-air-simulator includes a number of basic scenarios reflecting typical conditions used in research and testing, ranging from a single cell to heterogeneous network configurations, as depicted in Figure 2.2. The simplest one is called `SingleCell`: as the name suggests, it only contains one cell and a single omnidirectional base station, with a configurable number of users in it. Also, since there are no other cells around, there is no inter-cell interference. This architecture may be used, for instance, to evaluate peak throughput or coverage under ideal conditions or for isolated sites.

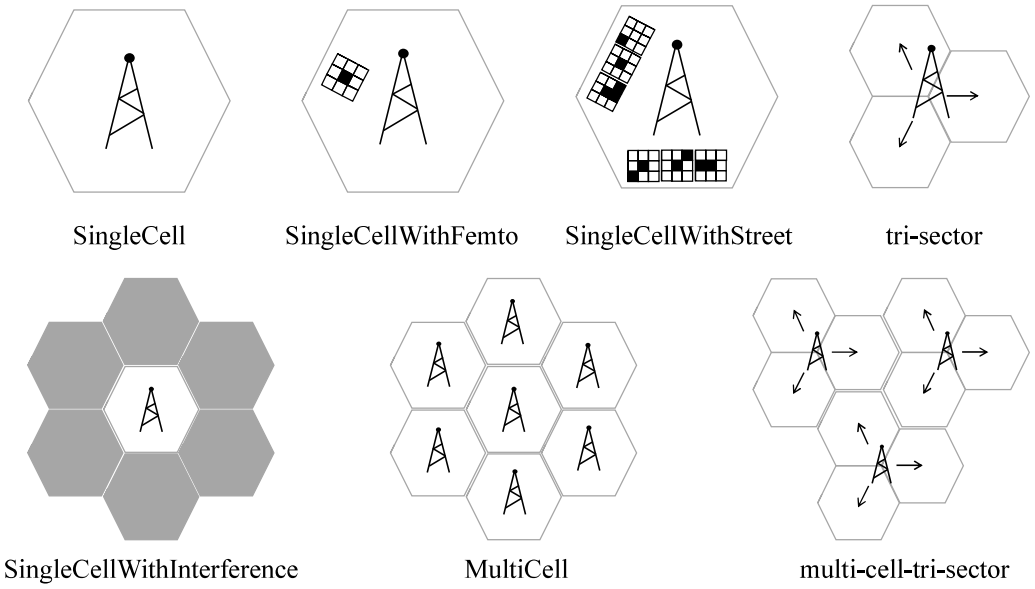


FIGURE 2.2: Network deployments available in 5G-air-simulator.

A more realistic configuration is constructed in the `SingleCellWithInterference` scenario. There is still a single omnidirectional base station with active users, but this time it is surrounded by other base stations that do not serve any user. Nonetheless, they still create inter-cell interference, modeled as an always-on transmission at full power in all the radio resources. Therefore, these surrounding interfering nodes influence the performance metrics measured in the primary cell.

For more in-depth evaluations, the `MultiCell` scenario creates a full multi-cell environment with multiple active base stations, each with multiple users. With this setup, it is possible to evaluate issues such as frequency reuse and handover, although the simulation time may increase substantially.

Instead of an omnidirectional base station, for some scenarios there are also versions with three-sector cells: `tri-sector` is similar to `SingleCell`, except that there are three co-located base stations and each one serves a 120wide sector. While there is still no interference from other base station sites, there is still interference among sectors, so it is not as idealized as the `SingleCell` case. Similarly, `multi-cell-tri-sector` is similar to `MultiCell` but with each cell replaced by three 120sectors. This configuration is the most similar to real-world deployments among those presented here.

Finally, two additional scenarios involve heterogeneous networks [95]. `SingleCellWithFemto` involves a single macro-cell together with many femtocells, deployed into houses belonging to a group of buildings. Some of the

users are placed into the coverage area of the cell, while others are created in clusters near the femtocells. Similarly, the `SingleCellWithStreet` scenario constructs a single macro-cell and a configurable number of streets, where each street includes a block of buildings and the corresponding femtocells. These scenarios are intended to investigate issues related to such heterogeneous configurations, such as access policies, handover, and scheduling [96].

5G-air-simulator includes different mobility models [97], and others may be easily implemented. `ConstantPosition` is the most simplistic one, as the users do not move. In `RandomDirection`, users move with a constant speed towards a given direction, that is randomly selected at the beginning of the simulation. When the limit of the simulation area is reached, a new direction is selected (pointing back towards the simulation area) and it is followed until the end of the area is reached again, then the process is repeated. Similarly, the `LinearMovement` model allows selecting a given direction. On the other hand, when using the `RandomWalk` model, users still move toward a random direction, but they do not need to reach the end of the simulation area. Instead, the direction is changed after traveling a certain distance. The last available mobility model is called `Manhattan`: in this case, the user can only move in a horizontal or vertical direction, and at any given time it has a certain probability to turn left or right. This is intended to represent city environments where all the streets are at straight angles.

As reported previously, there is the possibility to implement even more mobility models. For instance, it is possible to implement the well-known Random Waypoint as an extension of the baseline `RandomWayPoint` class, which is already available in the code. However, it is necessary to highlight that, at the time of this writing, mobility models do not interact with the possible presence of buildings in scenarios.

Finally, although users' arrivals and departures are not modeled, i.e., the number of users is fixed throughout an entire simulation, it is possible to model the arrivals, as well as the departures, of each traffic flow associated with the users, hence somehow achieving the same result.

2.2.3 Link Adaptation

The purpose of the link adaptation is to identify the Modulation and Coding Scheme (MCS) that is more appropriate for the channel quality perceived by each user. In fact, a modulation level that is too high could theoretically transmit more data, but it would also result in many more errors at the physical

layer, thus nullifying the advantage. Instead, a modulation level that is too low would result in lower speed without any significant gain. To this aim, it is of the utmost importance to find the optimal MCS. Basically, the users compute the Channel Quality Indicators (CQIs), i.e., quantized Signal to Interference plus Noise Ratio (SINR) values obtained through the estimation of the channel quality. Then, they feedback the CQIs to the base station (reporting procedure), which is in charge of mapping them to MCS indexes. An MCS index is used together with the number of RBs (as well as the number of spatial layers) to find the net amount of payload bits that can be transmitted to the user, as seen at the MAC layer, namely TBS, following a standardized procedure [98]. The `AMCModule` class is in charge of conducting this entire procedure during the resource allocation.

2.3 Supporting Models of 5G-air-simulator

This Section will provide an overview of the main models developed for supporting 5G-related Technical Components, which will be discussed in later sections. In particular, supporting models of 5G-air-simulator include a calibrated Link-To-System (L2S) model, a MIMO module, and a system-level implementation of the HARQ procedure.

2.3.1 Calibrated Link-to-System Model

The L2S model has the main purpose of quantifying the effectiveness of radio transmission, taking into account many phenomena, such as propagation and interference [99]. In a system-level simulator, this task should be accomplished without requiring explicit modeling of all the involved details, which would result in excessive complexity and very long running times. Instead, link-to-system models provide a simplified description of the phenomena of interest, which is still sufficiently accurate for the purpose of simulating a large system. Figure 2.3 depicts the main blocks of the link-to-system model designed for the 5G-air-simulator.

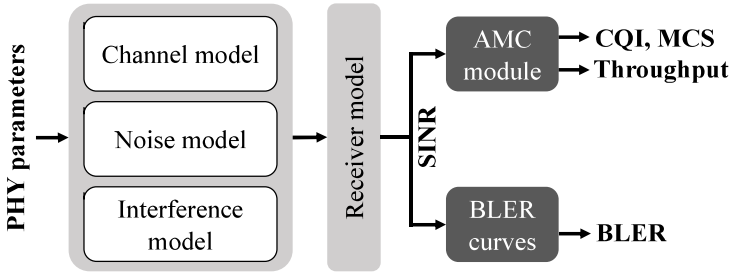
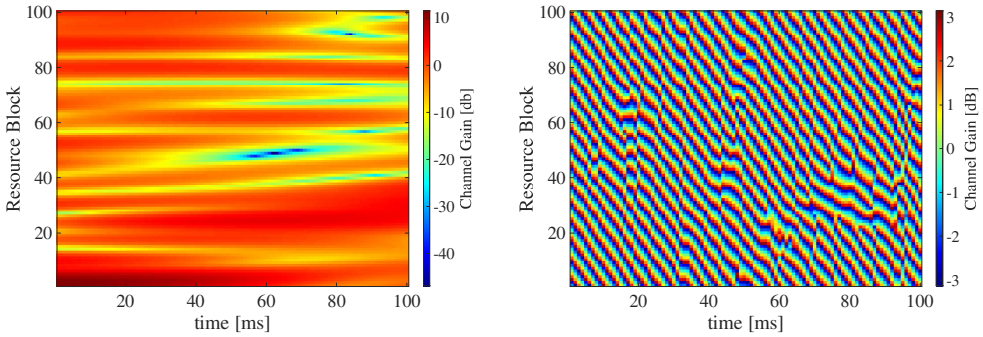


FIGURE 2.3: Components of the implemented link-to-system model.

The preliminary assumption in 5G-air-simulator's L2S model is that each transmitted signal is represented by its power spectral density value. Then, channel, noise, and interference models describe how this transmitted signal is seen at the receiver [100]:

- Channel models calculate the attenuation of the signals due to propagation, and they are composed of different parts: path loss, shadowing, penetration loss, and fast fading [101].
  Fast fading models describe the small-scale parameters, such as delays, powers, and directions of arrival and departure on a very short time scale, about the size of a TTI. These frequency selective channel variations, which are mainly due to multipath, are modeled with precomputed traces generated according to tabulated distribution functions and random parameters, as described in [102]–[104]. The stored traces include the effect of time, frequency, and antenna correlation when MIMO transmission is used, as well as multiple interactions with the scattering media. A sample of the channel gains in the time-frequency domain for different user speeds is shown in Figure 2.4.
  Shadowing is modeled as a log-normal variable, and penetration loss is usually set to a constant value for indoor users, but it may also be a random quantity too.
  The path loss depends mainly on the distance, the frequency, and the environment [105]. The models included in 5G-air-simulator are reported in Table 2.2 and Table 2.3, where the meaning of the most commonly used symbols is as follows: d is the distance between the base station and the user in km, d_{3D} is the 3D distance (including heights in the computation), f is the center frequency in GHz, H_{gnb} is the height of the base station, H_b is the average height of the buildings around it, and H_{ue} is the height of the mobile user.



(A) 3 km/h (B) 120 km/h

FIGURE 2.4: Fast fading realizations at different user speeds.

It is worth mentioning that other minor phenomena usually affecting radio channels, e.g., blockage, are not modeled in order to maintain the complexity of the L2S model at an acceptable level.

- The noise power is calculated as simple thermal noise, with a spectral density of -174 dBm/Hz that is integrated over the bandwidth of one RBs.

- The interference power is modeled as the sum of the contribution from all the base stations (except for the serving base station), again using all the channel models as described above.

After the reception, the receiver has an SINR value for each RBs n, i.e. $\gamma(n)$. For more details, see subsection 2.3.2, which thoroughly describes how the SINR is computed based on the selected transmission settings.

These SINRs need to be mapped to a single effective SINR value, reflecting the overall quality of the radio channel. Although different algorithms exist for this calculation, 5G-air-simulator utilizes the Mutual Information Effective SINR Mapping (MIESM) method [110], which is known to provide good results with minimal tuning. Let $\mathcal{N}$, N, $I(\cdot)$, and β be the set of RBs assigned to the user, the cardinality of $\mathcal{N}$, the mutual information function, and a parameter that can be adjusted to match specific combination of modulation schemes, respectively. Then, the effective SINR, namely $\bar{\gamma}$, is computed as:

$$\bar{\gamma} = \beta I^{-1}\left(\frac{1}{N}\sum_{n\in\mathcal{N}} I\frac{\gamma(n)}{\beta}\right). \tag{2.1}$$

Note that $\bar{\gamma}$ is used for two different purposes. First, it is exploited to estimate the BLER for the received data block using the SINR-BLER curves in Figure 2.5, which determines the probability that it has been received and

TABLE 2.2: Baseline path loss models available in 5G-air-simulator.

Name	Formula (dB)	Notes
Urban Macro-cell [106]	$80 + 40(1 - 40.001(H_{gnb} - H_b))log10(0.001d) - 18\log_{10}(H_{gnb} - H_b) + 21\log_{10}(f)$	
Suburban Macro-cell [106]	$128.1 + (37.6\log_{10}(0.001d))$	
Rural Macro-cell [106]	$69.55 + 26.16\log_{10}(f) - 13.82\log_{10}(H_{gnb}) + (44.9 - 6.55\log_{10}(H_{gnb}))\log_{10}(0.001d) - 4.78\log_{10}(f)^2 + 18.33\log_{10}(f) - 40.94$	
Urban Micro-cell [106]	$24 + (45\log_{10}(d))$	
Urban Femto-cell [107]	$max(45, 127 + (30\log_{10}(0.001d)) + 18.3n^{((n+2)/(n+1)-0.46)})$	
Winner downlink [108]	$A\log_{10}(d) + B + C\log_{10}(2.0/5.0) + 10.0nbWalls[1] + 20.0nbWalls[0]$	$LOS: A = 18.7$, $B = 46.8, C = 20.0$ $NLOS: A = 20.0$, $B = 46.4$, $C = 20.0$ $nbWalls[0]$ is the number of external walls, $nbWalls[1]$ is the number of internal walls
Basic downlink	$37 + (30log10(d))$	

must be discarded. Second, it is sent to the base station to inform it of the perceived channel quality through CQI feedbacks, so that it can properly perform the link adaptation procedure (as explained in subsection 2.2.3).

It is worth mentioning that although popular SDR platforms [111], [112] and link-level simulators may be used to deeply investigate topics limited to a reduced number of communication links, the obtained results can be leveraged to build additional supporting models to be integrated into 5G-air-simulator, e.g., new modulation and coding schemes, BLER/BER curves.

An important property that simulation tools can have is the calibration of the channel models. Having calibrated channel models means that the outcomes of specific simulations have been compared to those of other similar products considering the same scenario, and channel models and settings have been adjusted until the results are similar. The calibration is important to ensure that there are no major errors in the implementation and that results from different solutions can be compared without incurring into unwanted misalignments. The simulation assumptions and the reference data are available in [113] and [114], while the most relevant parameters are summarized in Table 2.4. 3D channel models [103], [104] are embraced in the calibration

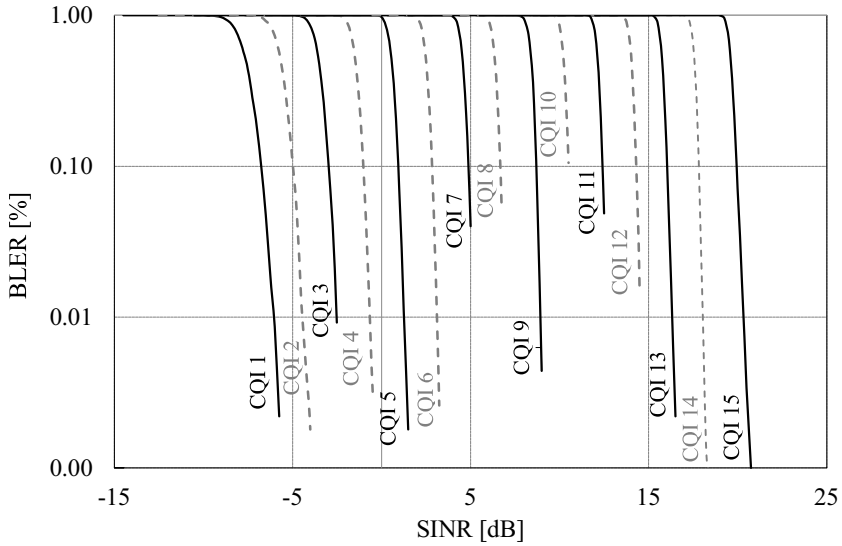


FIGURE 2.5: BLER curves for the link-to-system model obtained by MATLAB link-level Toolbox.

process, taking into account time, space, and frequency correlation. As an example, Figure 2.6 and Figure 2.7 show the calibration results for the path gain and the SINR in the reference urban scenario, confirming that 5G-air-simulator is well-calibrated.

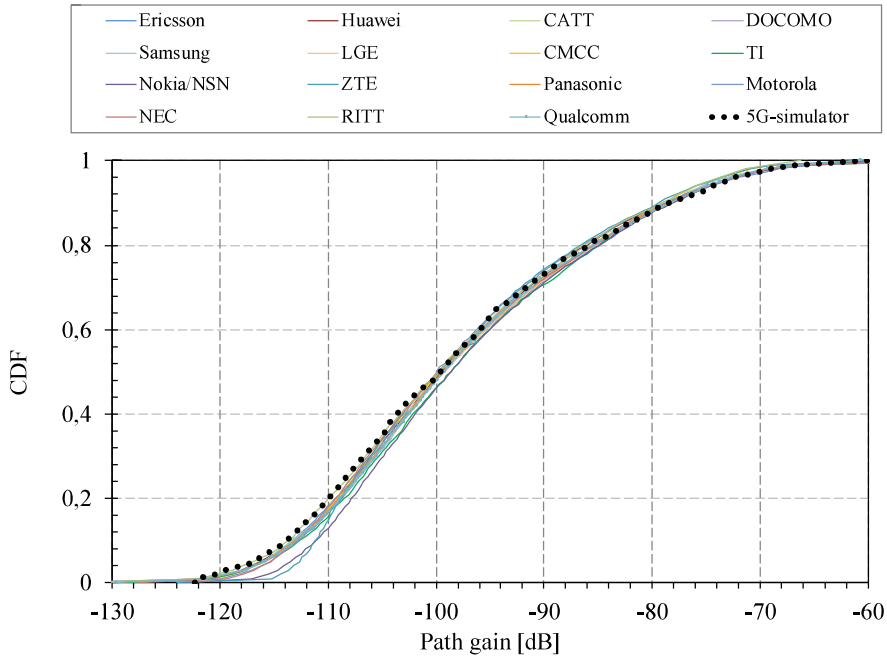


FIGURE 2.6: Calibration of path gain (urban scenario).

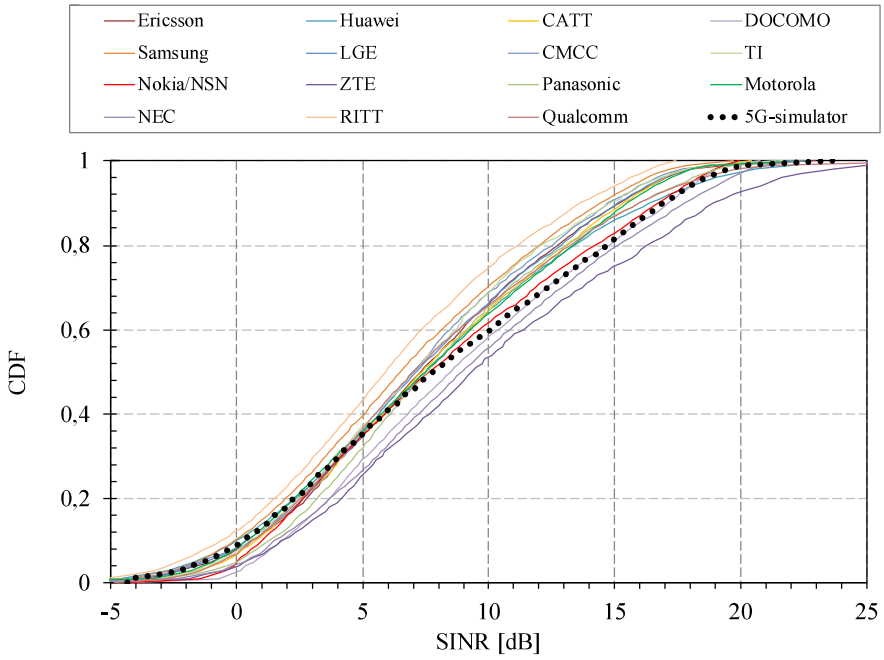


FIGURE 2.7: Calibration of SINR (urban scenario).

Similar results can be obtained for the other scenarios, like suburban or rural.

2.3.2 MIMO

In general, a MIMO system can be modeled according to Figure 2.8.

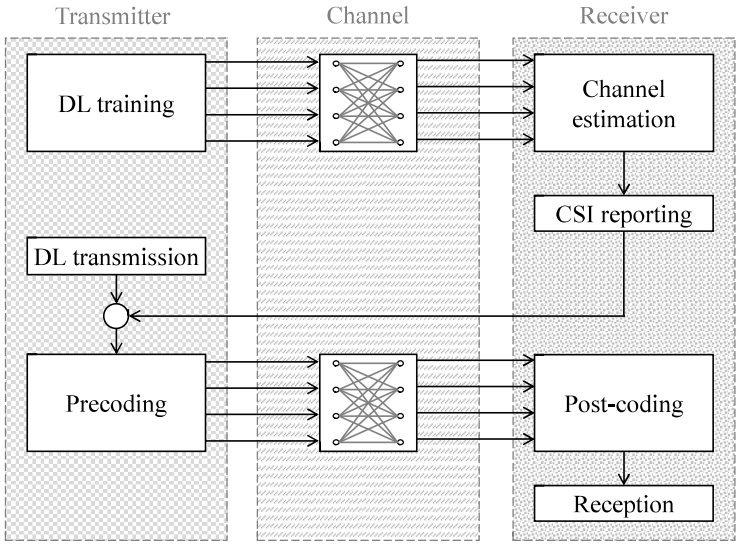


FIGURE 2.8: Block diagram of a MIMO system.

During the precoding step, the user data has to be mapped to the available antennas in an appropriate way, since the radio channel is multi-dimensional due to multiple transmit and receive antennas. The goal of precoding is to make good use of the spatial degrees of freedom. In particular, the latter is better achieved when some information about the channel is available at the transmitter, usually thanks to the Channel State Information (CSI) reporting procedure. Such information can be acquired via a channel estimation procedure, however, that is not done in all cases. Contrarily, channel information is always required at the receiver. It is acquired at the same time as the user data via training signals, and it is used in the post-coding phase to reconstruct the original data. Different MIMO schemes differ in how the precoding, post-coding, and channel estimation functions are designed. In LTE and LTE-A, the configuration of specific MIMO features is done with the so-called Transmission Modes (TMs): communication with any given user can be configured with one of the available TMs, which dictates whether MIMO is used and what specific technique is employed. The presented version of 5G-air-simulator supports the following TMs in downlink, derived from LTE specifications [115]

- TM1, single antenna transmission. The base station does not use any MIMO capability, however, if the mobile node is equipped with multiple antennas, it may still adopt Maximum Ratio Combining for improving the SINR. Let, $P_{tx}^{(j)}$ be the total transmit power from j-th base station (per cell), $P_{loss}^{(j)}$ the distance-dependent path loss, including shadowing and antenna gain/loss, $H^{(j)}(n)$ the channel gain from the j-th base station on the n-th sub-channel, N_r the number of receive antennas, $H_r^{(j)}(n)$ is the channel gain from the j-th base station to the r-th receive antenna on the n-th sub-channel, σ^2 the AWGN noise variance, and N_I the number of interferers. In the general case, the SINR of the n-th sub-channel is computed as:

$$\gamma(n) = \frac{P_{tx} P_{loss} \left|H(n)\right|^2}{\sigma^2 + \sum_{j=1}^{N_I} P_{tx}^{(j)} P_{loss}^{(j)} \left|H^{(j)}(n)\right|^2}, \tag{2.2}$$

while in case of Maximum Ratio Combining:

$$\gamma(n) = \frac{P_{tx} P_{loss} \left(\sum_{r=0}^{N_R-1} |H_r(n)|^2 \right)^2}{\left(\sum_{r=0}^{N_R-1} |H_r(n)|^2 \right) \sigma^2 + \sum_{j=1}^{N_I} P_{tx}^{(j)} P_{loss}^{(j)} \left| \sum_{r=0}^{N_R-1} H_r(n)^* H_r^{(j)}(n) \right|^2}. \tag{2.3}$$

- TM2, transmit diversity. Space-frequency coding is employed over 2 or 4 transmitting antennas, providing diversity and coding gain. The receiver may use one or multiple antennas. Given that $H_{t,r}^{(j)}(n)$ is the channel gain from the t-th transmit antenna of the j-th base station, to the r-th receive antenna, on the n-th sub-channel, the SINR is computed as:

$$\gamma(n) = \frac{P_S}{P_N + P_{I'} + P_{I''}}, \tag{2.4}$$

  where:

  - $P_s = P_{tx} P_{loss} \sigma^2 \left(\sum_{t=0}^{1} \sum_{r=0}^{N_R-1} |H_{t,r}(n)|^2 \right)^2$
  - $P_N = \left(\sum_{t=0}^{1} \sum_{r=0}^{N_R-1} |H_{t,r}(n)|^2 \right) \sigma^2$
  - $P_{I'} = \sum_{j \notin STBCset} P_{tx}^{(j)} P_{loss}^{(j)} \sigma_j^2 \cdot$
    $\cdot \left(\sum_{t=0}^{N_T^{(j)}-1} \left| \sum_{r=0}^{N_R-1} H_{0,r}(n)^* H_{t,r}^{(j)}(n) \right|^2 + \sum_{t=0}^{N_T^{(j)}-1} \sum_{r=0}^{N_R-1} H_{1,r}(n) H_{t,r}^{(j)}(n)^* \right)$
  - $P_{I''} = \sum_{j \in STBCset} P_{tx}^{(j)} P_{loss}^{(j)} \sigma_j^2 \, ( )$

- TM3, open-loop spatial multiplexing. This mode requires multiple antennas at both the transmitter and receiver, which are used to transmit multiple spatially multiplexed streams of data. The transmitter does not have fine-grained information about the MIMO channel, but only a Rank Indicator (RI). Hence, it uses a pre-defined and repeating sequence of precoding matrices, which is conceptually similar to periodically focusing the signal in different spatial directions. Receivers will get a good signal for some of the precoding matrices, and exploit the channel coding to recover the entire transmitted block from alternating good and bad observations.

- TM4, closed-loop spatial multiplexing. In this case, the mobile station provides the base station some information about the MIMO channel, in the form of Precoding Matrix Indicators (PMIs) and RIs. The PMI instructs the base station to use a specific precoding matrix from a shared codebook for the precoding, thus ensuring that the MIMO transmission is adapted to the instantaneous channel characteristics. This allows a higher throughput compared to open-loop MIMO but has the downside that much more information has to be transmitted on the reverse link, from the user to the base station. The maximum antenna configuration is 4x4, with a peak throughput up to 4 times greater than TM1.
- TM9, 8-layer MIMO. This mode was introduced in Release-10 and allows up to 8x8 MIMO with twice the throughput of TM4. PMI is still transmitted using an extended codebook, using twice as many bits as the TM3/TM4 codebooks, but in this case, the base station is not necessarily constrained to use one of the codebook's matrices for precoding.

For TM3, TM4, and TM9, the SINR is computed as:

$$\gamma_k(n) = \frac{\operatorname{diag}\left[\sigma^2 D(n) D^*(n)\right]_{kk}}{\operatorname{diag}\left[\sigma^2 W^*(n) W(n) + \sigma^2 I_{self} I^*_{self} + \sum_{j=1}^{N_I} P^{(j)}_{tx} P^{(j)}_{loss} \sigma^2_j W^*(n) H^{(j)}(n) H^{(j)}(n)^* W(n)\right]_{kk}} \tag{2.5}$$

where:

- $D(n) = \operatorname{diag}\left[W^*(n)\sqrt{P_{tx}} P_{loss} H(n)\right]$
- $I_{\text{self}}(n) = W^*(n)\sqrt{P_{tx}} P_{\text{loss}} H(n) - D(n)$
- $W(n) = \left(\sigma^2 P_{tx} P_{loss} H(n) H(n) + \tilde{\sigma}^2\right)^{-1} \sigma^2 \sqrt{P_{tx} P_{loss}} H(n)$
- $\tilde{\sigma}^2 = \sigma^2 I + \sum_{j=1}^{N_h} \sigma^2_j P^{(j)}_{tx} P^{(j)}_{loss} H^{(j)}(n) H^{(j)*}(n)$
- $H^{(j)}(n)$ is the channel gain matrix from the j-th base station.

Note that, without loss of generality, all the symbols without the index j are related to the target user/base station.

2.3.3 HARQ Procedure

HARQ is a retransmission method that works at the boundary between physical and MAC layers. It is a base feature of high-speed mobile networks, so it

has been implemented in 5G-air-simulator as well [116]. There are few variants of it, but the simulator currently supports chase combining (also called HARQ type I), which works in the following way:

- the transmitter sends a data block, encoded with a Forward Error Correction (FEC) code which allows error detection and correction to a certain extent.
- The receiver checks the received blocks for errors. If there are no errors, or the errors can be recovered, then a positive acknowledgment (ACK) is sent back and transmission is completed successfully. If there are non-recoverable errors, the received signal is held into a buffer and a negative acknowledgment (NACK) is sent.
- Upon receiving a NACK, the transmitter re-sends the same block again.
- The receiver gets a second copy of the message, which is combined via Maximum Ratio Combining (MRC) with the previous copy to increase the SINR. If the combined copy is now decodable an ACK is reported otherwise a NACK is returned.
- The transmission is repeated until the block can be decoded or the maximum number of attempts is reached.

Differently from plain Automatic Repeat reQuest (ARQ), which considers each retransmission separately, HARQ takes advantage of the combination of multiple copies of the signal. In the case of chase combining, the result is similar to the use of an additional repetition coding. Since the back-and-forth of HARQ may require some time, the receiver could remain stuck on a given block and unable to receive additional data. To avoid this, multiple HARQ sessions can be simultaneously active. These are called HARQ processes and up to 8 can be instantiated for each receiver, as the latency for single retransmission is typically 8 sub-frames. However, since HARQ is modeled from a system-level point of view, the number of processes, as well as their timers, may be set to custom values in order to speed up the entire procedure. In other words, the delay introduced by the HARQ procedure may be manually adjusted by defining a different ACK waiting time interval.

2.4 Simulation Tracing

5G-air-simulator provides a text trace during the execution of the simulation. Figure 2.9 shows an example of the text trace. The first field of each line re-

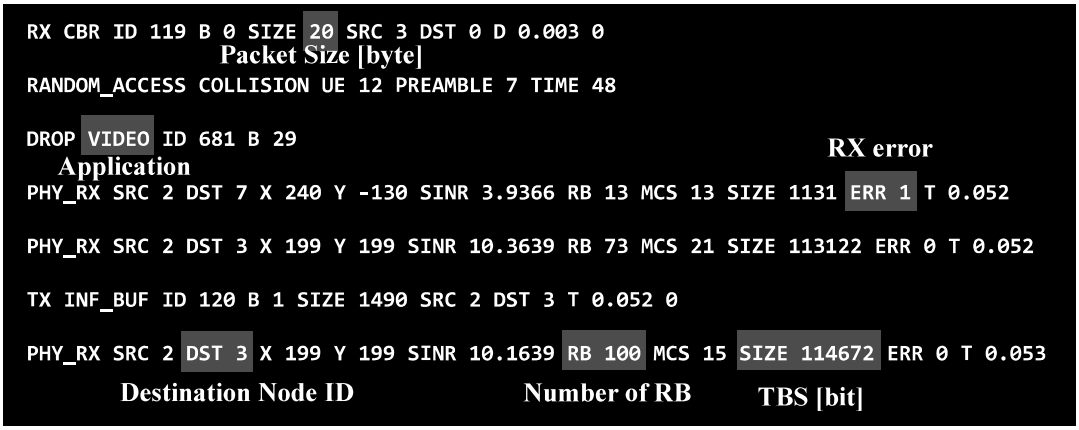


FIGURE 2.9: Example of the text trace of a simulation.

ports the event that triggered the tracing. Specifically, rows starting with TX, RX, and DROP are associated with packets that have been sent, received, and dropped, respectively. In addition, a line starting with PHY_RX indicates a reception event at the physical layer, while RANDOM_ACCESS provides information about the random access procedure. For lines related to packets, the second field describes the application type. SRC and DST identify the nodes that send and receive the packet, respectively, while ID identifies the packet uniquely, and B the bearer used to map the packet. The value after D during reception events represents the delay of the received packet in seconds. In general, T reports the time instant in which an event occurred, in seconds (TIME, instead, is in ms).

The main performance indicators can be retrieved by redirecting the console output of each simulation run to a different file and then extracting the relevant data. The main KPIs may be computed as follows.

- **Average User Throughput**: it is necessary to consider all the lines starting with the keyword `PHY_RX` and sum all the values appearing at the 17^{th} position (indicating the transport block size in bits), but only when the value at the 19^{th} position is 0 (indicating no reception errors). The resulting values should then be divided by the number of active users in the simulation.

- **Average Packet Loss Ratio (PLR)**: it is calculated as the ratio of lost packets over total transmitted packets at the application layer. In this case, the transmitted packets correspond to the number of lines starting with `TX`, while the received packets are identified by the number of lines starting with `RX`.

- **Average Random Access Collision Rate**: Similarly to the PLR, it is defined as the ratio between preamble collisions and successful completions. It is necessary to consider all the lines starting with the keyword `RANDOM_ACCESS COLLISION` to identify the collisions. On the other hand, the lines starting with `RANDOM_ACCESS RECEIVE_MSG4` indicate the procedure ending.

- **Cell Goodput**: consider the lines starting with the keyword `RX` and sum all the values appearing at the 8^{th} position, indicating the application data size in byte. Then, the sum must be multiplied by 8 and divided by the simulation duration in order to obtain the goodput in bps.

- **Average Packet Delay**: The 14^{th} position of all the lines starting with `RX` are already expressed in seconds. It is sufficient to collect all these values and compute their average.

- **Cell-Edge Throughput**: it is typically calculated as the 5%-ile of the throughput values achieved by the users. Similarly to the average user throughput, this can be extracted from the lines starting with `PHY_RX`, by summing the values on the 17^{th} field only when the 19^{th} field is 0. However, in this case, this should be done separately for each receiving user, i.e., for every different value of the 5^{th} field. Finally, the Empirical Cumulative Distribution Function (ECDF) of the values obtained for each user should be computed and the value corresponding to the 5% is taken as the cell-edge throughput.

Since the text trace contains much information, on the basis of the proposed approaches, further investigations are still possible in order to derive finer-grained or more detailed results.

At the time of this writing, 5G-air-simulator already comes with a number of GNU AWK [117] tools for processing the text trace of a simulation. For instance, `make_goodput` and `make_plr` compute the related KPIs, `make_cdf` computes the Empirical CDF of packet delays, the overall spectral efficiency for a given assigned bandwidth is given by `make_cell_spectral_efficiency` and `make_fairness_index` returns the Jain's fairness index, while `make_avg` is a general script to compute the average of text input.

2.5 User-defined Scenarios

5G-air-simulator already implements a wide set of simulation scenarios, willing to test and investigate SISO, MIMO, and Massive MIMO deployments, multicast and broadcast transmissions, high-speed use cases, and massive IoT and NB-IoT deployments. According to the guidelines provided by standardization documents and scientific literature, each simulation scenario implements specific network deployment, physical, channel, application, and mobility models. The test can be executed through a command-line instruction, that contains the name of the software executable, the name of the reference scenario to be investigated, and a list of parameters to generally control simulation details. For instance, they include the number of cells, the number of users, the number and type of applications, the user speed, physical and channel details, the scheduling algorithm, and some other technical details properly related to the technical component of interest.

Besides, one of the main advantages of 5G-air-simulator is that users can define additional customized simulation scenarios. The first simple way to customize the simulations is by considering a specific parameter set for the list of input arguments for each selected use case, to evaluate its performance. It is important to emphasize that this approach ensures an effective usage of the tool for users willing to evaluate the performance of customized use cases without requiring the editing of C++ sources.

As a second way to customize their simulations, users can extend the reference C++ library implementing a given use case and modify available settings. Possible modifications may refer to network deployment (e.g., position of base stations), physical settings (e.g, transmission mode, transmission power, number of transmitting and receiving antennas, and noise figure, plus, for base stations only, antenna bearing, e-tilt, antenna gain, horizontal and vertical beamwidth at 3 dB, feeder loss, and maximum horizontal and vertical attenuation), channel model (e.g., rural vs urban, path loss model, fading parameters), application-level (e.g., number and type of applications per user, time instant in which every single application starts and stops), and mobility models (e.g., static position, random direction, linear movement). However, differently from the previous approach, users have to modify the source code, by integrating available models within their simulation scenario. This activity requires a minimum level of knowledge of C++. In particular, a custom scenario can be created as a static function in a proper header file, which should also be included in the main program. In general,

a basic scenario includes an instance of `Simulator`, `NetworkManager`, `Flows Manager`, and `FrameManager` components, cells, gNBs, and UE objects, several applications, and the
`Simulator::Run()` function.

The third methodology offering the possibility to define custom simulation scenarios is rooted in the flexibility and extensibility of 5G-air-simulator. It can be adopted to effectively pursue new research questions arising from new applications/services and features, allowing researchers and practitioners to test, extend, and evaluate the advanced solutions for 5G. For instance, new technical components may be integrated to explore emerging research topics. Also, adding new mobility and application models is as simple as writing new classes derived from the baseline code. Clearly, this approach also requires a more complex intervention of tool users, due to the need for developing new functionalities starting from baseline C++ classes already available into the simulator.

2.6 Massive MIMO

Massive MIMO is an important transmission technique in cutting-edge communication systems. By using numerous antennas it is possible to increase throughput, spectral efficiency, and coverage, depending on channel conditions. Moreover, the scientific literature demonstrated that Massive MIMO allows serving many users simultaneously, with a mutual interference that approaches zero, while achieving a very large sum spectral efficiency [118]. As a consequence, it is not surprising that it emerged as a key technical component for the NR. The scientific literature proposes various methodologies for implementing Massive MIMO in mobile networks. The one taken into account for the developed 5G-air-simulator is based on the JSDM technique [119].

2.6.1 Theoretical Description of the Technical Component

Massive MIMO has been natively conceived to work in TDD operation mode since channel reciprocity limits the overhead due to the training procedure (i.e., the base station is able to learn the channel quality experienced by mobile terminals in the downlink by evaluating the signal received in the uplink)[59], [118], [120]. Nevertheless, several works investigated new approaches for implementing Massive MIMO in FDD operation mode [121]–

[125]. All of them introduce some strategies to deliver the downlink CSI feedbacks to the base station without incurring high overhead. Among the possible solutions available in the literature, JSDM is a very promising and well-known approach implemented in 5G-air-simulator. In line with the previous considerations, JSDM involves a 2-stage precoding scheme aimed at reducing the training overhead of Massive MIMO in FDD mode. Its block diagram is shown in Figure 2.10.

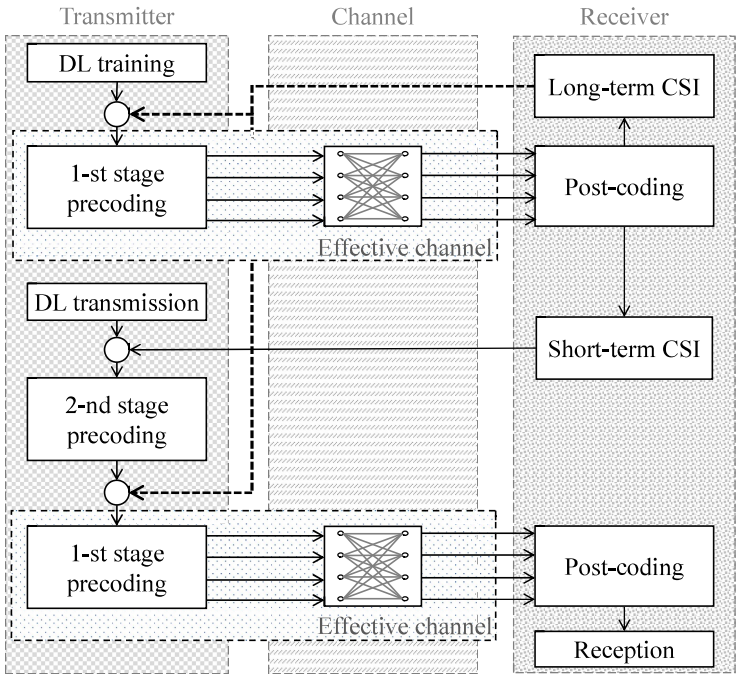


FIGURE 2.10: Block diagram of a JSDM Massive MIMO system.

From the system-level point of view, JSDM allows sending independent signals in fixed spatial directions, spanning the entire area of the cell sector. The resulting transmission is also referred to as Grid of Beams. To reach this goal, the precoding matrix is obtained as a combination of two matrices, one for each precoding stage. More specifically, the first stage precoder is used to capture the long-term and second-order channel statistics which are wideband and stable for a relatively long duration. At the receiver side, the precoded training signals are exploited to acquire long-term CSI, whose reporting ensures a limited training overhead. Instead, the second stage precoder is used to capture the short-term variation of the channel.

In this case, the resulting effective channel matrix has a reduced dimensionality, thanks to the utilization of the first-stage precoder. The second stage precoder is calculated by using the Regularized Zero-Forcing (RZF) scheme. Additionally, in order to improve JSDM performance levels, a

multi-cell interference reduction technique can be implemented on top of the first-stage precoding. The approach developed in the 5G-air-simulator uses the first-stage precoder also for beam coordination between base stations, hence achieving a flexible sub-sectorization of the covered geographical area. Specifically, three different configurations of beams are defined, which only cover a subset of the cell sector, instead of the entire sector as described above. Then, the configuration is frequently changed in a synchronized manner among nearby base stations in order to reduce interference.

2.6.2 Main Implementation Details

The single antenna transmission technique, already available in the original version of the LTE-sim tool, requires modeling the transmitted signal by means of a vector of elements describing the power density distributed across the available/selected subchannels. To implement MIMO and Massive MIMO mechanisms, 5G-air-simulator extends the baseline representation of the signal with a multidimensional approach. Rather than using a plain scalar number for each subchannel, the element of the transmitted or received signal becomes an array with a length equal to the number of transmitting or receiving antennas, respectively.

MIMO and Massive MIMO capabilities are mainly implemented in `ChannelRealization`, `PropagationLossModel`, `DownlinkPacketScheduler`, and `UePhy` classes, as summarized in Table 2.5.

TABLE 2.3: Extended path loss models available in 5G-air-simulator.

Name	Formula (dB)	Notes
Urban Macro-cell IMT (LOS, $d < d_{bp1}$) [102]	$22.0\log_{10}(d) + 28 + 20\log_{10}(0.001f)$	$d_{bp1} = 4(H_{gnb}-1)(H_{ue}-1)(f/300)$
Urban Macro-cell IMT (LOS, $d > d_{bp1}$) [102]	$40\log_{10}(d) + 7.8 - 18\log_{10}(H_{gnb}-1) - 18\log_{10}(H_{ue}-1) + 2\log_{10}(0.001f)$	
Urban Macro-cell IMT (NLOS) [102]	$161.04 - 7.1\log_{10}(20) + 7.5\log 10(H_b) - (24.37 - 3.7(H_b/H_{gnb})^2)) * \log_{10}(H_{gnb}) + (43.42 - 3.1\log_{10}(H_{gnb}))(\log_{10}(d) - 3) + 20\log_{10}(0.001f) - (3.2(\log_{10}(11.75H_{ue}))^2 - 4.97)$	
Urban Macro-cell IMT-3D (LOS, $d < d_{bp1}$) [109]	$22.0\log_{10}(d_{3D}) + 28 + 20\log_{10}(0.001f)$	$d_{bp1} = 4(H_{gnb}-1)(H_{ue}-1)(f/300)$
Urban Macro-cell IMT-3D (LOS, $d > d_{bp1}$) [109]	$40\log_{10}(d_{3D}) + 28 + 20\log_{10}(0.001f) - 9\log_{10}(d_{bp1}^2 + (H_{gnb}-H_{ue})^2)$	
Urban Macro-cell IMT-3D (NLOS) [109]	$161.04 - 7.1\log_{10}(20) + 7.5\log_{10}(H_b) - (24.37 - 3.7(H_b/H_{gnb})^2)\log_{10}(H_{gnb}) + (43.42 - 3.1\log_{10}(H_{gnb}))(\log_{10}(d_{3D}) - 3) + 20\log_{10}(0.001f) - (3.2\log_{10}(17.625)^2 - 4.97) - 0.6(H_{ue}-1.5)$	
Rural Macro-cell IMT (LOS, $d < d_{bp}$) [102]	$20\log_{10}(40\pi d(0.001f/3)) + \min(0.03H_b^{1.72}, 10.00)\log_{10}(d) - \min(0.044H_b^{1.72}, 14.77) + 0.002\log_{10}(H_b)d$	$d_{bp} = 2\pi H_{gnb}1.5(f/300)$
Rural Macro-cell IMT (LOS, $d > d_{bp}$) [102]	$20\log_{10}(40\pi d_{bp}(0.001f/3)) + min(0.03Hb^{1.72}, 10.00)\log_{10}(d_{bp}) - min(0.044Hb^{1.72}, 14.77) + 0.002\log_{10}(H_b)d_{bp} + (40\log_{10}(d/d_{bp}))$	
Rural Macro-cell IMT (NLOS) [102]	$161.04 - 7.1\log_{10}(20) + 7.5\log_{10}(H_b) - (24.37 - 3.7(H_b/H_{gnb})^2) * \log_{10}(H_{gnb}) + (43.42 - 3.1\log_{10}(H_{gnb}))(\log_{10}(d) - 3) + 20\log_{10}(0.001f) - (3.2\log_{10}(11.751.5)^2 - 4.97)$	

TABLE 2.4: Calibration parameters for urban scenario

Parameter	**Value**
Carrier Frequency	2.0 GHz
Inter Site Distance	500 m
Bandwidth	10 MHz
Penetration Loss	0 dB
Speed	30 km/h
Cellular layout	Hexagonal grid, 19 cell sites, 3 sectors per site
Number of users per cell	10
Antenna pattern (horizontal)	$\phi_{3dB} = 70^{\circ}, A_m = 20dB$
Antenna pattern (vertical)	$\theta_{3dB} = 15, A_m = 20dB$
Noise Figure	5 dB
Base station max antenna gain	17 dBi
Base Station Antenna height	25 m
Total BS transmit power	46 dBm
Minimum distance between UE and Serving Cell	25 m
Duplex method	FDD
Downlink transmission scheme	1x2 SIMO
Downlink Scheduler	Round robin with full bandwidth allocation
CQI reporting	Wideband CQI, 5 ms periodicity, 6 ms delay total. CQI measurement error: None.
Downlink HARQ	Maximum four transmissions
Downlink HARQ	Maximum four transmissions
Downlink receiver type	MRC
Antenna configuration	Vertically polarized antennas 0.5 wavelength separation at UE, 10 wavelength separation at base station
Channel estimation	Ideal, both demodulation and sounding
BS antenna downtilt	12 deg
BS feeder loss	2 dB

TABLE 2.5: Main functions related to MIMO and Massive MIMO features.

Key functionality	**Class**	**Method**	**Parameters**
Allocate the fast fading component of the channel model, considering the number of transmitting and receiving antennas	`ChannelRealization`	`enableFastFading()`	(none)
Update the fast fading realization	`ChannelRealization`	`UpdateFastFading()`	(none)
Get the propagation loss for each channel path	`ChannelRealization`	`GetLoss()`	(none)
Get the beamforming gain for a specific beam	`ChannelRealization`	`GetBeamformingGain()`	beam index
Get the beamforming gain for a specific beam and coordination pattern	`ChannelRealization`	`GetBeamforming Gain_cover()`	beam index, beam coordination pattern index
Apply the propagation loss to a transmitted signal	`PropagationLossModel`	`AddLossModel()`	source node, destination node, signal
Allocate RBs to users, supporting multiple allocation with Massive MIMO	`DownlinkPacket Scheduler`	`RBsAllocation()`	(none)
Receive a radio signal and calculate the SINR, including MIMO post-processing when required	`UePhy`	`StartRx()`	packet burst, signal
Create channel state feedbacks, including those for MIMO and Massive MIMO when required	`UePhy`	`CreateCqiFeedbacks()`	SINR vector

Transmission and reception procedures are handled by `DownlinkPacketScheduler` and `UePhy` classes, respectively. `DownlinkPacketScheduler` implements the transmitter process, including radio resource allocation, precoding operation, and simultaneous transmission to multiple users, which is a key advantage of Massive MIMO. On the other hand, `UePhy` is in charge of carrying out the reception of the radio signals and the SINR computation, as well as the calculation of CSI, when required. These values are forwarded to the base station, similar to the CQI report, where they are used for the precoding operation. In order to perform the SINR computation, `ChannelRealization` and `PropagationLossModels` classes are used. In particular, the `ChannelRealization` class deals with fast fading and beamforming, according to the selected channel model. Instead, propagation loss is applied to each transmitted signal through the class `PropagationLossModels`. The SINR and CSI feedbacks are computed according to the procedures described in subsection 2.3.1, which mimic the procedures already present in both LTE and NR standards.

2.6.3 Reference Test and Results

The performance of the Massive MIMO transmission scheme has been investigated by considering a practical implementation envisaged within the FANTASTIC-5G EU H2020 project. The reference scenario is based on the `test-multi-cell-tri-sector` deployment and each base station is equipped with a rectangular antenna array of 256 elements (16 horizontal elements $\times$ 8 vertical $\times$ 2 polarizations) [126]. At the application layer, the model is the `InfiniteBuffer` while the mobility model is `RandomWalk`. The first stage precoder is configured in order to achieve 16 horizontal beams with two alternating elevation angles. The packet scheduler supports 2 spatially multiplexed data streams per user. The scenario is called `f5g-uc1` because it models the use case 1 in the project's documentation and the syntax used to perform the test is as follows:

```
$ ./5G-air-simulator f5g-uc1 env isd density speed time tm nTx n
Mu nRx sched (seed)
```

where

- `env` is the propagation environment used for channel models;
- `isd` is the Inter-Site Distance (ISD) in km;
- `density` is the user density measured in users/km^2;

- `speed` is the speed of the mobile users in km/h;
- `time` is the duration in seconds of each simulation run;
- `tm` is the TM adopted for transmission, where values 1, 2, 3, 4, 9 have the same meaning as in the LTE specifications, and 11 represents Massive MIMO;
- `nTx` is the number of beams used at the base station for the first-stage precoding;
- `nMu` is the number of users that can be scheduled simultaneously for each RBs and TTI;
- `nRx` is the number of receiving antennas at the mobile terminals;
- `sched` is the scheduling algorithm;
- `seed` is an optional seed to initialize random quantities to different, but reproducible, values in each simulation run.

The performance of the developed Massive MIMO technique is evaluated in urban, suburban, and rural scenarios, with different parameter settings, as reported in Table 2.6. The main performance indicator that was considered

TABLE 2.6: Adopted Values for the Parameters of the Scenario

Parameter	**Value**		
`environment`	"urban"	"suburban"	"rural"
`isd`	0.2 km	0.6 km	1 km
`userDensity`	2500 users/km^2	400 users/km^2	100 users/km^2
`speed`	3 km/h		
`duration`	10 s		
`tm`	9, 11		
`nbTx`	32 beams		
`nbMu`	8 users		
`nbRx`	2 antennas		
`sched`	round-robin		
`seed`	1-30		

is the user throughput. The results reported in Figure 2.11 demonstrate that Massive MIMO, implemented with the JSDM technique, can provide huge throughput gains over state-of-the-art LTE capabilities, up to around 800% when the beam coordination technique is employed.

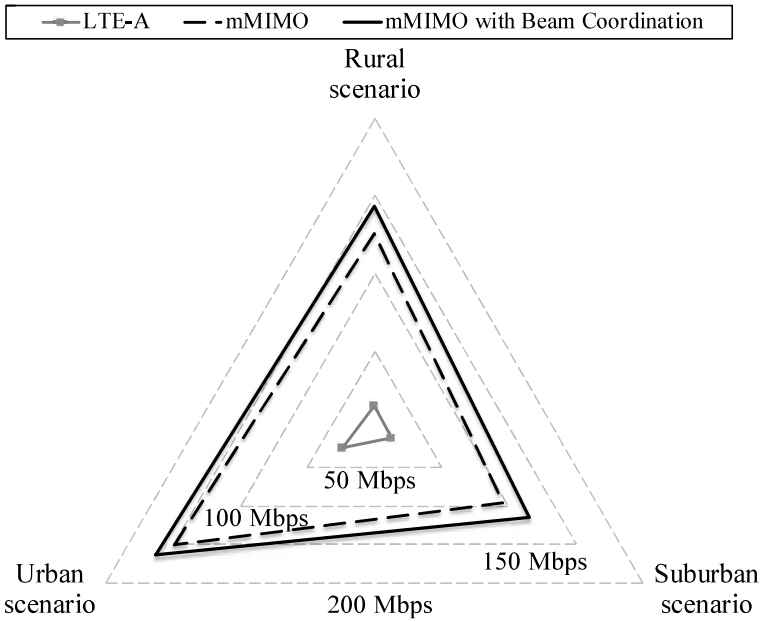


FIGURE 2.11: Throughput comparison between MIMO and Massive MIMO.

It is important to note that the subtle differences among different scenarios are also caused by a different traffic density (i.e. Gbps/km^2) characterizing the environments, as reported in Figure 2.12.

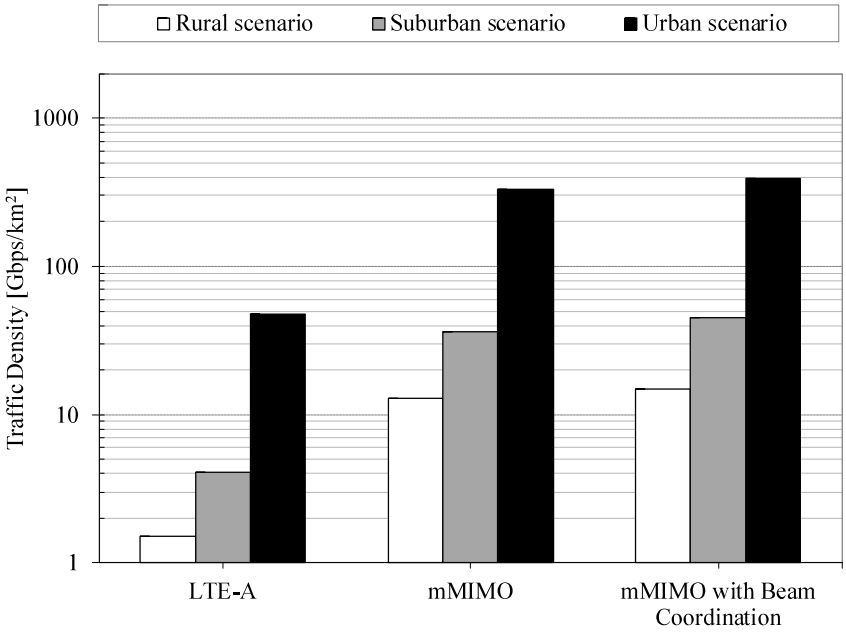


FIGURE 2.12: Traffic densities of the evaluated scenarios

2.7 Extended Multicast and Broadcast Transmission

Since the radio channel is intrinsically a shared medium, multicast and broadcast communication can be attained at the physical layer with relatively low complexity and high efficiency. This mode of operation can be very useful in some specific circumstances, e.g. for video broadcasting of a live event, software upgrades, and so on. Since LTE Release 9, the support of multicast and broadcast communications has been provided through the MBSFN architecture [19]. Today MBSFN still represents a technical component for 5G though some upgrades have been recently proposed in the scientific literature. Among them, adaptive MCS selection and HARQ retransmission [127] are those modeled within 5G-air-simulator.

2.7.1 Theoretical Description of the Technical Component

With MBSFN, multiple base stations work in a coordinated manner to define a MBSFN area and the broadcast signal is transmitted on the entire bandwidth during pre-defined time slots. This mechanism brings two main advantages. First, an extremely high bandwidth saving can be achieved, since many users can be served by using the same set of radio resources. Second, signals from multiple surrounding base stations can combine constructively if their delays are within the cyclic prefix duration, thanks to the properties of OFDM [128]. This improves communication performance and, most importantly, removes the main sources of interference.

A major drawback of the initial MBSFN architecture is that there is no reverse link between the base stations and the users. Thus, the base station has no knowledge of the users' channels, or whether packets are correctly received. For this reason, the MCS selection has to be very conservative and thus possibly inefficient.

5G intends to overcome this important limitation by introducing novel methodologies aiming at improving the overall communication process. In line with these premises, the EU H2020 FANTASTIC-5G project proposed to extend MBSFN beyond the original 3GPP standard with adaptive MCS selection and HARQ retransmissions functionalities [129]. Both rely on the introduction of a unicast uplink feedback channel for the MBSFN downlink channel. For the adaptive MCS selection, users send CQI feedbacks describing the quality of the broadcast channel, and the base stations perform the

Link Adaptation procedure by selecting the most suitable MCS to be used for transmission so that all the users (or any desired fraction of them) can correctly receive the transmitted data. As for the HARQ retransmission, instead, users send an ordinary ack/nack feedback, i.e., as in the unicast case, and the packets that are lost are transmitted again. However, while the first transmission takes place on the MBSFN channel, subsequent re-transmissions are sent in unicast mode only to the interested users, so that more efficient techniques such as MIMO can be employed. Figure 2.13 shows the block diagram of the extended MBSFN architecture just discussed.

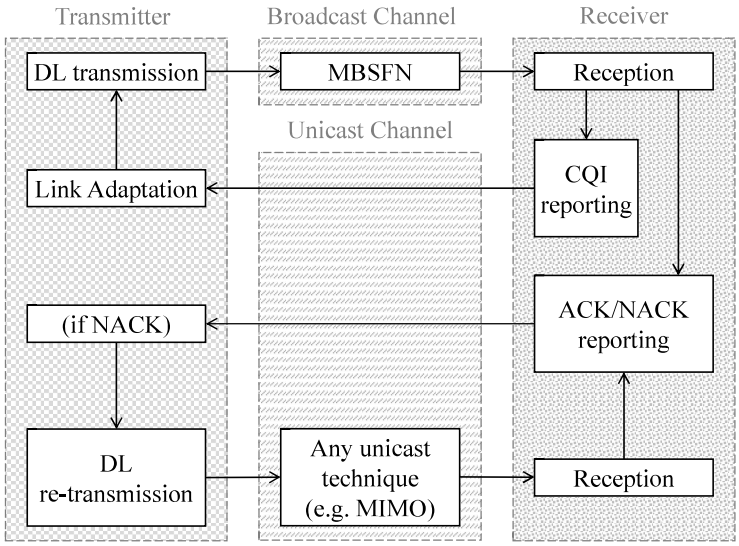


FIGURE 2.13: Block diagram of a MBSFN system with proposed extensions.

2.7.2 Main implementation details

Table 2.7 summarizes the main methods and classes involved in the multicast/broadcast operation.

TABLE 2.7: Main functions related to multicast/broadcast features.

Key functionality	**Class**	**Method**	**Parameters**
Create a multicast group on the given base station	`MulticastDestination`	`MulticastDestination()`	ID, cell, target BS
Add a base station to the multicast group	`MulticastDestination`	`AddSource()`	node
Remove a base station from the multicast group	`MulticastDestination`	`DeleteSource()`	node
Get the list of base station in the multicast group	`MulticastDestination`	`GetSources()`	(none)
Add a user equipment to the multicast group	`MulticastDestination`	`AddDestination()`	node
Remove a user equipment from the multicast group	`MulticastDestination`	`DeleteDestination()`	node
Get the list of user equipments in the multicast group	`MulticastDestination`	`GetDestinations()`	(none)
Create the physical layer interface for a multicast group	`MulticastDestination Phy`	`MulticastDestination Phy()`	(none)
Create the CQI feedbacks for a multicast group	`MulticastDestination Phy`	`CreateCqiFeedbacks()`	(none)
Compute the inter-cell interference for a user equipment, assuming useful signal from base stations in the same multicast group	`Interference`	`ComputeInterference()`	user equipment, flag for MBSFN interference model
Set the number of sub-frames dedicated to MBSFN in each radio frame	`FrameManager`	`SetMbsfnPattern()`	number of MBSFN sub-frames
Check whether MBSFN is enabled	`FrameManager`	`MbsfnEnabled()`	(none)
Check whether the current sub-frame is a MBSFN sub-frame	`FrameManager`	`isMbsfnSubframe()`	(none)
Allocate RBs in the downlink. During MBSFN sub-frames, all the RBs are allocated to all the users of the multicast group	`DownlinkPacket Scheduler`	`RBsAllocation()`	(none)

To implement the baseline MBSFN architecture in the 5G-air-simulator, the classes `MulticastDestination` and `MulticastDestinationPhy` are derived from the classes `UserEquipment` and `UePhy`, respectively. `MulticastDestination` contains pointers to all the users receiving the broadcast signal. With this information, when an application flow is created with a `MulticastDestination` object as the receiver, a radio bearer sink for the corresponding radio bearer is created for each receiving user so that the transmitted packets are received by all the users belonging to the multicast group. At the same time, the downlink packet scheduler distributes RBs between unicast and multicast/broadcast communications according to a well-defined MBSFN frame structure [130]. Specifically, the `FrameManager` class is extended with the notion of MBSFN sub-frames and non-MBSFN sub-frames. Scheduling and transmission of MBSFN and non-MBSFN applications flow only happens during the corresponding sub-frame type. According to the specifications, up to 6 sub-frames in a radio frame can be used for MBSFN operation, whose pattern is defined by using the `FrameManager::SetMbsfnPattern()` method and repeated cyclically for the entire simulation. Besides, the `Interference::ComputeInterference()` function already available in the original version of the LTE-Sim tool, has been extended in order to account for MBSFN operation: all the signals coming from the same MBSFN area are treated as useful signal rather than interference.

The adaptive MCS selection scheme is mainly implemented by the `MulticastDestinationPhy::CreateCqiFeedbacks()` method. In summary, it manages the collection of CQIs feedbacks of users receiving the multicast stream, and it helps the packet scheduler in the selection of the most suitable MCS index to be used for future multicast/broadcast communications (i.e., link adaptation). Particularly, the MCS selection may be done either by considering the absolute lowest CQI value of the multicast group or to ensure the correct reception of a pre-determined percentage of users.

As for the HARQ retransmissions, most of the features are modeled within the `DownlinkPacketScheduler` class. In `DownlinkPacketScheduler::RBsAllocation`, when a multicast flow is scheduled, multiple copies of the corresponding `PacketScheduler::FlowToSchedule` structure are created for each destination user, and they are added to the corresponding `HarqManager` object. Additionally, in `DownlinkPacketScheduler::DoStopSchedule()`, the packet burst of the original flow is copied in the duplicated flows. From then on, they act as any other unicast flow, including retransmissions.

2.7.3 Reference Test and Results

An example network configuration taken into account to evaluate the performance of the aforementioned MBSFN architecture is shown in Figure 2.14.

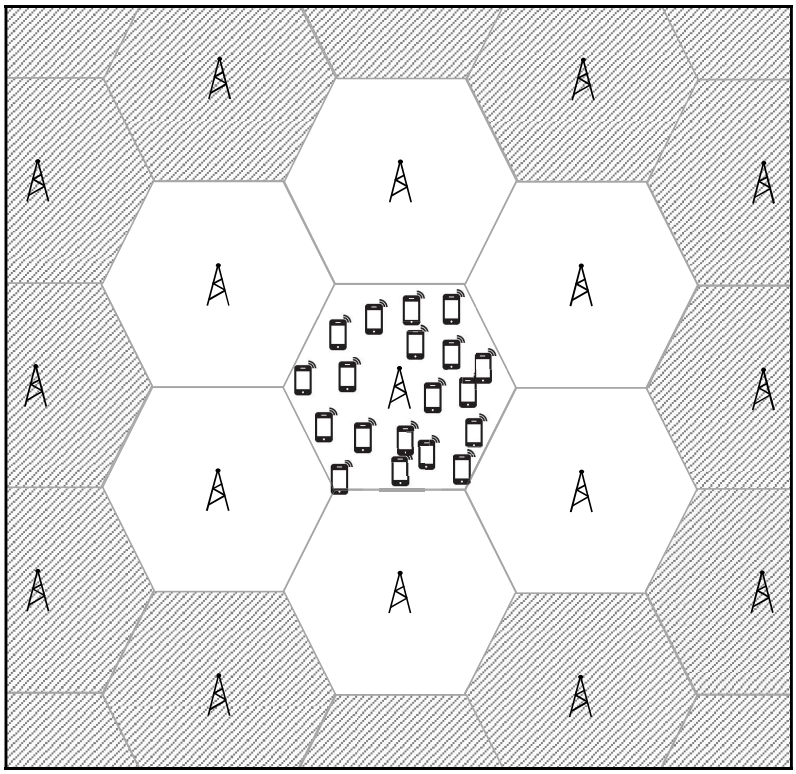


FIGURE 2.14: Scenario for multicast/broadcast use case evaluation.

The clear cells are part of the MBSFN area, while the shaded cells are excluded and count as interference sources. The end users are only located in the center cell, while the other cells of the MBSFN area act as buffers to reduce the interference. This is called the *assisting ring* arrangement, and it is employed to reduce the interference at the edges of the MBSFN area. Note that the layout of Figure 2.14 models only one serving cell and one assisting ring, but many possibilities can be implemented, including irregularly shaped areas of many cells and multiple assisting rings. The performance achieved with MBSFN, both with and without the 5G extensions, has been investigated by considering a practical implementation envisaged within the FANTASTIC-5G EU H2020 project. The reference scenario instantiates an MBSFN network with one serving cell, containing the users, and two assisting rings around it, which are then surrounded by a ring of interfering cells. During the MBSFN sub-frames, the base stations of the MBSFN area

transmit an HD video flow with a bit rate of 17 Mbps, therefore at the application layer, the model is the `TraceBased`. The mobility model is `Constant Position` while the scheduling algorithm is round-robin. The scenario is called `f5g-uc6` because it models the use case 6 in the project's documentation and the command-line syntax to use it is:

```
./5G air simulator f5g uc6 env isd density pattern time mcs harq (seed)
```

where

- `env` is the propagation environment used for channel models, either "suburban" or "rural";
- `isd` is the ISD distance in km;
- `density` is the user density measured in users/km^2;
- `pattern` is the number of sub-frames to reserve for MBSFN, from 0 to 6, where 0 disables MBSFN;
- `duration` is the duration in seconds of each simulation run;
- `mcs` is the MCS to use for transmission, it can be set to a fixed value from 0 to 28, or the value -1 can be used to enable the automatic selection based on CQI feedbacks;
- `harq` indicates whether to enable HARQ retransmission of broadcast packets or not, using the values 1 or 0, respectively;
- `seed` is an optional seed to initialize random quantities to different and reproducible values for each simulation run.

The performance of MBSFN and its 5G extensions is evaluated with the parameter settings shown in Table 2.8. Specifically, two main performance in-

TABLE 2.8: Adopted Values for the Parameters of the Scenario

Parameter	Value
`environment`	suburban
`isd`	0.6 km
`userDensity`	400 users/km^2
`mbsfnPattern`	6
`duration`	10 s
`mcs`	16, 18, 20, -1
`use_harq`	1 with adaptive MCS, 0 otherwise
`seed`	1-30

dicators have been considered: the cell-edge throughput at the application

layer and the PLR. Figure 2.15 shows the Cumulative Distribution Function (CDF) plot from 0% to 5% for the considered scenarios, so that the cell-edge throughput values are at the top of the curves.

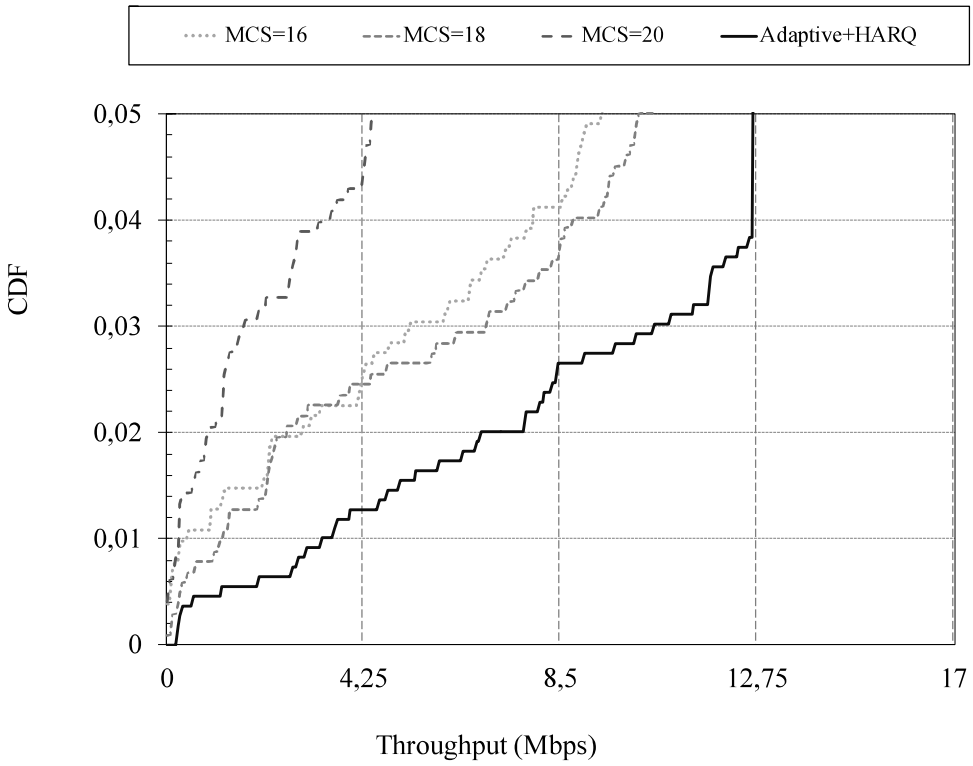


FIGURE 2.15: Cell-edge throughput at the application layer in the broadcast test.

The use of adaptive MCS with HARQ retransmissions allows more than 25% increase in cell-edge throughput compared to the best situation with a fixed MCS, while also avoiding the necessity to find the optimal value beforehand. Figure 2.16 shows the CDF plot for all the registered throughput values, also confirming previous results.

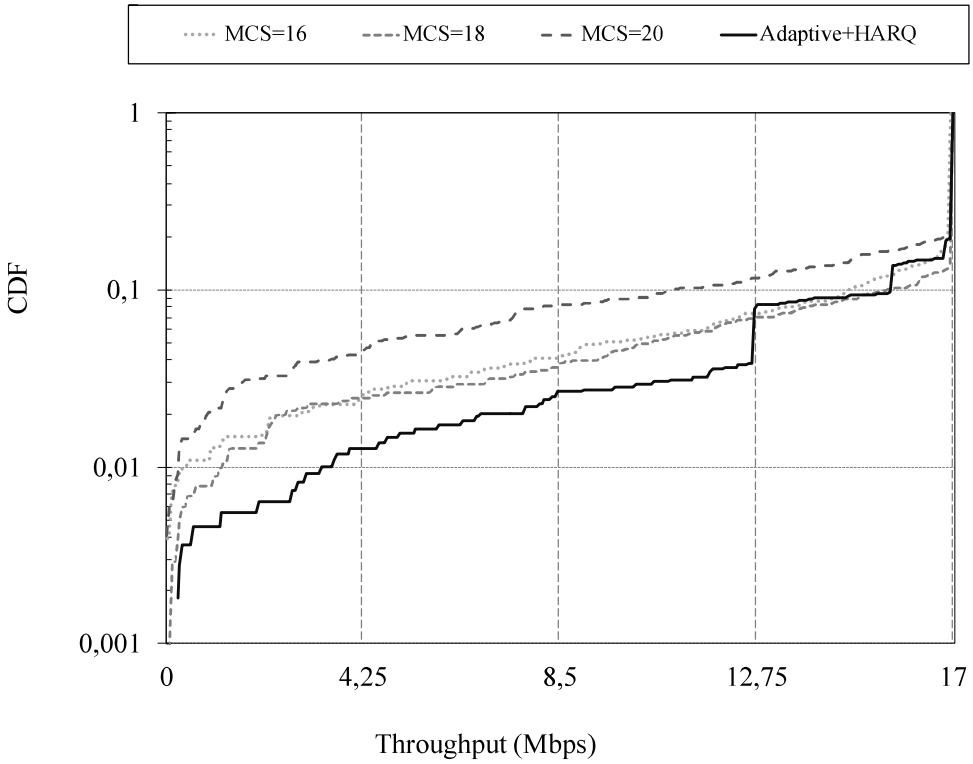


FIGURE 2.16: Throughput at the application layer in the broadcast test.

As for the PLR, Table 2.9 shows the results of the simulations, demonstrating that the introduction of the new extensions added on top of MBSFN also reduce the PLR values compared to MBSFN alone. They also achieve a

TABLE 2.9: PLR registered in broadcast test.

	Modulation and Coding Scheme (MCS)			
	16	**18**	**20**	**Adaptive+HARQ**
Average PLR (%)	7.3	5.1	9.0	3.3

better trade-off between throughput and PLR, as the best configuration for cell-edge throughput of the baseline MBSFN is not the same as the best one in terms of PLR, but the adaptive solution is better than both of them at the same time.

2.8 High-speed environment and predictor antennas

When a mobile user is moving at a considerably high speed, such as 250 km/h or more, the quality of the radio link drops substantially (this can

be the case of fast trains traveling long distances [131]). As expected, the reduced quality of the radio link results in a reduced capacity. Nevertheless, at the same time, it would be desirable to maintain a good QoS also in high-speed environments. Transmission schemes adopting predictor antennas emerged as a valuable technical component for achieving this goal in 5G.

2.8.1 Theoretical Description of the Technical Component

In general, one of the main causes of performance loss is the aging of the CSI between acquisition and utilization. The impact of this aging becomes disrupting in high-speed scenarios, where the channel quality rapidly changes also because of the interference originated by the Doppler spread. Without loss of generality, this Section considers a challenging use case of interest for the 5G, which is a high-speed train. Here, the SRTA-PI [131] emerged as a powerful technical component able to improve the connection quality by tackling the CSI aging phenomenon through an array of antennas deployed on top of the train. In other words, the train hosts a relay device offering wireless connectivity to passengers. According to the proposed technique, the antenna array samples the channel in different positions. Intermediate positions are obtained through interpolation of known positions so that the channel for the intended receiving antennas can be estimated with great accuracy despite the train's movement. For this reason, the antennas of the array act as predictor antennas [132]. Figure 2.17 shows the complete workflow for the SRTA-PI technique.

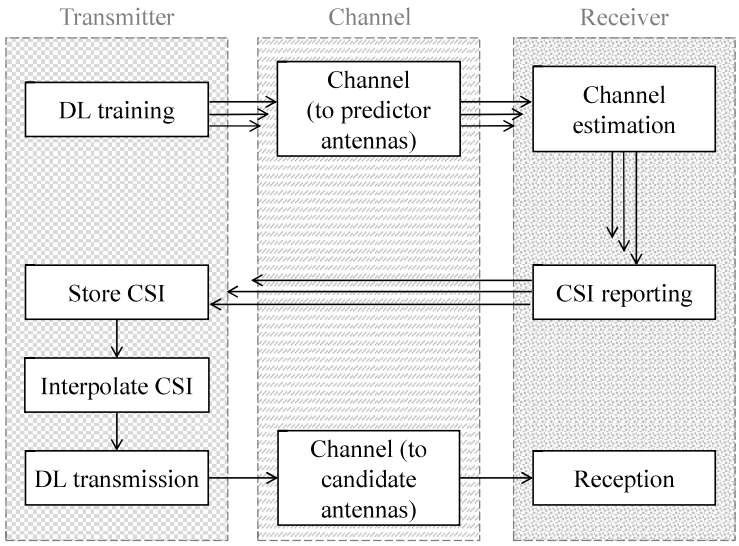


FIGURE 2.17: Block diagram of the SRTA-PI technique.

During the training phase, the predictor antennas sample the channel and the receiver sends back those estimated samples to the base station during the CSI reporting phase. The base station stores the CSI obtained at regular intervals so that it can estimate the channel for any spatial position falling between the sampled positions by interpolation. By also knowing the position and speed of the receiver, it can determine the current position of the candidate receiving antennas and use the appropriate precoder.

2.8.2 Main implementation details

Table 2.10 summarizes the main functions related to the high-speed simulation scenarios.

TABLE 2.10: Main functions related to high speed simulations.

Key functionality	Class	Method	Parameters
Enable or disable the SRTA-PI technique for accurate channel estimation at thigh speeds.	`Phy`	`SetSrtaPi()`	flag to enable or disable SRTA-PI
Set the waveform type, which affect the amount of Doppler spread interference at high speeds.	`Phy`	`SetWaveformType()`	waveform type
Calculate the Doppler interference produced at a given speed and for a given waveform type.	`Interference`	`ComputeDoppler Interference()`	speed, waveform type
Receive a radio signal and calculate the SINR. With SRTA-PI enabled, consider the same channel realization for precoding and reception.	`UePhy`	`StartRx()`	packet burst, signal
Allocate RBs in the downlink. When using SRTA-PI, associate the transmitted signal with the channel realization used during precoding.	`DownlinkPacket Scheduler`	`RBsAllocation()`	(none)

`Phy::SetSrtaPi()` sets the use of the SRTA-PI technique, that is disabled by default. It should be invoked for each device that adopts such a technique before the simulation is started. `Phy::SetWaveformType (Phy::Waveform Type)` allows the selection of the waveform, to model Doppler spread interference. At the time of this writing, the selection is limited to OFDM, pulse-shaped OFDM, and an ideal waveform that is immune to Doppler spread. Doppler spread interference is then drawn from look-up tables as a function of the speed and the transmission power. Finally, the association between the signal and the corresponding channel realization is conducted in `Downlink PacketScheduler::RBsAllocation()` and `UePhy::StartRx()`, for transmission and reception procedures, respectively.

2.8.3 Reference Test and Results

The performance achieved with the aforementioned SRTA-PI technique has been investigated by considering a practical implementation envisaged within the FANTASTIC-5G EU H2020 project. Specifically, the network layout used to evaluate the SRTA-PI technique is shown in Figure 2.18.

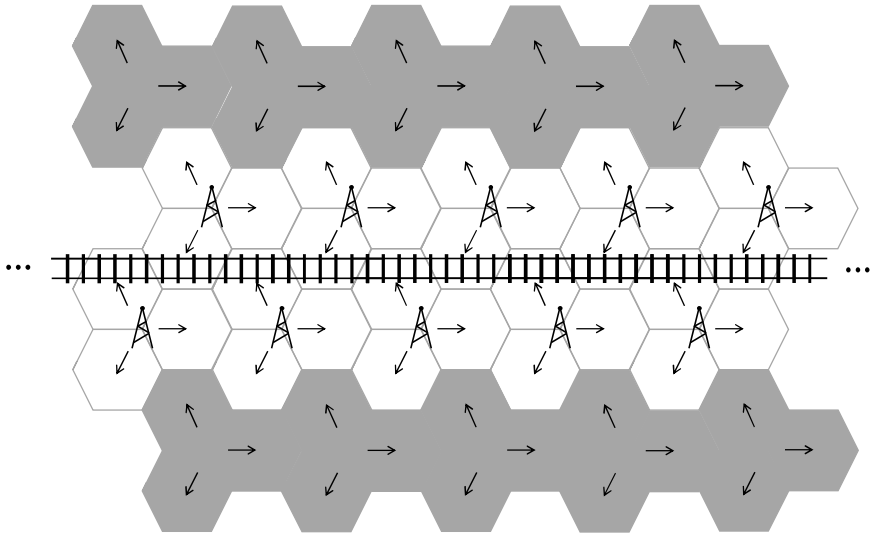


FIGURE 2.18: Reference scenario for the high-speed use case.

The reference scenario considers a train that moves on a straight horizontal line, with three-sectored base stations placed at both sides of the track in a hexagonal grid. There are two rows of base stations at each side, where the first one is used for service and the second one for modeling interference from the rest of the network. This layout is extended indefinitely by using

wrap-around in the horizontal direction, so that movement can be extended for as long as needed without increasing the complexity. The mobile users are on the train, which acts as a relay station between them and the base stations. In this regard, the train can act as a single mobile user from the network's point of view, or as multiple users, depending on how many receiving units are installed. At the application layer, the model is the `Infinite Buffer` while the mobility model is `LinearMovement`. The scenario is called `f5g-uc2` because it models the use case 2 in the project's documentation and the syntax used to perform the test is as follows:

```
./5G air simulator f5g uc2 env isd nUe speed time nTx nM nRx sched
srta wfIdx (seed)
```

where

- `env` is the propagation environment used for channel models, either "suburban" or "rural";
- `isd` is the inter-site distance in km;
- `nUE` is the number of receiving units on the train;
- `speed` is the speed of the train in km/h;
- `time` is the duration in seconds of each simulation run;
- `nTx` is the number of beams used at the base station for the first-stage precoding;
- `nM` is the number of users that can be scheduled simultaneously for each RBs and TTI;
- `nRx` is the number of antennas used at each receiving unit, excluding predictor antennas;
- `sched` is the scheduling algorithm;
- `srta` indicates whether to enable SRTA-PI or not, using values 1 and 0 respectively;
- `wfIdx` indicates the waveform type to consider for Doppler spread calculation, where 0 is OFDM, 1 is pulse-shaped OFDM, and 2 is an ideal waveform without Doppler spread interference;
- `seed` is an optional seed to initialize random quantities to different and reproducible values for each simulation run.

The performance of the developed SRTA-PI technique is evaluated when the train is moving at different speeds, with different parameter settings, as reported in Table 2.11.

TABLE 2.11: Adopted Values for the Parameters of the Scenario

Parameter	**Value**
`environment`	"suburban"
`isd`	0.5, 1, 2 km
`nUE`	8 receiving units
`speed`	30, 120, 250, and 500 km/h
`time`	(it depends on the speed, so that the distance traveled is always the same)
`nTx`	32 beams
`nM`	8 users
`nRx`	2 antennas
`sched`	round-robin
`srta`	0, 1
`wfIdx`	0
`seed`	1-30

The most relevant outcome of this test is the cell throughput, which shows how SRTA-PI improves the overall performance, at the expense of greater complexity. The results obtained by either applying or not the SRTA-PI technique are shown in Figure 2.19.

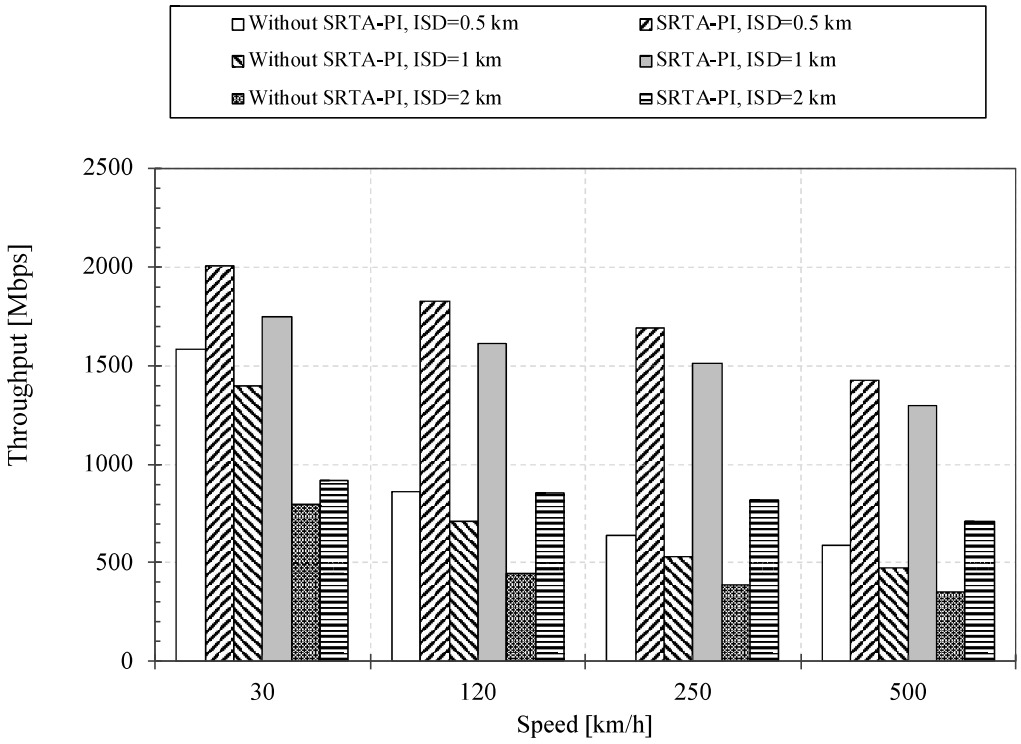


FIGURE 2.19: Throughput achieved in SRTA-PI test.

Results show that SRTA-PI can provide a throughput gain up to 100% at the highest speed compared to classic transmissions. However, the effect of Doppler spread still leaves a visible reduction of the performance at higher speeds.

2.9 The Enhanced Random Access Procedure

5G-air-simulator natively provides the support for the random access procedure, which endorses mobile terminals to establish a connection with the base station, without requiring any previously shared information. The presented version of the simulator supports both the baseline 5G contention-based random access procedure [25] and an enhanced procedure, as described in [133], [134].

2.9.1 Theoretical Description of the Technical Component

The random access procedure is per se contention-based: since numerous users can access the shared channel at the same time, collisions may occur.

The contention can be prevented, leading to a contention-free Random Access CHannel (RACH) procedure. Since this happens infrequently, a key aspect characterizing a RACH procedure is the contention resolution. The baseline 5G random access procedure standardized by 3GPP [135], which is initiated by the mobile terminal, is based on a 4-message handshake whose purpose is two-folded. Users achieve tight timing synchronization with the base station, on the one hand, and they receive an allocation grant of uplink resources, on the other. The first message can only be sent during a Random Access Opportunity, which is periodically scheduled by the base station. It consists of a preamble sequence, randomly chosen from a set of orthogonal sequences. The main goal of the preamble is to indicate the presence of an access request and to allow the base station to estimate the distance of the mobile terminal for the Timing Advance procedure. If two or more devices choose the same preamble during the same Random Access Opportunity, a collision occurs and the procedure may fail either immediately or at a later stage. Upon proper preambles detection, the base station sends back a Random Access Response (RAR) message. It consists of relevant information for each detected preamble, including the specific uplink resources to be used for sending the third message. If a collision is detected for a specific preamble, then the corresponding RAR will not be sent and the devices retry the procedure after a specific waiting time. Conversely, if a collision is unrecognized, two or more users will be assigned the same uplink resources and they will collide again on the third message, namely the Connection Request, that will be lost. Finally, after the successful delivery of the Connection Request, the base station replies with the last message, which is the Contention Resolution, also known as Msg4. When a device receives a Contention Resolution message addressed to it, the random access procedure is assumed to be completed. From this moment on, the successful devices can have reliable, collision-free communication with the base station. On the contrary, if the Msg4 is not properly received the Random Access Procedure has to be restarted.

As mentioned above, this procedure has been enhanced in [133], which emerged as a potential technical component for 5G systems. Specifically, after the base station performs the detection of preambles, it sends back a RAR containing multiple responses for each identified preamble, with different uplink resources assigned. Every mobile terminal which receives the RAR can randomly choose one of the uplink resources reserved for the preamble,

selected during the first step. Thanks to this additional randomness, a collision over the selection of the preamble does not translate to a failure in the access procedure. This way, if two or more mobile terminals use the same preamble, they still have a chance to select a different resource assignment in the RAR and thus avoid the collision at the third step of the protocol. Basically, the multiple RARs act as multipliers for the number of preambles. The downside of this technique is that a correspondingly larger amount of uplink resources must be reserved for the transmission of Connection Request messages, which shrinks the resources available for actual user data.

Figure 2.20 summarizes the enhanced random access procedure.

2.9.2 Main Implementation Details

Most of the random access features are summarized in Table 2.12.

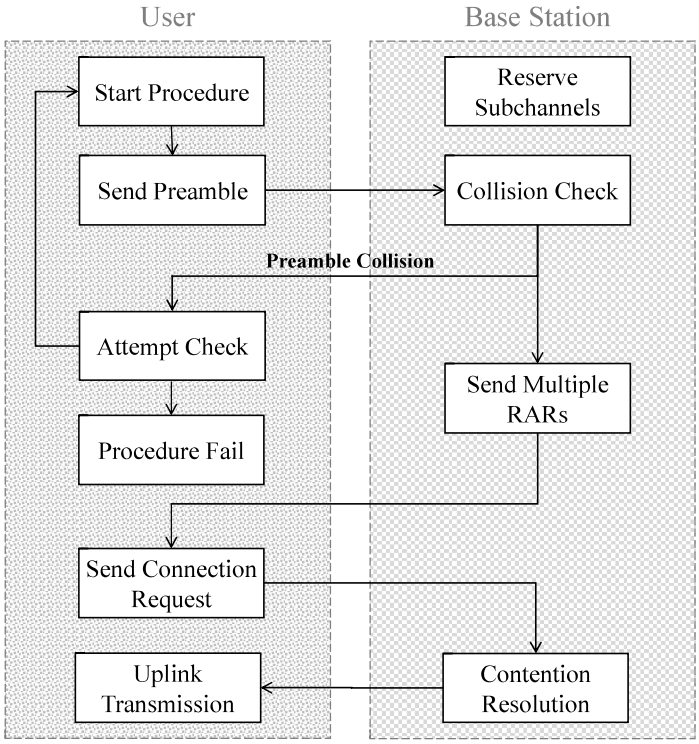


FIGURE 2.20: Block diagram of the enhanced random access procedure.

TABLE 2.12: Main functions related to random access.

Key Functionality	Class	Method	Parameters
Choose the random access procedure strategy to use for the base station	`GnbRandomAccess`	`SetGnbRandomAccessType()`	Random access strategy
Choose the random access procedure strategy to use for users	`UeRandomAccess`	`SetUeRandomAccessType()`	Random access strategy
Determine if the timeout is expired	`UeBaselineRandomAccess`	`checkRAProcedureTimeout()`	(none)
Start the random access procedure	`UeBaselineRandomAccess`, `UeEnhancedRandomAccess`	`StartRaProcedure()`	(none)
Restart the random access procedure if the user is allowed to	`UeBaselineRandomAccess`, `UeEnhancedRandomAccess`	`ReStartRaProcedure()`	(none)
Reserve radio resources for RACH	`GnbEnhancedRandomAccess`, `GnbBaselineRandomAccess`	`SetRachReservedSub Channels()`	(none)
Verify whether sent preambles collided	`GnbEnhancedRandomAccess`, `GnbBaselineRandomAccess`	`CheckCollisions()`	(none)
Check whether radio resources have been reserved for RACH in current TTI	`GnbEnhancedRandomAccess`, `GnbBaselineRandomAccess`	`isRachOpportunity()`	(none)

Two different classes model the base station and the user general behavior, that are `GnbRandomAccess` and `UeRandomAccess`, respectively. Different random access procedure strategies are then selected thanks to `GnbRandomAccess:SetGnbRandomAccessType()` and `UeRandomAccess::SetUeRandomAccessType()` methods.

As for the user side, as soon as the application-layer traffic generator creates a packet, the user MAC entity starts the random access procedure immediately by calling the appropriate `StartRaProcedure()` method. On the other hand, the base station periodically scans all the preambles sent in RACH resources in order to find potential collisions. This is done through the `CheckCollisions()` method. In case of a preamble collision, the procedure fails for all the devices involved, which then need to call the `ReStartRaProcedure()` method. The latter is also in charge of verifying whether the maximum number of retry attempts has been reached, and in the affirmative case, it stops immediately the procedure. Conversely, the base station sends the second message to those devices that successfully completed the first message transmission. Lastly, the message flow advances until the procedure is properly complete, making the end-user active and able to transmit (or receive) data. In addition, the base station handles the RACH resources management by means of the `SetRachReservedSubChannels()` method. Specifically, if resources meet the configuration parameters, e.g., the periodicity, they are reserved for the RACH. Thus, end users are unable to exploit them to transmit regular data in the uplink, since only preambles can be transmitted. This control is performed during the uplink scheduling procedure by the method `isRachOpportunity()`. Finally, the constructors of the classes `GnbBaselineRandomAccess` and `GnbEnhancedRandomAccess` allow to control RACH resources occurrence, as well as the number of different preambles and the maximum number of retry attempts, among others.

2.9.3 Reference Test and Results

Conducted tests evaluated a reference mMTC scenario since a multitude of simultaneous transmissions brings to performance degradation in terms of preamble collisions during the random access procedure. It is then assumed a simple communication scheme, which is well suited to the low-cost requirements of MTC devices. The reference scenario is based on the `SingleCell` deployment, at the application layer the model is the `CBR`, while the mobility model is `ConstantPosition`. In particular, the activation time of the devices

may be chosen either to follow a beta distribution with parameters (3,4) over a time interval of 10 s, or uniform distribution over a time interval of 5 s, hence modeling different event-driven transmission bursts. The users are positioned with a uniform random distribution over the simulated cell. As for the RACH configuration, one of the most common configurations has been chosen as a reference (RACH opportunities every 5 ms and 54 different congestion-free preambles). For the enhanced approach, the number of RARs transmitted for each preamble is set to 2 and 4. It is important to note that in case of collision, the procedure fails for all the involved devices, and can be repeated a maximum of 3 times. The scenario is called `MMC1` and the syntax used to perform the test is as follows:

```
./5G air simulator MMC1 r nUe traf sched frStr maxD cbrI sync
type (seed)
```

where

- `r` is the cell radius
- `nUe` is the number of users in the cell (i.e., a given user density, depending on the value of `r`)
- `traf` is the application layer traffic
- `frStr` refers to the duplexing method
- `maxD` is the maximum tolerable delay of the transmissions
- `cbrT` is the time interval between two successive transmissions by the same user
- `sync` is used to determine whether users transmit synchronously or not
- `type` is the random access procedure strategy, 1 for the baseline and 2 for the enhanced procedure
- `seed` is an optional seed to initialize random quantities to different and reproducible values for each simulation run

The performance is evaluated with different parameter settings, as reported in Table 2.13.

Baseline results investigate the collision rate of the random access process. Figure 2.21 shows the collision rate for each approach of interest.

TABLE 2.13: Adopted Values for the Parameters of the Scenario

Parameter	Value
r	290 m
nUe	156000 (10^6 UE/km^2)
traf	CBR
frStr	FDD
maxD	10 s
cbrT	300 s
sync	"sync"
type	0, 1
seed	1-50

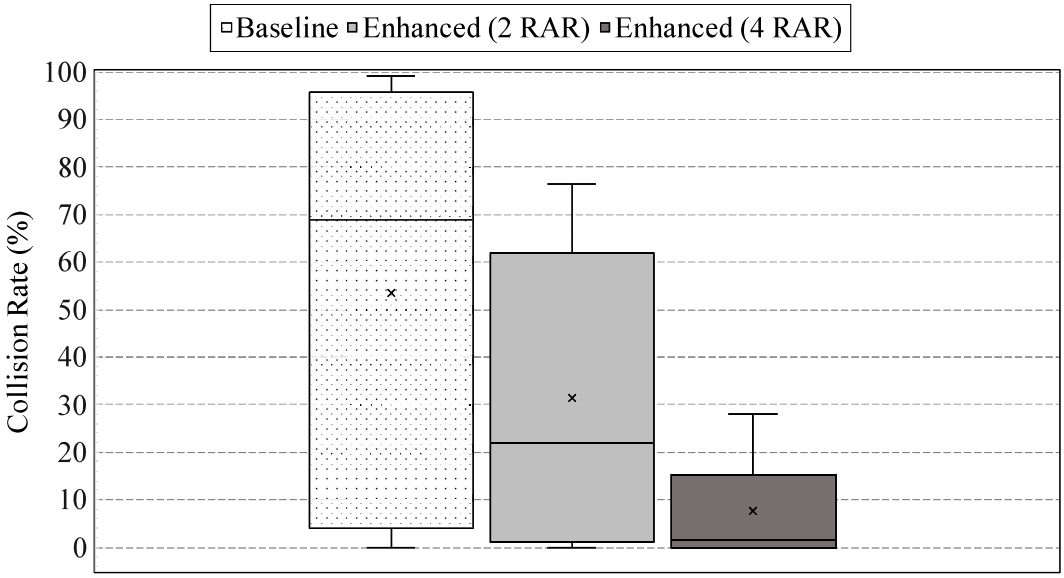


FIGURE 2.21: Comparison between the obtained preamble collision rates

The figure highlights the collision rates, their centroids (i.e., the small crosses), the 25^{th} and the 75^{th} percentile (i.e., the bottom line and the top line of the rectangles), as well as the minimum and the maximum measured value (i.e., the edges of the vertical lines). The most noticeable feature is the incredibly high collision rate of the baseline random access. In this case, almost no preambles can be detected, and the devices retry connection until they succeed or fail. In contrast, the 5G enhanced approach with 4 RARs experiences only 28% collision rate at most, proving the effectiveness of the enhanced approach.

Chapter 3

NarrowBand-Internet of Things: Modelling and Analysis

NB-IoT is emerging as one of the most promising licensed Low Power Wide Area Networks, addressing the needs of low data rate devices. It is a cellular radio access technology properly designed by the 3GPP to reuse the existing mobile infrastructure while enhancing the coverage and battery lifetime of devices massively deployed within a cell. Furthermore, it requires a small bandwidth to properly work and it can be efficiently implemented in already existing cellular technologies. Therefore, it is usually regarded as a promising standard to meet the requirements of the future 5G & Beyond development for the IoT. For these reasons, this Chapter first covers this promising radio interface in Section 3.1 and Section 3.2, with a specific focus on the Random Access Procedure (Section 3.3). Then, its implementation in 5G-air-simulator is described in Section 3.4, along with numerical simulation results conducted to analyze its performance (Section 3.5).

3.1 Introduction

The IoT phenomenon is broadening at an astoundingly fast rate [136], which promoted the birth of several novel services in different application domains, including, but not limited to, Industry 4.0 [137], Smart Cities [138], Intelligent Transportation Systems [139] , Precision Agriculture [140] , healthcare [141] and environmental monitoring [142]. Hence, an increasing number of constrained smart devices are currently joining the worldwide Internet, asking for suitable wireless communication technology, capable of offering both extremely low power consumption and support for densely populated deployments. In general, these low-powered devices individually require low data transfer rates as well, even though a densely populated MTC deployment

might become incredibly bandwidth-hungry. At the same time, devices generally require extremely low power, remaining idle for prolonged periods.

Besides, either the physical location of devices in such scenarios may not be reached by fixed networks (e.g., basements) or the apparatuses have to be arbitrarily deployed or moved anywhere [143]. Low Power Wide Area Networks (LPWANs) are an innovative communication pattern addressing the aforementioned requirements of emerging IoT applications [144]. Essentially, they complement cellular and short-range wireless technologies, offering exclusive features for low complexity devices. Nowadays there are a number of LPWANs, each employing different techniques to meet the Machine-to-Machine (M2M) requirements. In the unlicensed spectrums, LoRa [145] and SigFox [146] are the most common. However, the licensed spectrum is widely known to supply a higher degree of reliability and Quality of Service. Among the licensed LPWANs, NB-IoT has been recognized as a promising and effective technology offering wireless connectivity to smart devices in 5G & Beyond networks. Moreover, the adoption of NB-IoT is able to satisfy most of the requirements of all the possible application scenarios, by dynamically adapting to different use cases. Standardized by the 3GPP starting from Release 13 [147], [148], NB-IoT natively supports the transmission of marginal amounts of data, while requiring low-energy consumption and limited bandwidth usage [149]. Differently from LTE, NB-IoT has been designed to be extremely energy efficient and to manage conceivably tens of thousands of devices with a reduced complexity per cell, each of them sporadically transmitting a little amount of data [150], [151]. Indeed, in a typical NB-IoT deployment, a multitude of constrained devices are located within a single cell and the vast majority of messages are exchanged in the uplink direction [152].

3.2 NB-IoT Radio Interface

NB-IoT is a cellular radio access technology that requires a 180 kHz bandwidth for both downlink and uplink [148]. Three different operation modes are possible:

- **Stand-Alone** operation: an operator can replace one GSM carrier of 200 kHz with NB-IoT, leaving a guard interval of 10 kHz on both sides of the spectrum.

- **In-Band** operation: one or more NB-IoT carriers are deployed inside a larger LTE/NR channel.
- **Guard-Band** operation: allocation of one or more carriers within the guard-band of LTE/NR bandwidth to NB-IoT.

All three deployment scenarios are transparent to non-NB-IoT devices. As a consequence, devices that do not implement NB-IoT functionality simply do not see the NB-IoT channel inside the main cellular bandwidth or in the guard band. At the same time, legacy GSM devices will not see an NB-IoT carrier if used alongside 180 kHz GSM carriers. Such devices will just see noise where NB-IoT is active [153]. A single carrier can be configured according to the chosen operation mode, whereas multiple carriers can also be used in order to supply a higher bandwidth.

At the physical layer, Orthogonal Frequency Division Multiple Access (OFDMA) and Single-Carrier Frequency Division Multiple Access (SC-FDMA) are used for downlink and uplink transmissions, respectively. NB-IoT fully inherits from LTE in the downlink, i.e. NR numerology 0. The transmission scheme is based on conventional OFDM using normal Cyclic Prefix (CP). An end user operates in the Downlink using 7 consecutive symbols on 12 subcarriers (also referred to as *tones*) grouped into a RB, with a subcarrier spacing of 15 kHz [148]. The duration of slot, subframe, and frame is analogous to LTE, as well. Therefore, in a single NB-IoT carrier there is only 1 RB and only one user can receive data at a time using all 12 subcarriers for each subframe. As regards modulation, only QPSK is supported.

Instead, the uplink supports not only the standard 15 kHz spacing but also a subcarrier spacing of 3.75 kHz. The elementary NB-IoT radio resource, termed RU, is the smallest unit to map a transport block [154]. It is assigned to a single user only. Unlike the well-known RB of LTE, an RU is dynamically defined as shown in Table 3.1 [155].

TABLE 3.1: Uplink RUs in NB-IoT

Transmission Mode	Subcarriers	Subcarrier Spacing [kHz]	RU Duration [ms]
Single-Tone	1	3,75	32
	1	15	8
Multi-Tone	3	15	4
	6	15	2
	12	15	1

Specifically, the uplink supports two different configurations, that are *Single-Tone* and *Multi-Tone*. *Single-Tone* uses either 3.75 or 15 kHz subcarrier spacing

and each subcarrier represents an RU. As a consequence, a 180 kHz carrier is divided in either 48 or 12 RUs, respectively, thus resulting in two different RU length values, that are 32 and 8 ms. In the case of *Multi-Tone* configuration, the subcarrier spacing is set to 15 kHz only. Nevertheless, a number of 3, 6, or 12 adjacent subcarriers may shape a single RU. Again, depending on the number of tones per RU, its length changes accordingly.

The modulation available in the uplink are restricted to BPSK and QPSK.

One of the main objectives defined in NB-IoT study item description [156] is to achieve 20 dB coverage extension compared with legacy GPRS while limiting the device maximum transmit power to 23 dBm (i.e., 200 mW), which is a factor of ten lower than the maximum output power of GPRS devices. Coverage extension is achieved by increasing the number of repetitions at the expense of higher data rates. Moreover, NB-IoT supports up to three coverage classes, in order to serve devices experiencing different ranges of path loss. In particular:

- *Normal* is similar to legacy GPRS coverage;
- *Extended* corresponds to about 10 dB improvement with respect to legacy GPRS;
- *Extreme* achieves 20 dB extension compared to legacy GPRS.

The number of coverage classes is configurable by System Information. Let N_{CC} be the number of available coverage classes so that $1 \leq N_{CC} \leq 3$. Different coverage classes correspond to operation with different modulation orders, coding rates, repetition factors, and subcarrier spacings, in order to match the data rate for each user to its available link budget. This allows devices having good coverage to operate at higher data rates and with lower latency than devices that have poor coverage. Therefore, the system is designed to meet the throughput and latency requirements for devices in *Extreme* coverage, while devices in *Normal* or *Extended* coverage achieve improved performance.

NB-IoT generally takes advantage of the existing LTE physical channels, revised properly to fit into the narrower bandwidth [157]. In the downlink direction, there are three channels:

- Narrowband Physical Downlink Control Channel (NPDCCH) carries scheduling assignments, as well as HARQ acknowledgments, paging indication, and system information update. It is considered as the core

element of the scheduling procedure as it carries Downlink Control Informations (DCIs), which contain uplink scheduling grants, downlink scheduling assignments, and the type of modulation.

- Narrowband Physical Downlink Shared Channel (NPDSCH) carries data from the higher layers as well as paging message, system information messages, and the RAR message. It is scheduled by the NPDCCH but is transmitted after a certain time delay (indicated in the NPDCCH too) to allow the low-complexity NB-IoT devices enough time to decode the NPDCCH.

- Narrowband Physical Broadcast Channel (NPBCH) is transmitted in the first subframe and carries the Master Information Block - Narrow-Band (MIB-NB) over 8 consecutive radio frames, repeated 8 times to cope with extreme coverage conditions; the MIB-NB content remains, therefore, unchanged for 640 ms.

On the other hand, for the uplink direction, there are two different channels, specifically:

- Narrowband Physical Uplink Shared Channel (NPUSCH) has two formats. Format 1 is used for carrying uplink data and uses the same LTE error correction code. Format 2 is used for signaling HARQ acknowledgement for NPDSCH and uses a repetition code for error correction.

- NPRACH enables the random access procedure and it has been completely redesigned [158]. It will be described in detail in the following Section.

As regards synchronization signals, Narrowband Primary Synchronization Signal (NPSS) and Narrowband Secondary Synchronization Signal (NSSS) are used by an NB-IoT device to perform cell search, which includes time and frequency synchronization and cell identity detection.

3.3 Random Access Procedure: Description and Model

The random access procedure represents a key interaction between end devices and the base station. The NPRACH only uses the *Single-Tone* configuration and 3.75 kHz subcarrier spacing. In essence, it allows users to send

random access preambles. The preamble is mainly characterized by four symbol groups, each of them comprising five OFDM symbols plus the cyclic prefix. A pseudo-random frequency hopping algorithm offers as different preambles as the number of subcarriers allocated to the NPRACH. As a consequence, selecting different subcarriers at the beginning of the transmission ensures hopping schemes that never overlap. In addition, different NPRACH resource configurations can be deployed in a cell, each corresponding to a different coverage class. Each one is described through periodicity, the number of repetitions, starting time, frequency location, and number of subcarriers, as depicted in Figure 3.1.

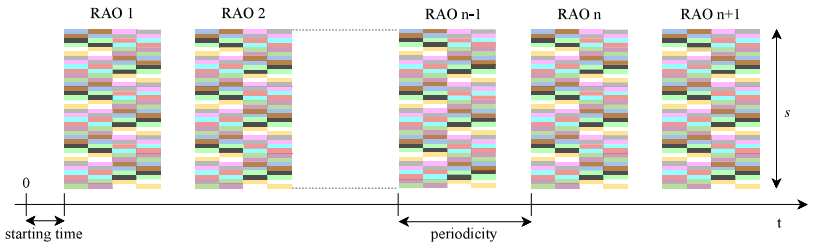


FIGURE 3.1: RAOs timing diagram.

Generally, an NPRACH resource is called RAO.

NB-IoT random access procedure is per se contention based [159] and entails the exchange of four messages, as depicted in Figure 3.2:

1. The device finds the first available RAO and then transmits a random preamble, chosen among the available ones. Let s be the number of the available subcarriers. Then user starts the RAR window, W.

2. Upon preamble reception, the base station transmits an RAR, that explicitly instructs the user on which uplink resources have to be utilized for the transmission of the next message. If this message is not received, the user keeps waiting for it until the expiration of W.

3. Exploiting the scheduled resources, the user transmits its identity and other important information; this message is also known as Msg3. Subsequently, the user starts the Contention Resolution Timer.

4. The base station performs the contention resolution and sends back to the devices the identity of the winning users through the Contention Resolution Message. If this message is not received, users keep waiting for it up to the Contention Resolution Timer expiration.

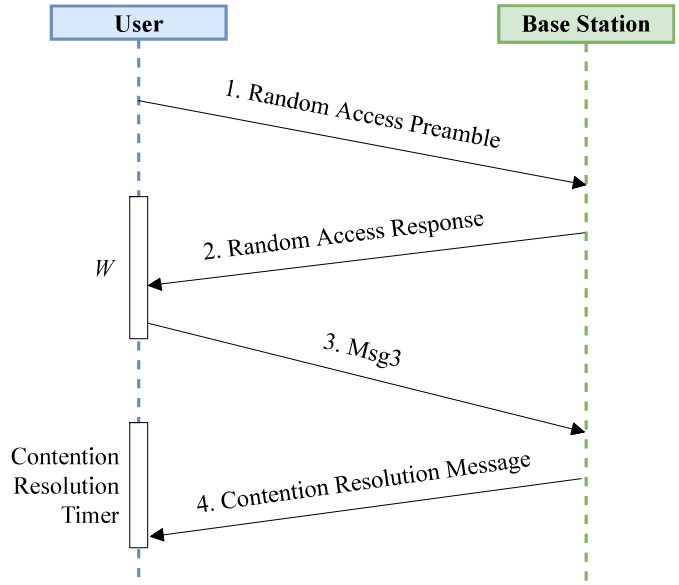


FIGURE 3.2: Random Access Procedure sequence diagram.

The procedure fails when either the RAR or the Contention Resolution Message is not correctly received by mobile terminals in the proper windows. Henceforward, a worst-case scenario will be assumed. Particularly, overlapping preambles always destructively interfere, therefore the base station never detects them. In other words, if two or more users send the same preamble in a single RAO, each user fails the procedure.

Each collided user may retransmit a preamble after a backoff time b, chosen uniformly and at random within the interval $[0, B]$. The maximum number of preamble transmissions that a single mobile terminal can attempt in a general coverage class c is set to a_c. If a user fails the a_c-th attempt, it considers being in the next higher coverage class if exists. This behavior is repeated until the maximum number of transmissions that can be tried globally in all classes, i.e., , is reached.

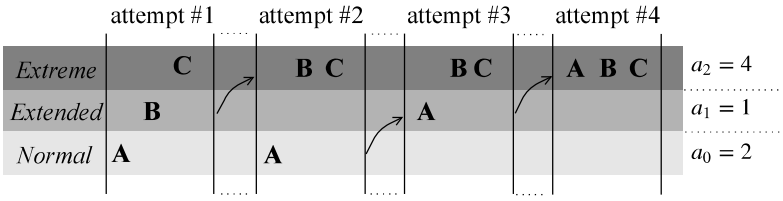


FIGURE 3.3: Coverage class hopping of 3 distinct users during random access procedure with = 4.

As a case in point, Figure 3.3 reports three different users, i.e., A, B, and C, starting the first transmission attempt in the *Normal*, *Extended* and *Extreme*

class, respectively, based on the assumption that they always collide with other users. Since $a_0 = 2$, as soon as user A fails the second attempt, it retries the subsequent transmission in the *Extended* coverage class. Here, both A and B can try up to $a_1 = 1$ preamble transmission, hence jumping to the *Extreme* class in the immediately following attempt. On the other hand, user C, whose class is the *Extreme*, is forced to persist in its class. Anyway, even though $a_2 = 3$, a total amount of $= 4$ attempts are foreseen for the C user as well.

3.3.1 A Preliminary Analytical Model

Since most of the NB-IoT transmissions occur in the uplink, the NPRACH may usually become the main bottleneck of the entire system. For this reason, analytical models and simulation tools able to investigate its behavior in different scenarios are of the utmost importance for driving current and future research activities. Unfortunately, scientific literature partially addresses the current open issues by means of simplified and, in many cases, not standard-compliant approaches.

In order to provide a significant step forward in this direction, this Section formulates an analytical model capturing both collision and success probabilities associated with the aforementioned procedure.

The collision probability and the success probability are analytically derived below, as a function of the actual number of users accessing an RAO, N, and the number of available NPRACH subcarriers, s. Let the *Collision Probability*, $\mathcal{P}_{N,s}$, be the probability that, in a given RAO, a preamble collision happens.

First of all, if no users access the channel, no collisions will occur. Instead, the probability P_k that k among N users choose the same subcarrier follows a binomial distribution. For this reason, $P_k = \binom{N}{k} p^k (1-p)^{N-k}$, where p is the probability of choosing a specific subcarrier. According to NB-IoT specification, each subcarrier is selected with the same probability, hence $p = \frac{1}{s}$. If a preamble is chosen only once, that is $k = 1$, the probability that only one of the N users chooses the specific subcarrier is equal to:

$$P_1 = N \frac{1}{s} \left(1 - \frac{1}{s}\right)^{N-1}. \tag{3.1}$$

Let $\widehat{N}$ be the number of users not colliding. It is indeed equal to the average

number of preambles chosen only once. In fact, given the (3.1), $\widehat{N}$ is computed as the product of P_1 and the number of different preambles, s:

$$\widehat{N} = sP_1 = N\left(1 - \frac{1}{s}\right)^{N-1}. \tag{3.2}$$

As a result, the average number of collided users, that is $\widetilde{N}$, corresponds to the total amount of users, i.e., N, except the users whose preambles not collide, i.e., $\widehat{N}$, that is:

$$\widetilde{N} = N - \widehat{N} = N\left[1 - \left(1 - \frac{1}{s}\right)^{N-1}\right]. \tag{3.3}$$

In conclusion, the Collision Probability can be expressed as follows:

$$\mathcal{P}_{N,s} = \frac{\widetilde{N}}{N} = 1 - \left(1 - \frac{1}{s}\right)^{N-1}, \qquad N > 0. \tag{3.4}$$

Similarly, let the *Success Probability* be the probability that, in a given RAO, a user successfully completes a preamble transmission. As previously stated, $\widehat{N}$ is the number of users not colliding. Since $\widehat{N} = N(1 - 1/s)^{N-1}$, as the (3.2) reports, thus $(1 - 1/s)^{N-1}$ clearly represents this success probability. The other way around, the success probability $\mathcal{S}_{N,s}$ can be intuitively modeled as:

$$\mathcal{S}_{N,s} = 1 - \mathcal{P}_{N,s} = \left(1 - \frac{1}{s}\right)^{N-1}, \qquad N > 0. \tag{3.5}$$

Accordingly, it is worth noting that $\mathcal{S}_{1,s} = 1, \forall s > 0$, as if a single user accesses an RAO, it will not certainly collide.

3.4 NB-IoT in 5G-air-simulator

5G-air-simulator provides the support for a variety of NB-IoT features. As already mentioned, in a mMTC context, when a massive number of devices typically sends data, the uplink direction generally deserves more attention than the downlink. Accordingly, NB-IoT main novelties focus on the uplink direction, where the channel is more congested. For the same reason, the 5G-air-simulator implements only the NB-IoT uplink side. Figure 3.4 shows a high-level illustration of the main features implemented in the simulator, whereas Table 3.2 summarizes the related code.

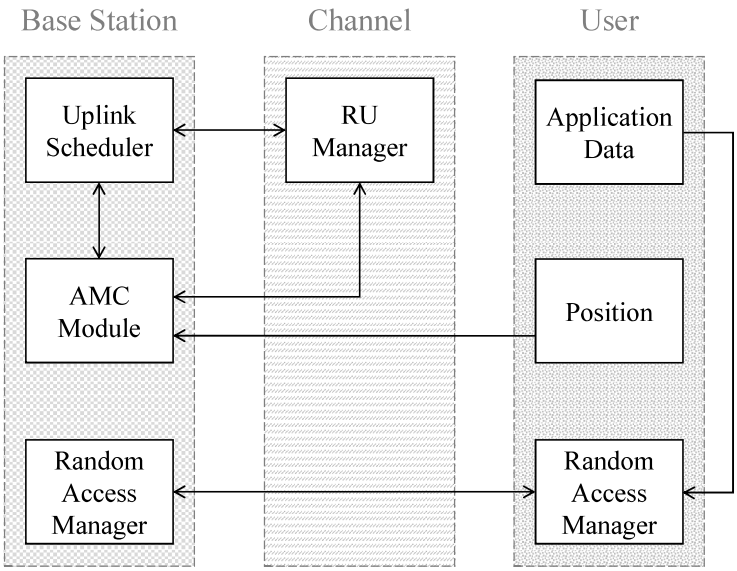


FIGURE 3.4: Block diagram of main NB-IoT features implemented in 5G-air-simulator.

TABLE 3.2: 5G-air-simulator methods related to NB-IoT features.

Key Functionality	**Class**	**Method**	**Parameters**
Generate NB-IoT carriers and sub-channels	BandwidthManager	CreateNbIoTspectrum()	Number of carriers, subcarrier spacing, number of tones
Determine the maximum number of carriers available for NB-IoT depending on the chosen reference bandwidth.	BandwidthManager	GetMaxNbIoTcarriers()	(none)
Get the usable RBs for NB-IoT implementation.	BandwidthManager	GetNbIoTrb()	(none)
Apply the correct RU duration.	FrameManager	setTTILength()	number of tones, subcarrier spacing
Get the proper number of RUs needed to transmit a packet.	nb-AMCModule	GetNbOfRUsFromSize()	MCS index, packet size
Get the TBS related to a specifc MCS index and number of RUs.	nb-AMCModule	GetTBSizeFromMCS()	mcs index, RU index
Get the proper number of RUs needed to transmit a packet.	nb-AMCModule	GetMCSfromDistance()	Distance from base station, cell radius, number of tones
Start the uplink scheduling procedure for NB-IoT devices.	nbUplinkPacketScheduler	DoSchedule()	(none)
Perform the RU assignment for scheduled users following the Round Robin algorithm.	nbRoundRobinUplinkPacketScheduler	RUsAllocation()	(none)
Perform the RU assignment for scheduled users according to a FIFO strategy.	nbFifoUplinkPacketScheduler	RUsAllocation()	(none)

Essentially, the classes `BandwithManager`, `nb-AMCModule` and `nbUplinkPacketScheduler` provide the support for NB-IoT features. `BandwithManager` deals with multiple carriers, subcarrier spacings, *Single-Tone* and *Multi-Tone* transmissions; the latter can be configured for using either 3, 6 or 12 tones. The AMC module, namely `nbAMCmodule`, picks the appropriate TBS, starting from the selected MCS index and number of RUs given by the chosen scheduling strategy. The flexible structure of an uplink RU greatly complicates radio resource management. As a consequence, MAC schedulers should carefully consider the different RU lengths when performing resource assignments. The length of a RU can be set via the *FrameManager::setTTIlength()* method. Essentially, it gives the correct length of the TTI, based on the chosen numerology.

According to what has been previously stated, the scheduling paradigm is redesigned. To this purpose, the abstract class `nbUplinkPacketScheduler` provides inherent methods of all the scheduling strategies. Then, two different scheduler classes have been developed, related to two well-known strategies: First-In First-Out (FIFO) and Round-Robin (RR). It is important to observe that additional scheduling algorithms can also be developed following the same rationale.

3.4.1 Random Access Procedure

At the code level, the `EnbNbIoTRandomAccess` and `UeNbIoTRandomAccess` classes handle the random access procedure from the base station and user point of view, respectively. These classes have been deeply revised since they are not capable of addressing the multi-class attempts control and transitions between coverage classes (as discussed in Section 3.3).

As for the `UeNbIoTRandomAccess`, there are several attributes and methods enabling coverage class transitions and attempts-checking. In particular, the `UeNbIoTRandomAccess::SendMessage1()` method selects the appropriate NPRACH resources for transmitting preambles in the correct coverage class. In addition, in the `UeNbIoTRandomAccess::ReStartRaProcedure()` method there is an attribute that stores the number of failed attempts for the current coverage class, in order to manage the transitions between coverage classes.

On the other hand, the `EnbNbIoTRandomAccess` class handles the remaining random access procedure features. First of all, the base station configures NPRACH resources for each coverage class c, according to the chosen configuration parameters. Specifically, the

base station actually allocates resources to the NPRACH by periodically calling the `EnbNbIoTRandomAccess::SetRachReservedSubChannels()` method. It is worth mentioning that other configuration parameters, e.g., B and W, are handled by the base station as well. The `EnbNbIoTRandomAccess::CheckCollisions()` method is of paramount importance. Indeed, for each given RAO of all the coverage classes, it checks whether any collisions happened (i.e., 2 or more users selected the same preamble) and it schedules new preamble transmission after a random backoff time b. Then, as explained in Algorithm 1, both attempt counters are incremented and subsequently checked with the chosen values of a_c and , in order to determine a coverage class transition or the procedure failure. Specifically, if the user has any overall attempt left but no more attempts for its class, it switches to the higher class and the number of its attempts for its class is set to zero. If no higher class is available, i.e., the user is already in the *Extreme* class, the user goes on until it has overall attempts left. Next, a new preamble transmission is solicited. If the counter of the overall attempts hits its maximum, the procedure fails and both attempt counters are set to zero. On the contrary, the exchange of the next messages is conducted until the end of the procedure for the users not experiencing a collision.

A simulation run may provide three additional output variables, that are the number of users accessing an RAO, N, as well as the collision and success

probabilities, denoted by $\overline{\mathcal{P}_{N,s}}$ and $\overline{\mathcal{S}_{N,s}}$, respectively.

Algorithm 1: The implemented random access procedure.

Require: $i, j = 1$
get c, N_{CC}, s, W, , a_c, B
randomly select one of s preambles ; wait next RAO; send the preamble
if preamble collided **then**
  $i = i + 1$; $j = j + 1$
  draw random b from $[0, B]$
  wait W
  if $j \leq$ **then**
    if $c = N_{CC}$ **then**
      wait b and go to 2
    else
      if $i \leq a_c$ **then**
        wait b and go to 2
      else
        $i = 1$; $c = c + 1$
        wait b and go to 2
      end if
    end if
  else
    procedure FAIL
  end if
else
  exchange RAR
  exchange Msg3
  exchange Contention Resolution Message
end if

3.4.2 Satellite NB-IoT

A number of recent studies specifically considered NB-IoT as a promising technology for 5G satellite MTC [160]–[166].

However, the scientific literature is mainly focusing attention on physical and link-level analysis only. To overcome this issue, the availability of a system-level simulator to support the research activities represent a mandatory requirement. The majority of the aforementioned papers employ link-level simulators and focus the attention on a single communication link. At the same time, recent works suggest that there is a growing demand for flexible tools for designing and testing new algorithms and protocols for NB-IoT-based satellite scenarios.

In this context, 5G-air-simulator appears as a solid instrument to support the NB-IoT technology in a satellite scenario. Moreover, it is important to

emphasize that there exists some preliminary research work already using the baseline version of 5G-air-simulator [167], [168], confirming that the simulation tool has recently gained currency also in SatCom.

In the following, the main features of the proposed tool will be described, including L2S model, management of blind repetitions and their impact on BLER curves, cell selection, and mobility models for a constellation of satellites. On the one hand, the proposed L2S model perfectly integrates channel and communication models widely accepted in the current state of the art. On the other hand, the rest of implemented features offers the opportunity to test new (because beyond the current studies presented in the literature) SATCOM configurations, while satisfying the reference 3GPP specifications.

Figure 3.5 shows a general overview of the implemented module and remarks the interaction between different building blocks, presented below.

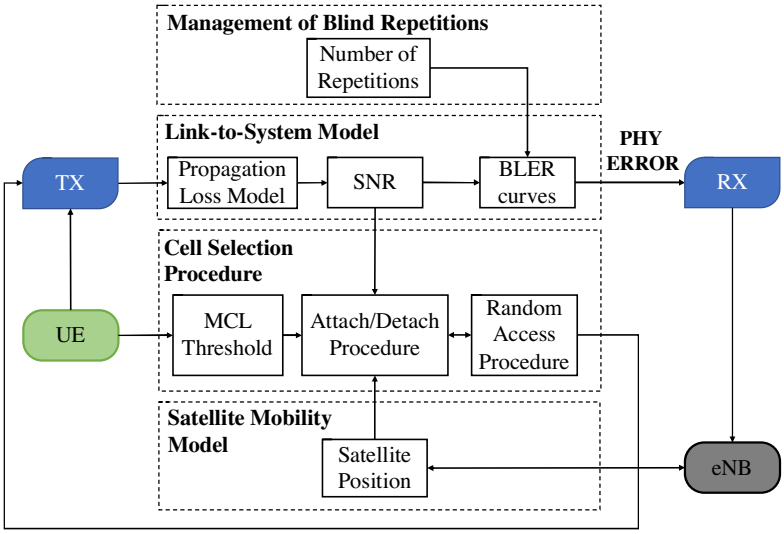
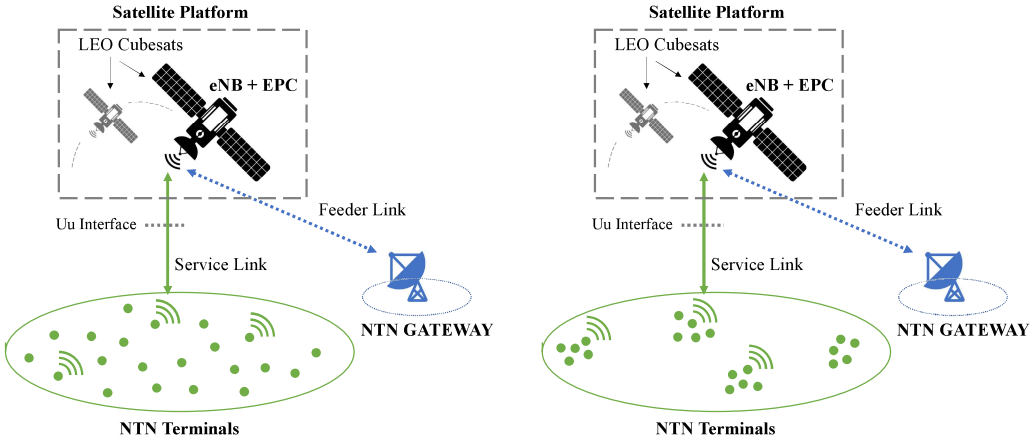


FIGURE 3.5: Overall vision of the interaction between the implemented simulator features.



(A) Uniform Distribution

(B) Cluster Distribution

FIGURE 3.6: The reference network architecture.

3.4.2.1 Initial assumption on the Architecture

Figure 3.6 depicts the NB-IoT Satellite-Based architecture considered in this work. The figure points out two possible configurations regarding the distribution of the device on the ground. A configurable number of fixed NTN terminals may be positioned randomly on the simulated area, either following a uniform distribution or in many smaller clusters. This latter case allows simulating different use cases where NTN terminals are tightly deployed only in specific areas.

Coverage and traffic profiles are other important aspects that have a major impact on the choice of an effective satellite network architecture. First of all, it is critical to allow NTN terminals to transmit data when needed, preventing the congestion of transmission buffers as much as possible. For this reason, a satellite should have a relatively low orbital time to avoid users remaining for too long without service. As expected, satellite communications present serious propagation impairments, due to the long distance. In line with 3GPP guidelines [169], this work assumes to use LEO satellites for guaranteeing feasible communication links with satisfactory levels of Signal to Noise Ratios (SNRs).

However, a single LEO satellite may not be able to run across its entire orbit at the aforementioned rate. Therefore, it is necessary to consider several satellites per orbit, forming a constellation, to drastically reduce the time periods during which ground-based devices remain without satellite coverage [170]. In this context, the Cubesats [171] are a solution that provides low

costs and several simplifications in the system deployments for the satellite constellation. These small satellites composing the Satellite Platform can be assembled in a fully scalable and flexible way in order to address the required performance, while keeping device costs low. In this regard, the tool allows the possibility to configure the number of Cubesats per orbit to ensure greater deployment flexibility.

Each NTN terminal is a 3GPP NB-IoT User Equipment (UE) able to use a direct satellite access, thanks to an adapted Uu interface. The NB-IoT technology is used to implement the service link, established between the NTN terminal and the remote satellite.

The feeder link is the radio link between the NTN Gateways and the Satellite Platform. It is preferable to have a limited number of gateways in order to dramatically reduce system costs, even though this leads to relatively long periods of time of service unavailability, i.e., satellites can only offload their data when the feeder link is active. However, this is not a problem since the targeted scenarios generally consider delay-tolerant applications.

According to this architecture, every satellite of the LEO constellation implements a Base Station. As a consequence, NTN terminals are expected to perform again the network attachment procedure each time they are covered by a different satellite.

It is important to stress that during the creation of the NTN terminals in the configured scenario only uplink channels are considered, i.e., the downlink transmission is not modeled. Moreover, only Single-Tone transmissions are taken into account, in order to both achieve better performance due to the increased robustness over the service link and further exploit NB-IoT capabilities to manage a multitude of users thanks to its wise bandwidth management.

3.4.2.2 Management of Blind Repetitions

The first extension introduced in the 5G-air-simulator is the handling of the blind repetitions. It provides the transmission in a bundle of the same Transport Block, replicated for a specified number of times. This key feature enables communication even at low SNR values. Indeed, it is crucial to maximize both the visibility time and the total throughput.

The number of total blind repetitions can be set for the NPUSCH transmissions via the `FrameManager::SetNRep` method. Then, this value is retrieved when performing the scheduling procedure. Specifically, now the methods `RUsAllocation` of both the implemented scheduler classes (i.e.,

FIFO and Round Robin) take into account this parameter while assigning RUs to users and finalizing the scheduling procedure. In this way, the reception event happens after the correct amount of time, which depends on the number of repetitions and the actual slot duration, as well as on the number of RUs allocated to the UE.

It is important to highlight that also the L2S model uses the adopted number of repetitions to estimate a suitable BLER, as explained in what follows.

3.4.2.3 Link-to-System Model

In the context of non-terrestrial NB-IoT networks, one of the most important factors which determine architectural decisions is represented by the satellite link performance, which allows supporting direct connectivity between NTN terminals and satellite. Therefore the L2S model is of fundamental relevance since it offers a simplified (but still accurate) abstraction of transmission, propagation, and reception functionalities. It associates a link-level analysis to the system-level simulation tool. Indeed, in a system-level simulator, it is reasonable to provide a simplified channel model. Otherwise, it would result in excessive complexity and execution time. This model contains the SNR expressions for both downlink and uplink channels, and the BLER curves for each transmission mode.

The 5G-air-simulator did not originally model the radio channel for NB-IoT. Thus, a new propagation loss model is developed to evaluate the signal received by satellite, considering the non-idealities of the channel in the satellite scenario.

Specifically, the SNR is analytically modeled by taking into account the power gains and losses due to the propagation over the radio channel. Given the elevation angle of the service link, i.e., θ_{el}, and the carrier frequency, f_c, the SNR SNR quantifying the link performance, evaluated in dB, can be modeled as follows [172]:

$$\begin{aligned} SNR(\theta_{el}, f_c) = P + G_{ANT}(\theta_{el}, f_c) \quad PL(\theta_{el}, f_c) \\ L_{imp}(\theta_{el}, f_c) + DCF(\theta_{el}, f_c) \quad N, \end{aligned} \tag{3.6}$$

where P represents the signal transmission power and G_{ANT} represents the sum of the antenna gains of satellite and NTN terminal (in dBi). PL is the free space path loss that accounts for the radiowave attenuation due to propagation, and L_{imp} represents additional losses due to all the impairments considered, such as:

- the air attenuation, which accounts for the dry air absorption [173];
- the fog attenuation, which predicts the attenuation due to clouds and fog on Earth-space paths [174];
- the attenuation due to atmospheric gas absorption, which estimates of gaseous attenuation [173], [175];
- the attenuation due to droplets and rainfall, which estimates the slant-path rain attenuation as described in [176];
- the polarization attenuation that accounts for the difference between the polarization of the receiving antenna and the polarization of the incoming (incident) wave [172];
- the attenuation due to scintillation, which accounts for small time-scale fluctuations (on the order of fractions of a second) of the amplitude and the phase of a radio wave [175].

In addition, *DCF* is the sum, in dB, of the diagram correction factors of transmitting and receiving antennas. These factors take into account the mismatch between theoretical and real antenna radiation diagrams, as a function of the elevation angle and the carrier frequency.

Finally, the noise power N can be evaluated by taking into account the system noise power at the receiving antenna, which is a function of the equivalent noise temperature at the satellite and the noise figure of the amplifiers at the receiver front end (for a detailed computation of the system noise power please refer to [175]).

The `PropagationLossModel::AddLossModel` method is properly extended to obtain the power value of the received signal by using the model described above.

For this purpose, a new header file is defined containing the results of the link-level analysis, such as the received power from the satellite at NTN terminal side and the received power from the NTN terminal at satellite side at different elevation angles, as well as the BLER curves for each transmission mode.

The new method `BLERvsSINR_NBIoT_SAT::GetRxPowerfromElAngle_SAT` evaluates the received power at the satellite side for each value of the elevation angle experienced by the NTN terminal. As a consequence, during the reception, the satellite retrieves an SNR value related to the uplink configuration used for the transmission, which reflects the quality of the channel.

In essence, this SNR value is exploited to estimate the BLER for the received block using new SNR-BLER curves, which determines the probability that it has been correctly received. Specifically, this operation is performed in the method `NBIoTSimpleErrorModel::CheckForPhysical Error`.

To this end, the BLER is estimated by considering the chosen MCS, the number of used RUs, the number of NPUSCH blind repetitions, and the SNR experienced at the satellite during the reception. The BLER value is drawn by `BLERvsSINR_NBIoT_SAT::GetBLER_SAT`, using SNR-BLER curves stored into the header file and generated using the MATLAB LTE Toolbox [177].

As a representative example, Figure 3.7 depicts the BLER curves for a fixed TBS and four blind repetitions.

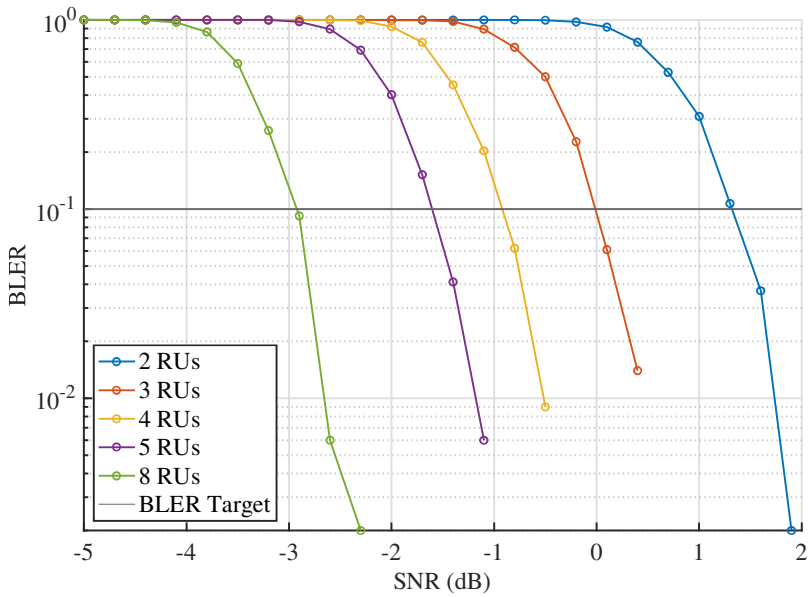


FIGURE 3.7: Example BLER curves for TBS of 256 bits and blind repetitions set to 4.

3.4.2.4 Satellite Mobility Model

A new mobility model, i.e., `SatelliteMovement` has been defined. It manages the movement of the satellites by defining their coordinates in the selected scenario. Figure 3.8 provides a general overview on the implemented mobility model. Specifically, `SatelliteMovement::GetSatPosition` tracks the position of the satellite.

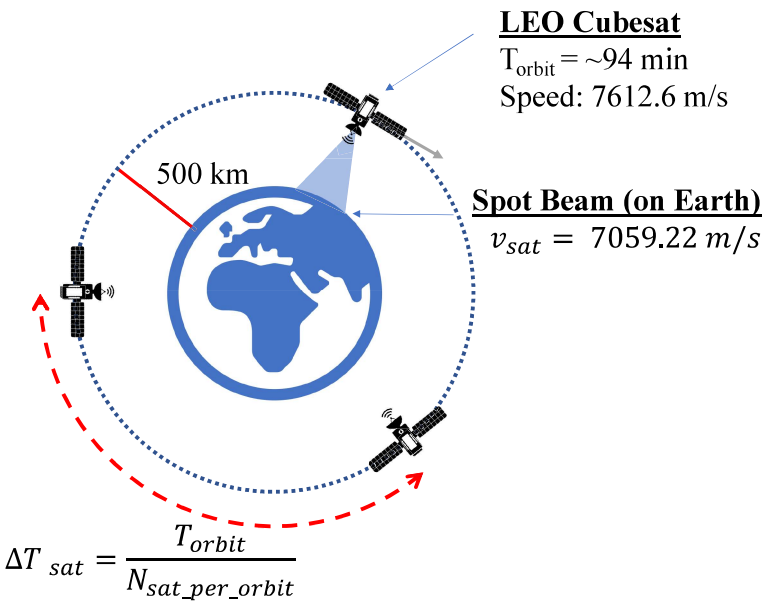


FIGURE 3.8: Key parameters of the implemented mobility model.

For the purposes of the simulation, and without loss of generality, satellites movement was considered exclusively in one direction on a reference axis of the Cartesian plane, i.e., the x-axis. The point value of the considered position refers to the center of the beam that covers the area on the ground. Based on the number of satellites in the orbit and the time instant, this method provides the updated value of the position according to the following equation:

$$x_{Sat}(t) = x_{0,Sat} + v_{sat}(t \mod \Delta T_{sat}), \tag{3.7}$$

where $x_{0,Sat}$ corresponds to the initial position of the satellite, v_{sat} represents the relative speed of the satellite spot beam on the Earth, t represents the time instant considered and the modulo operation is needed to exploit the periodicity of the position function. Finally, ΔT_{sat} represents the elapsed time between two different satellites. It is given by T_{orbit}, that is the time taken by the satellite to make one complete revolution around the Earth, e.g., about 94 minutes, over $N_{sat\_per\_orbit}$, that is the number of the satellites in a single orbit. ΔT_{sat} may be expressed as:

$$\Delta T_{sat} = \frac{T_{orbit}}{N_{sat\_per\_orbit}}. \tag{3.8}$$

3.4.2.5 Cell Selection Procedure

The position of the satellites is useful for determining whether the entities involved in the communication, i.e., NTN terminals and the satellite, are actually in reciprocal visibility and therefore able to communicate or not. For this purpose, a new extension is introduced. This computation is performed within the `UserEquipment::UpdateUserPosition` method.

First, NTN terminals not having an empty transmission buffer measure the power of the downlink signal received from the satellite. To this end, an essential parameter to determine the maximum coverage the cellular system can support is defined by the Maximum Coupling Loss (MCL) and expressed as follows:

$$MCL[dB] = P_{TX} \quad P_{RX}. \tag{3.9}$$

P_{TX} is the transmitted signal power by the satellite and P_{RX} represents the received signal power on NTN terminal side. The former is equal to 33 dBm of the parabolic reflector antenna used for the simulation. The latter is estimated by the L2S model, specifically from the link model for downlink channel, starting from the elevation angle. It can be evaluated since the simulator knows both NTN terminal and satellite positions.

Once the MCL goes under a defined threshold, i.e. 164 dB, the NTN terminal starts the attach procedure to the satellite. For evaluation purposes, the MATLAB LTE Toolbox was used to estimate if the NTN terminal can retrieve the correct Cell Information from the downlink signal at different SNR values.

Figure 3.9 depicts the success probability of the cell selection procedure. The NTN terminal can start the Random Access Procedure after the attach procedure is successfully completed.

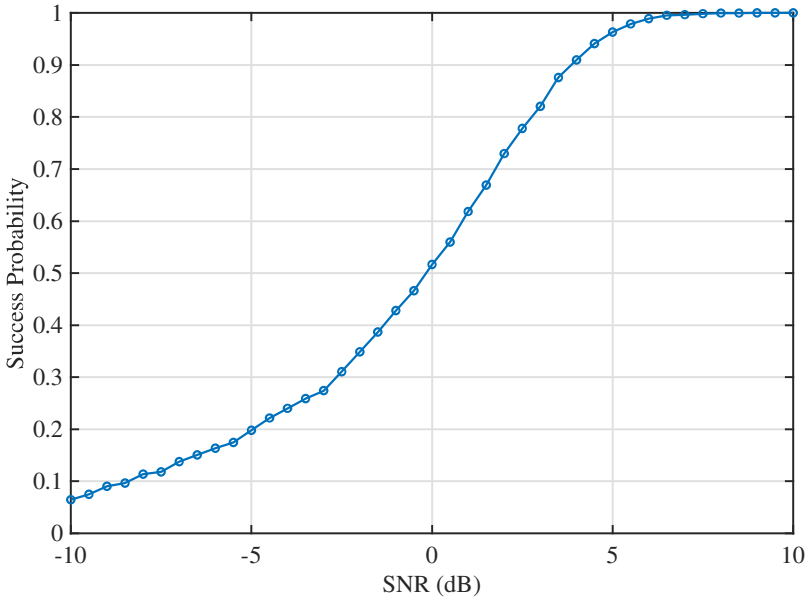


FIGURE 3.9: Cell Selection success probability at different SNR values.

The NTN terminal continuously monitors the downlink power signal in order to maintain the connection with the satellite. Thanks to this approach, the simulator can model an error during the Msg2 and Msg4 reception of the Random Access Procedure. If so, the procedure has to be rescheduled again. On the other hand, the NTN terminal may fail the attach procedure even if it accomplishes the Random Access Procedure, breaking the possibility to communicate.

3.5 Performance Evaluation

A first evaluation has been conducted using 5G-air-simulator to fulfill a preliminary performance assessment of the baseline NB-IoT technical component. The reference scenario is based on the `SingleCell` deployment, at the application layer the model is the `CBR`, while the mobility model is `ConstantPosition`. The users are positioned with a uniform random distribution over the simulated cell. The scenario is called `nbCell` and the command-line syntax to use it is:

```
./5G air simulator nbCell sched time r nUe bw nC spa tones
cbrT cbrS aMax nCl [p0] [pAtt] [pRep] [rWind] [nP] [period]
[o] [boW] (seed)
```

where

- `sched` is the scheduling algorithm;

- `time` is the duration in seconds of each simulation run;
- `r` is the cell radius;
- `nUe` is the number of users in the cell;
- `bw` is the total bandwidth used by the base station;
- `nC` is the number of NB-IoT carriers;
- `spa` is the subcarrier spacing;
- `tones` is the number of tones;
- `cbrT` is the time interval between two successive transmission by the same user;
- `cbrS` is the size of the data sent by the users at each transmission;
- `aMax` is the maximum number of retry attempts for the random access procedure;
- `nCl` is the number of the coverage classes;
- `[p]` is the probability that users belong to the coverage classes;
- `[pA]` is the number of preamble transmission attempts;
- `[pRep]` is the number of preamble repetition;
- `[rWind]` is the duration of the RAR window;
- `[nP]` is the number of different RACH preambles;
- `[period]` is the periodicity of RACH resources;
- `[o]` is the starting time of RACH resources;
- `[boW]` is the duration of the RACH backoff window;

Conducted tests demonstrated the impact of the average number of transmission requests per second, λ, on a reference NB-IoT network, with the parameter settings shown in Table 3.3. Simulations consider a plausible IoT scenario, where devices generate at the application layer packets of either 128 or 256 Bytes every 60 s.

TABLE 3.3: Adopted Values for the Parameters of the NB-IoT Scenario

Parameter	Value
sched	FIFO, Round-Robin
time	150 s
r	1000 m
nUe	1200, 2400, 3600 users
bw	5 MHz
nC	1 carrier
spa	3.75, 15 kHz
tones	1, 3, 12
cbrT	60 s
cbrS	128, 256 Bytes
aMax	4 Attempts
nCl	1 coverage class
[p]	1 (100%)
[pA]	3 attempts
[pRep]	1 repetition
[rWind]	12 ms
[nP]	48 preambles
[period]	320 ms
[o]	8 ms
[boW]	256 ms
seed	1-50

Since transmission requests follow a discrete uniform distribution in the interval $[1, \texttt{cbrT}]$, every cbrT seconds there are, on average, $\lambda \cdot \texttt{cbrT}$ different mobile terminals that want to transmit a packet of size cbrS. In other words, $\lambda = \texttt{nUe}/\texttt{cbrT}$. System performance with both Single-Tone and Multi-Tone transmissions has been evaluated when both FIFO and RR scheduling algorithms are taken into account. System performance was evaluated in terms of the average number of users managed by the packet scheduler, average system goodput, and E2E packet delays. In order to increase the level of confidence of reported results, each simulation has been repeated 50 times.

The average number of users that are managed by the scheduler when the packet size cbrS is set to 128 Bytes and 256 Bytes is reported in Table 3.4 and Table 3.5, respectively. When cbrS = 128 Bytes, the average number of users that wait to transmit data over the radio interface is generally lower

TABLE 3.4: Average number of scheduled users, `cbrS` = 128 Bytes

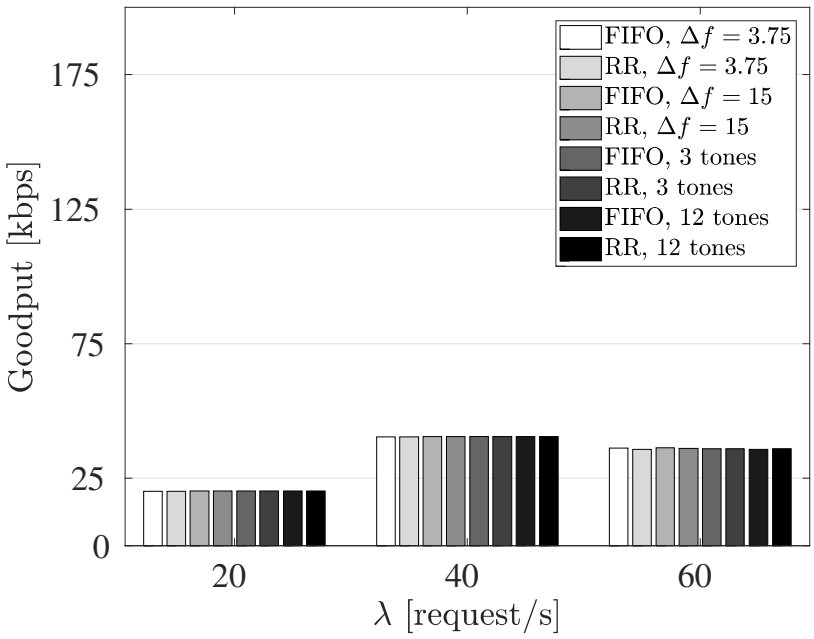
Scheduling Algorithm	**Number of Tones**	λ			Δf
		20	40	60	
FIFO	1	13.5	26.8	23.9	3.75 kHz
		3.3	7.4	6.6	15 kHz
	3	1.8	6	6	
	6	1.5	5.5	5.5	
	12	1.4	5.3	5.1	
RR	1	13.3	26.9	23.6	3.75 kHz
		3.4	7.4	5.7	15 kHz
	3	1.8	5.9	5.4	
	6	1.5	5.5	5.3	
	12	1.4	5.3	5.2	

TABLE 3.5: Average number of scheduled users, `cbrS` = 256 Bytes

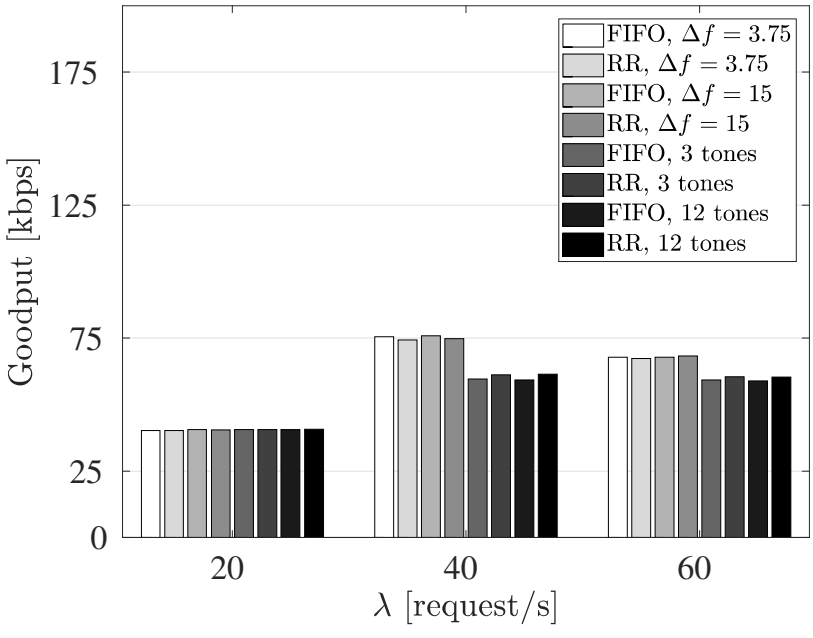
Scheduling Algorithm	**Number of Tones**	λ			Δf
		20	40	60	
FIFO	1	25	235	106	3.75 kHz
		7	202	70	15 kHz
	3	5	794	252	
	6	4	796	251	
	12	5	807	263	
RR	1	25	277	106	3.75 kHz
		7	254	76	15 kHz
	3	5	771	288	
	6	5	772	272	
	12	5	762	299	

than the number of users that generates packets to transmit during the unit of time. This means that NB-IoT is able to manage the amount of requests that successfully passes the random access procedure. When `cbrS` = 256 Bytes, instead, the amount of resources available in a simple configuration based on a 180 kHz carrier only is not suitable to support the overall traffic load. In this case, the system collapse and a very high number of users has to wait for a long time in the scheduling queue .The significant reduction in the average number of schedulable users is registered when $\lambda = 60$. In fact, many users are unable to complete the random access procedure and transmit a packet. In general, both schedulers offer a similar behavior. But, in a scenario with $\lambda = 40$ and Multi-Tone configuration, the RR scheduler handles queued users in a more efficient manner.

Figure 3.10 shows the average system goodput.



(A) cbrS = 128 Bytes



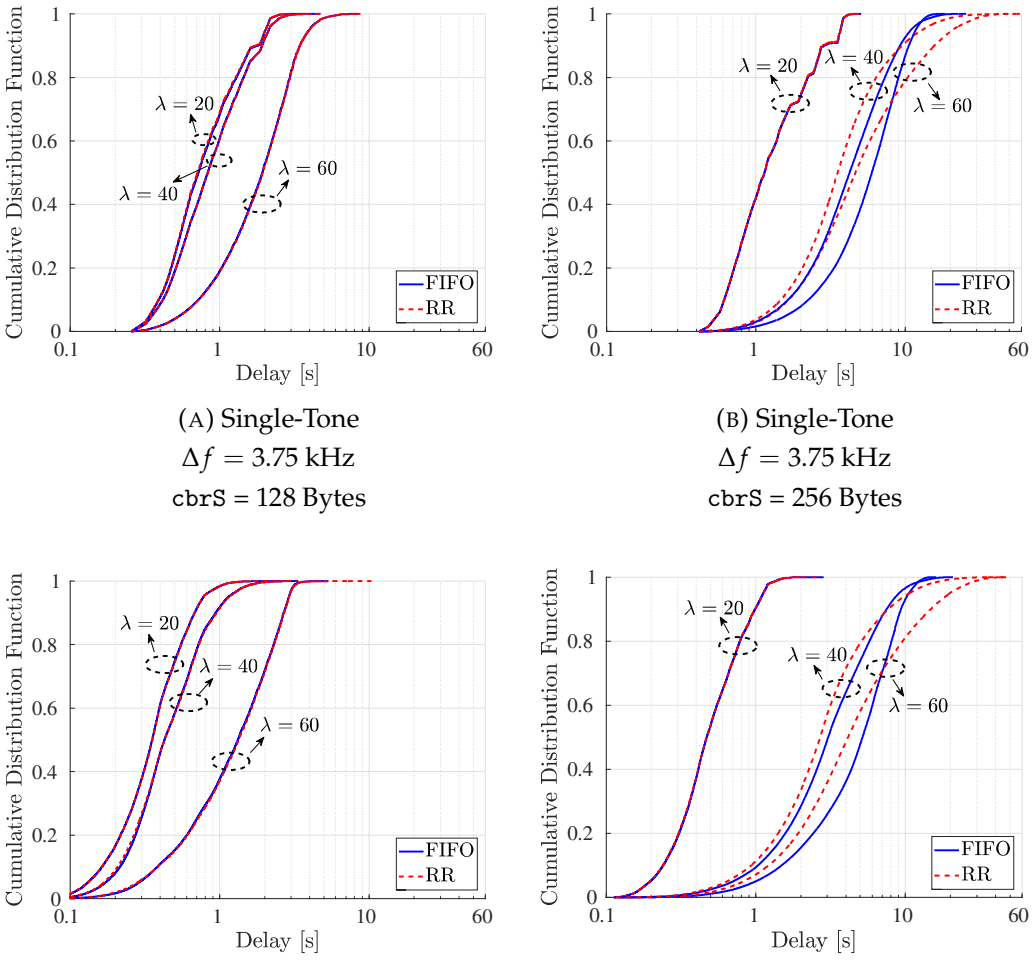
(B) cbrS = 256 Bytes

FIGURE 3.10: Average Goodput.

When cbrS = 128 Bytes, obtained results are nearly independent of the transmission type, Δf, number of tones, or the adopted scheduling algorithm. The results do not follow a monotonic behavior: moving from $\lambda = 40$ to $\lambda = 60$, the network registers a performance degradation. This is not caused by radio channel errors (which were not implemented yet), but it is due to the limited number of mobile terminals that completed the random

access procedure. These considerations are still valid also in the case `cbrS` = 256 Bytes. Nevertheless, Figure 3.10(B) demonstrates how Single-Tone mode is indeed capable of handling a higher number of transmission requests. In fact, more users can be scheduled at the same time during a TTI compared to the Multi-Tone configuration. The goodput is indeed improved.

Lastly, Figure 3.11 and Figure 3.12 shows the cumulative distribution functions of the E2E packet delays, calculated by considering the influence of random access procedure, scheduling decisions, and physical transmission.



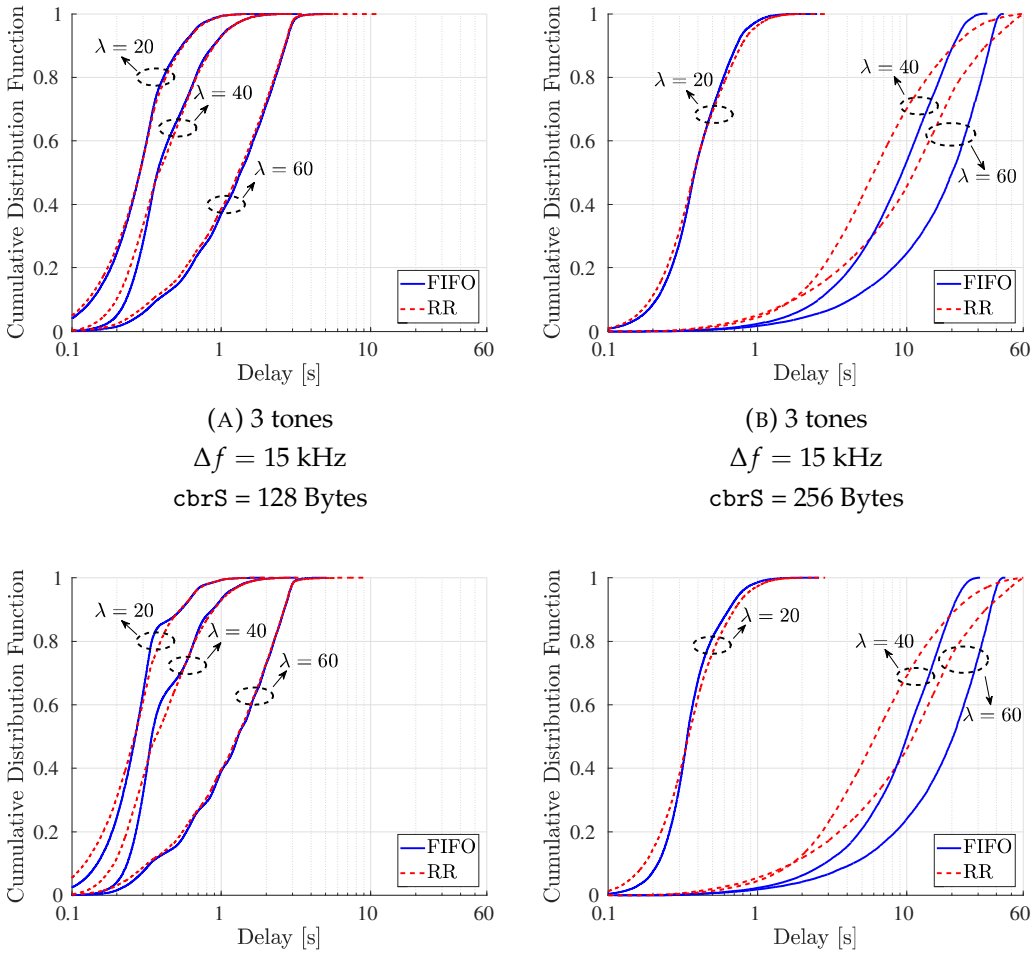
(A) Single-Tone $\Delta f = 3.75$ kHz `cbrS` = 128 Bytes

(B) Single-Tone $\Delta f = 3.75$ kHz `cbrS` = 256 Bytes

(C) Single-Tone $\Delta f = 15$ kHz `cbrS` = 128 Bytes

(D) Single-Tone $\Delta f = 15$ kHz `cbrS` = 256 Bytes

FIGURE 3.11: Cumulative distribution functions of E2E packet delay for Single-Tone.



(A) 3 tones
$\Delta f = 15$ kHz
cbrS = 128 Bytes

(B) 3 tones
$\Delta f = 15$ kHz
cbrS = 256 Bytes

(C) 12 tones
$\Delta f = 15$ kHz
cbrS = 128 Bytes

(D) 12 tones
$\Delta f = 15$ kHz
cbrS = 256 Bytes

FIGURE 3.12: Cumulative distribution functions of E2E packet delay for Multi-Tones.

On one side, when cbrS = 128 Bytes, both schedulers reach a similar performance. The reason is that the base station is able to empty the scheduling queue before the arrival of new requests. Therefore, scheduling decisions bring negligible differences. At the same time, however, packet delays grow with the value of λ. When λ increases, in fact, more users perform the random access procedure and the amount of time needed to complete it increases as well. Also, when $\Delta f = 3.75$ kHz (see Figure 3.11(A) and Figure 3.11(B)), longer duration of the TTI certainly gives rise to greater delays. But, a higher number of subcarriers translates into a limited susceptibility to the λ variation (the three curves are closer), since it allows simultaneous transmission of a greater number of users. On the other side, when cbrS = 256 Bytes and λ is greater than 20, the difference between scheduling policies

becomes more noticeable. In particular, RR guarantees lower delays with respect to FIFO for most of the users. Also, when $\lambda = 60$ higher delays are registered even if the number of users that are actually scheduled is less than the other cases (as confirmed by both Table 3.4, and Table 3.5). This solely depends on the longer time needed by mobile terminals to successfully complete the random access procedure.

3.5.1 Random Access Procedure

In the following, the accuracy of the model presented in Section 3.3 is evaluated, by taking into account reference applications scenarios of sensor networks enabling periodic reporting in monitoring infrastructures. Conducted tests consider a 3GPP reference scenario, as described in [178]. In particular, $\texttt{nUe} = M = 5000$ or $\texttt{nUe} = M = 10000$ motionless mobile terminals are uniformly distributed within a cell with a radius of 1.5 km. The base station uses a single NB-IoT carrier of 180 kHz. Again, transmission requests arrival is modeled according to the uniform distribution in the interval $(0, 60)$ s. All the details related to the considered NPRACH resource configurations are summarized in Table 3.6 and Table 3.7.

TABLE 3.6: NPRACH Configuration 1

Parameter	**Value**
aMax [#]	10
nCl [#]	1
Class c	***Normal***
[pA] [#]	6
α [#]	10
period [ms]	80
pA [#]	36
boW [ms]	512
rWind [ms]	24

TABLE 3.7: NPRACH Configuration 2

Parameter	**Value**		
`aMax` [#]	10		
`nCl` [#]	3		
Class c	***Normal***	***Extended***	***Extreme***
`pA` [#]	3	3	6
`period` [ms]	80	80	80
`pA` [#]	12	12	24
`boW` [ms]	256	512	1024
`rWind` [ms]	12	48	256

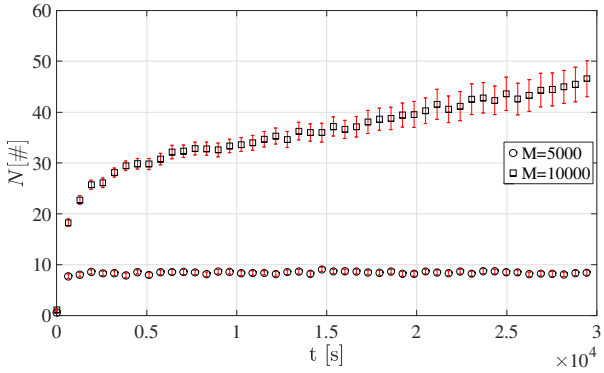
Note that for Configuration 1 the *Normal* class only is considered, while Configuration 2 contemplates all the three different coverage classes. It is worth mentioning that in order to increase the level of confidence of reported results, each simulation has been repeated 150 times.

System performance was evaluated in terms of the number of end-users involved in the random access procedure, as well as collision and success probabilities. Finally, a brief analysis of the percentage error between the estimated success probability, i.e., $\mathcal{S}_{N,s}$, and the simulated success probability, that is $\overline{\mathcal{S}_{N,s}}$, is conducted for each set of simulations.

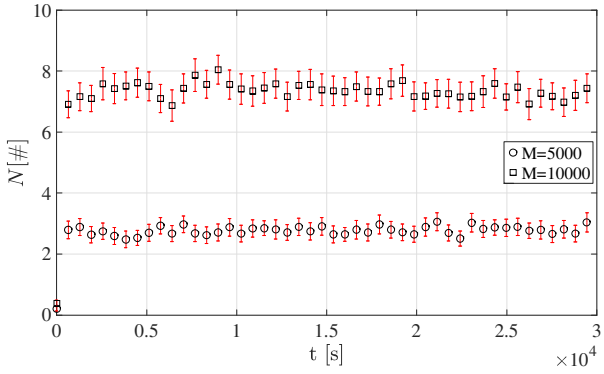
3.5.1.1 Number of Users Accessing the NPRACH

Figure 3.13 shows the number of users accessing RAOs, N. As expected, greater M values lead to an overall higher number of users in the NPRACH. This evidently holds for all the sets of simulations.

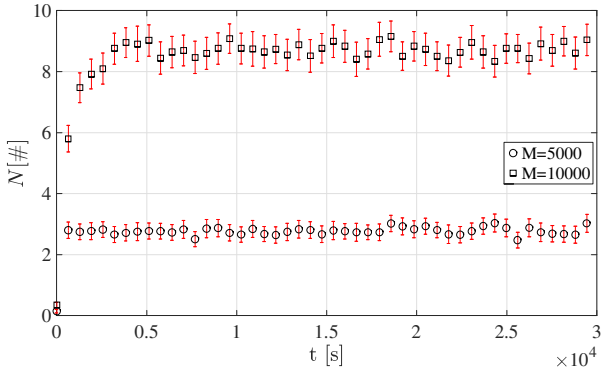
The average number of users accessing RAOs reaches the highest values in Configuration 1 since all the terminals belong to the only available coverage class, i.e., *Normal*. In addition, when $M = 10000$, the growth of the number of users performing the preamble transmission stems from the scarce amount of NPRACH resources. Conversely, the average number of users accessing RAOs is considerably lower in Configuration 2. In fact, mobile terminals are split among the three available coverage classes. It is important to emphasize that *Extended* class presents a higher N. This is mainly due both to the low number of different preambles, i.e., $s = 12$, and to the *Normal* class users attempting new transmission in this class. For the same reason the *Extreme* RAOs should be the most congested, however the number of different preambles is greater, i.e., $s = 24$.



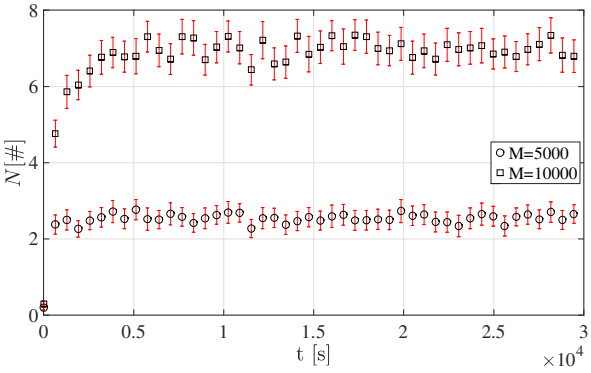
(A) *Normal* class, Configuration 1



(B) *Normal* class, Configuration 2



(C) *Extended* class, Configuration 2

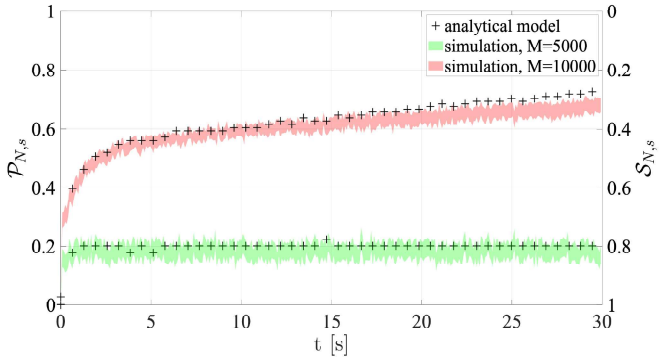


(D) *Extreme* class, Configuration 2

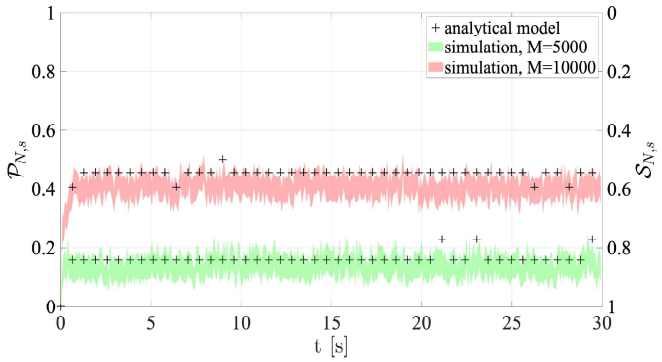
FIGURE 3.13: Average number of users accessing RAOs.

3.5.1.2 Collision and Success Probabilities

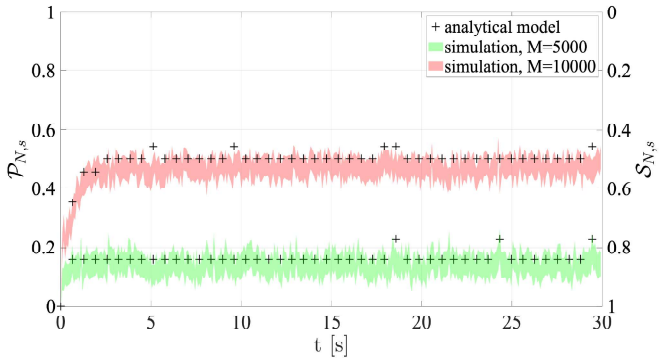
Figure 3.14 shows the success and collision probabilities in each RAO. It is important to note that each plot shows the collision and success probabilities directly derived from network simulations, as well as the probabilities computed using the presented analytical models, where the average number of users accessing RAOs computed by the tool is an input for the eq. (3.4) and the eq. (3.5). The most noticeable feature is that the predicted probabilities get closer to the actual probabilities as the number of users in the cell increases. Furthermore, the results are also consistent with the previous ones. In fact, when $M = 10000$, the collision probability is always higher than the $M = 5000$ case. All the figures demonstrate that performance is significantly reduced under high traffic load, especially for Configuration 1. In fact, according to what has been mentioned above, Configuration 1 shows the least $\mathcal{S}_{N,s}$ values. This depends on the fact that the NPRACH configuration is not adequate to the average number of preamble transmissions. As for Configuration 2, *Extreme* coverage class holds the greatest success probability, since $s = 24$, as outlined earlier.



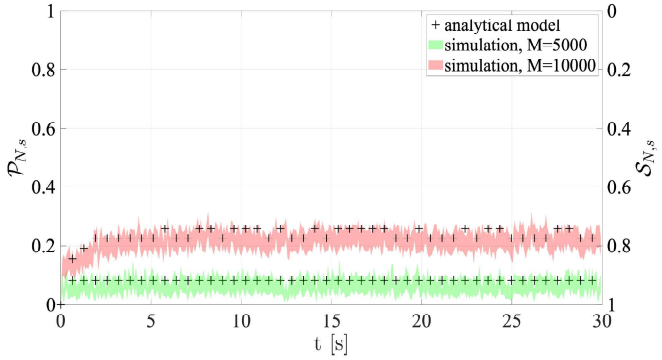
(A) *Normal* class, Configuration 1



(B) *Normal* class, Configuration 2



(C) *Extended* class, Configuration 2



(D) *Extreme* class, Configuration 2

FIGURE 3.14: Collision and Success probabilities of RAOs

3.5.1.3 Percentage Error of the Success Probabilities

As regards the cross-validation of the formulated model, the discrepancies between the simulated and the analytical success probabilities are quantified in terms of percentage error. The latter is reported in Table 3.8 and Table 3.9, which show its mean value, as well as the 90^{th} and the 10^{th} percentiles. Reported results clearly highlight that the analytical model boasts a good overall accuracy, keeping the mean percentage error always below 6% and the 90^{th} percentile slightly above 10%. In particular, Configuration 1 outperforms, demonstrating a mean percentage error lower than 3.5%. Nonetheless, in general, the analytical model is more accurate when the first resource configuration is adopted, according to the previous figures, too. This is mainly due to the fact that in Configuration 2 the average number of users accessing RAOs is considerably lower. Further analysis could be conducted to demonstrate that the analytical model actually holds higher accuracy when the number of users accessing an RAO is relatively large.

TABLE 3.8: Percentage Error Between Success Probabilities for Configuration 1

M [#]	Mean	90^{th} Percentile	10^{th} Percentile
5000	1.7	3.2	0.3
10000	3.4	6.5	0.6

TABLE 3.9: Percentage Error Between Success Probabilities for Configuration 2

Class	M [#]	Mean	90^{th} Percentile	10^{th} Percentile
Normal	5000	6.0	10.4	1.6
Extended	5000	5.9	10.5	1.4
Extreme	5000	2.5	10.5	1.4
Normal	10000	3.4	7.1	0.4
Extended	10000	4.2	8.1	0.7
Extreme	10000	2.6	8.1	0.4

3.5.2 Satellite NB-IoT

This Section presents a preliminary performance assessment of the NB-IoT satellite-based communication system for reference monitoring scenarios. To

demonstrate the actual effectiveness of the developed tool, the conducted system-level study highlights how network and satellite configuration significantly impact system performance.

3.5.2.1 Simulation Scenario

The reference scenario is called nb-Cell-sat and the command-line syntax to investigate it is:

```
./5G air simulator nbCell Sat nS nUe nC mcs nR
period boW spa (seed),
```

where:

- `nS` is the number of satellites per orbit;
- `nUe` is the number of users in the simulation area;
- `mcs` is theMCS to be used for the transmissions;
- `nR` is the number of NPUSCH and NPRACH repetitions;
- `spa` is the subcarrier spacing;
- `seed` is an optional seed to initialize random quantities to different, but reproducible, values in each simulation run.

For the simulation purpose, the fixed area on the Earth, that contains the NTN terminals, was chosen with a circular shape of the same size as the satellite spot beam.

At the application layer, the selected traffic model is the periodic uplink reporting [156]. In fact, monitoring is one of the most common use cases for MTC in NTNs [169]. The application payload size follows a Pareto distribution with shape parameter $= 2.5$, characterized by a minimum size of 20 bytes and a cut-off of 200 bytes. The split of inter-arrival time periodicity is 1 day (40%), 2 hours (40%), 1 hour (15%), and 30 minutes (5%).

Regarding the Random Access, the number of possible NPRACH preambles `nP` is the maximum allowed by the standard, i.e., 48. The NPRACH periodicity `period` is set to 240 ms while the Backoff Parameter `boW` is set to 2048 ms. In this way, the probability of collisions due to the preamble retransmissions may be mitigated. Moreover, it is important to emphasize that these values are also compatible with higher RTTs typical of NTNs [165]. The duration of the RAR window and the Contention Resolution Timer is set in accordance with the defined number of NPRACH repetitions.

TABLE 3.10: Parameters of the Scenario

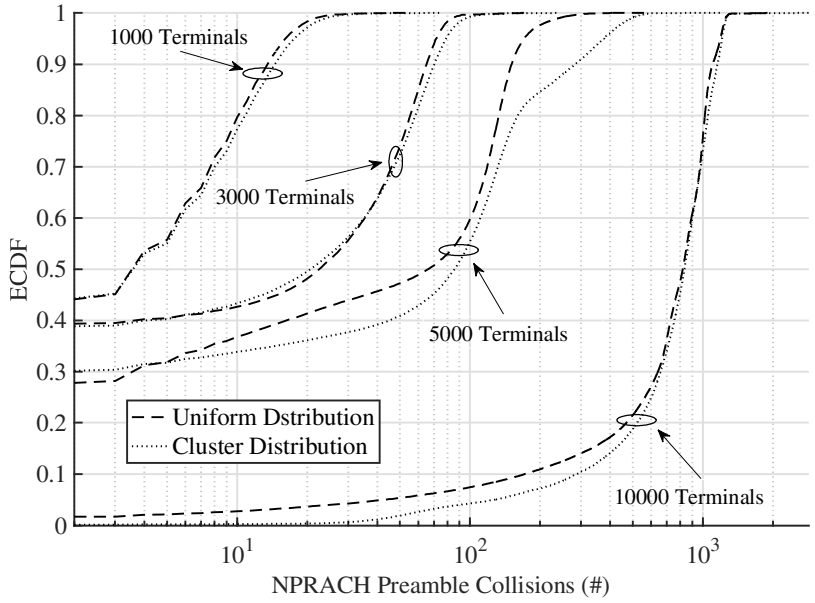
Parameter	Value
nS	4, 8 satellites
nUE	1000, 3000, 5000, 10000 users [178]
MCL	164 dB
MCS	2, 4, 6, 8
nR	1, 2, 4
uplinkBand	1980-2000 MHz
uplinkConfig	Single-Tone
spa	3.75, 15 kHz
nP	48
period	240 ms
boW	2048 ms
ueAntennaPower	23 dBm
eNBAntennaPower	33 dBm
seed	1-50

Although the simulator provides support to multiple coverage classes, in this case, only one coverage class has been considered since all the NTN terminals are supposed to experience the same level of coverage with respect to the satellite. In particular, an MCL value of 164 dB is considered. The chosen scheduling algorithm is the Round Robin since it is already proven to guarantee lower average delays with respect to FIFO and avoid the starvation problem. The duration of the simulation has been chosen in order to allow a vision of at least 8 cycles of visibility by the satellites on the area involved in the communication. A 20 MHz bandwidth from 1980 MHz to 2000 MHz frequency and a single NB-IoT carrier are taken into account. The main parameter settings considered in this study are summarized in Table 3.10.

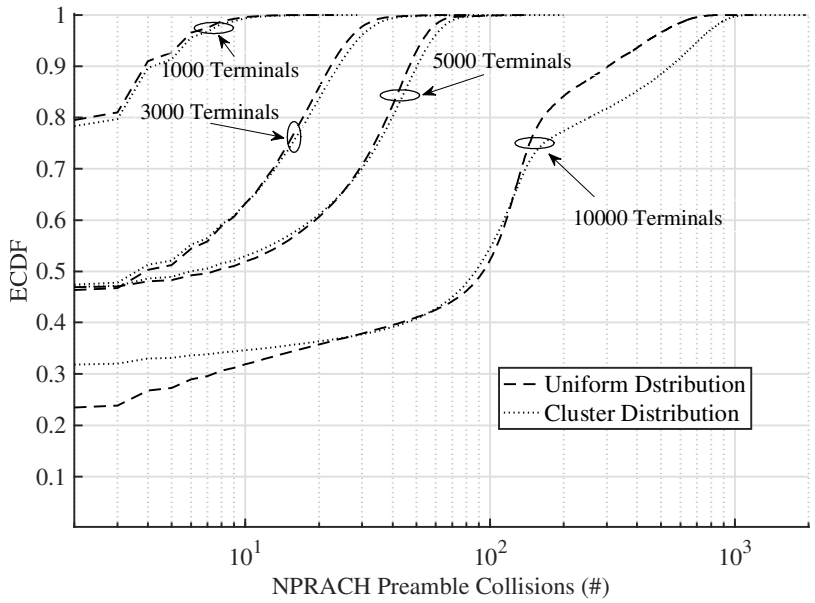
Different KPIs have been measured by processing the output trace files. In particular, the number of NPRACH preamble collisions and the E2E delay are statistically analyzed. Finally, the average delivery ratio of the packets is investigated for further completeness of the performance evaluation.

3.5.2.2 NPRACH Preamble Collision

Figure 3.15 shows the number of NPRACH preamble collisions.



(A) 4 Cubesats



(B) 8 Cubesats

FIGURE 3.15: ECDF of the NPRACH Preamble collisions

First of all, the number of Cubesats in the Satellite Platform greatly impacts NPRACH performance. In fact, with fewer Cubesats, ground terminals remain without satellite coverage for longer periods. As soon as they return in visibility, a great burst of NPRACH preamble transmissions occurs, hence leading to several collisions.

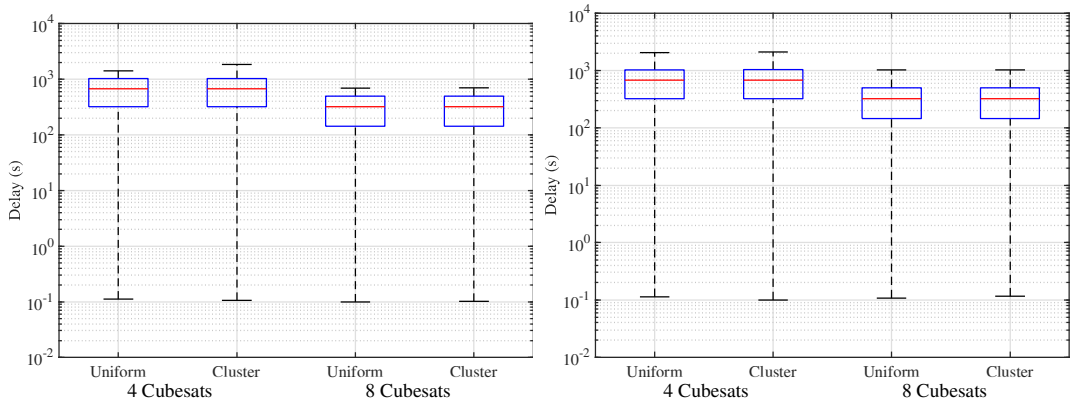
Besides, also a greater number of NTN terminals leads to an overall higher number of preamble collisions, as expected. For instance, with 4 Cubesats and 10000 NTN terminals, the probability of having less than 100

collisions is below 10%. This demonstrates that NPRACH represents a bottleneck for dense network deployments.

Finally, it is important to highlight that the cluster distribution impacts negatively on the performance, and this is more true for a higher number of terminals. Indeed, according to the implemented mobility models, all the users of a cluster come back in coverage practically at the same time. In contrast, the attachment is more gradual when NTN terminals are uniformly distributed.

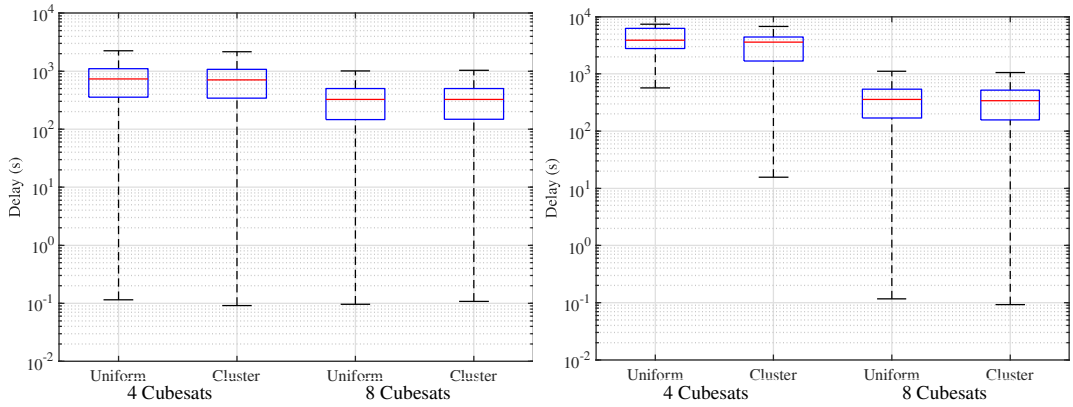
3.5.2.3 End-to-End Packet Delays

End-to-End packet delays are reported in Figure 3.16.



(A) 1000 NTN Terminals

(B) 3000 NTN Terminals



(C) 5000 NTN Terminals

(D) 10000 NTN Terminals

FIGURE 3.16: Box plots of the E2E packet delays. Each box plots identifies the median delay (i.e., the red line), the 25^{th} and the 75^{th} percentile (i.e., the bottom line and the top line of the blue rectangle), as well as the minimum and the maximum measured delay value (i.e., the edges of the vertical black line).

They are computed by taking into account the influence of cell selection, random access procedure, scheduling decisions, and the actual physical transmission. Following the previous NPRACH considerations, the most noticeable feature is that the constellation numerousness significantly affects the E2E packet delays. In particular, more Cubesats allow covering NTN terminals for more protracted periods, hence reducing E2E delays. Besides, the amount of time needed to complete the random access procedure increases with a higher number of NTN terminals. Indeed, when more users perform the random access procedure, the number of collisions rises. As a consequence, packet delays also grow with the number of NTN terminals.

It is important to mention that, for 3000 and 5000 NTN terminals, the difference in performance is barely noticeable. A considerably higher delay would be expected in the case of 5000 NTN terminals since the number of collisions is higher. Nonetheless, the scheduling delay is limited in such a case, precisely because fewer users successfully complete the random access procedure. The other way around happens with 3000 NTN terminals. As a consequence, performance is similar in both cases.

3.5.2.4 Delivery Ratio

To conclude, the packet delivery ratio (i.e., the ratio between correctly received packets and transmitted packets) is analyzed for all the sets of simulations. Specifically, Figure 3.17 shows the achieved average delivery ratio.

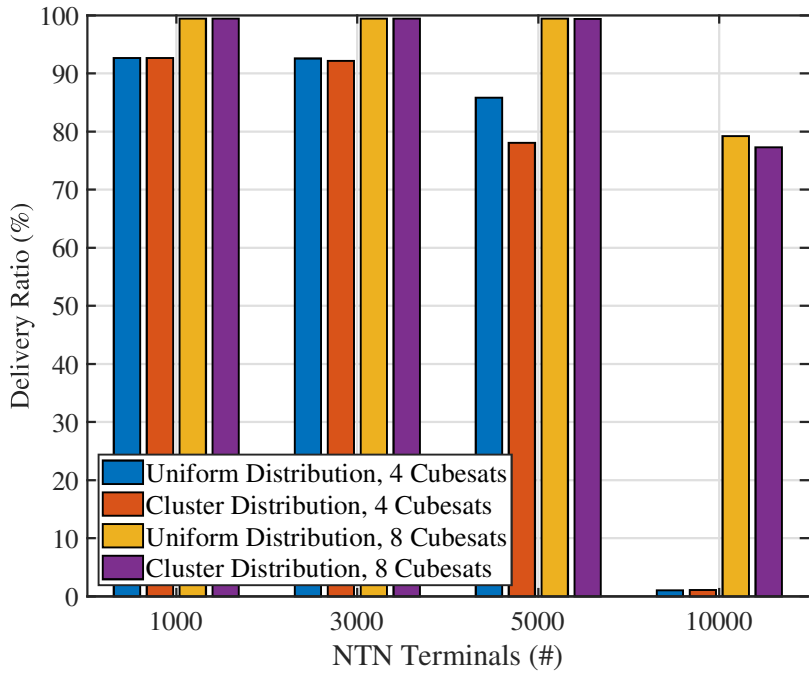


FIGURE 3.17: Delivery Ratio.

Evidently, 8 Cubesats constellations hold the greatest packet delivery ratios. Conversely, when a massive number of NTN terminals is deployed, performance is extremely reduced. Nonetheless, it is important to emphasize that fewer satellites provide decent performance (e.g., delivery ratios higher than 90%) for a reduced number of the NTN terminals. This also highlights the fact that proper constellation dimensioning is crucial.

Finally, the aforementioned considerations for NPRACH and delays reflect in the obtained delivery ratios, therefore proving the accuracy of the analysis.

Chapter 4

Dynamic Management of RAN Slicing

The idea to support orthogonal logical segments also at the radio interface of 5G and B5G deployments recently gained momentum. In the RAN, slicing is based on the virtualization of the radio resources and strongly leverages SDN and NFV to create different E2E virtual networks over the same physical infrastructure. Unlike the conventional network slicing concept, RAN slicing is less mature and more challenging, because of the intrinsically shared and unpredictable nature of wireless resources and the need of novel RRM functionalities [69]. Indeed, the integration in the RAN of the same attributes of core network slicing is a complex task, which requires the definition of novel RRM functionalities. e.g., spectrum planning, interference coordination, packet scheduling, and admission control [85].

This Chapter first describe the current state of the art on RAN slicing. Then, a novel architecture to realize TNT-driven RAN slicing for Latency Sensitive Services is explored from subsection 4.2.1 to Section 4.4, with a specific focus on AI techniques. Finally, Section 4.6 concludes the Chapter with the proposal of applying the concept of RAN slicing to WLAN in order support indoor healthcare monitoring, with a focus on epileptic patients.

4.1 State of the Art

The management of network slicing in the core network was deeply investigated in the current scientific literature. On the contrary, handling network

slicing in the RAN is still an open issue. In fact, the unpredictable variability of the wireless channel, network dynamics, slice isolation, scarcity of resources, increased inter-cell/inter-tier interference caused by spatial multiplexing of the spectrum, as well as diverse QoS requirements of different services pose significant technical challenges on the management and provisioning of RAN slicing [47], [69]. A straightforward way of allocating resources to different slices is through static partitioning of the resources, which can, however, lead to low efficiency. At the same time, static partitioning is not adequate to enforce the full vision of network slicing where different operators wish to have a detailed vision of the underlying infrastructure as a virtualized entity of which they have a - at least partial - control and which they can configure and operate independently.

To overcome these issues, recently, many scientific works dealt with the realization of RAN slices.

In [179], the authors propose a 3GPP compliant vision and a new framework to enforce network slices, featuring radio resources abstraction. In particular, a fully programmable network slicing architecture based on the 3GPP dedicated core network and a flexible RAN is proposed. As for the RAN, a two-level MAC scheduler to abstract and share the physical resources among slices is conceived. The idea of dividing the slicing scheduling problem at RAN level into inter and intra-slice scheduling is then considered in many subsequent works, e.g., see [180].

In [181] the concept of the 5G Network Slice Broker is introduced, which enables new players to request and lease resources from IPs dynamically via well-defined interfaces. Specifically, its behavior is based on traffic monitoring, and it takes into account SLAs. In addition, it configures RAN schedulers to provide an inter-slice resource allocation as well as the selection of RBs from a shared pooled spectrum.

An interesting holistic vision of RAN slicing addressing the above mentioned considerations, is presented in [48]. In this work, the importance of RAN slicing enforcement in 5G networks is discussed and an overview of the RAN slicing formation procedures involving an IP and multiple TNTs is presented, from the operator's selection of base stations to the waveform-level scheduling of radio resources. Moreover, timing aspects involving the creation of the slices and the lower-layer MAC scheduling procedures are discussed, putting in evidence that different time scales for slice formation and radio resource allocation are in general expected.

In [85] an overall analysis of the RAN slicing problem in a multi-cell network is presented and four different slicing approaches working at different RRM functionalities levels are proposed, namely spectrum planning, Inter-Cell Interference Coordination, Packet Scheduling, and Admission Control levels. Different solutions achieve a different trade-off between isolation and optimized resource utilization.

The problem of slice enforcement expressed as RBs allocation to each slice for a given resource partitioning policy among the admitted slices in presented in [182], addressing, in particular, the isolation problem, i.e., the possibility that a non-optimal RB allocation introduces interference among different operators. Here, the concept of linked bandwidth is introduced, intended as the amount of frequency resources shared among users of the same slice in adjacent (i.e., interfering) cells.

A distributed approach for performing RAN slice formation among competing TNTs is presented in [183] and [184]. In [183] the problem is formulated as a potential game where the players are the different operators, and the goal is to optimize a global utility. In this scenario, each operator aims at selecting the serving base stations for each node with the aim of minimizing the cost and the congestion level, as well. The timing problem of slice formation is not specifically considered, although the possibility of adapting to the number and the positions of active nodes would require fast slicing allocation. In [184] a game-theoretic approach to realize network slicing is presented, enabling TNTs to both gain the benefits of infrastructure sharing and customize their own users' allocation.

Some other papers related to RAN slicing, e.g., [185], [186], and [187] envisage an architecture where the control part of the RAN, i.e., Layer 2 scheduling, base stations assignment, etc., are partially in charge of the TNT, that is responsible for making slice formation requests and for operating slice enforcement using open interfaces to communicate with the IP. In particular, [185] proposes a network slicing solution for enabling efficient coexistence of different services. In this setting, the authors try to validate the feasibility of on-demand creation and configuration of network slices by means of an SDN-based slicing application. In [186] and [187], the problem of slice formation and resource allocation are considered as coupled in a single resource allocation framework, where the general problem is seen as a hierarchical allocation problem in which the slice formation is executed in a distributed fashion involving the IPs and the TNTs which compete for accessing the required resources. More specifically, a hierarchical radio resource allocation

architecture is proposed in [186], where a global resource manager is responsible for allocating sub-channels to local radio resource managers in slices, which then allocate the assigned resources to their users. Under this architecture, a hierarchical resource allocation problem is formulated. In [187] a two-level hierarchical resource allocation problem is proposed, while ensuring efficient resource allocation, inter-slice isolation, and intra-slice customization. To this end, authors design a hierarchical combinatorial auction mechanism, from which a truthful and sub-efficient allocation framework stems.

The problem of providing URLLC using 5G network slices is addressed in [188]–[191]. Specifically, [188] provides an overview on current technologies, open issues, and possible solutions to the network slicing problem for URLLC service type. The analysis is conducted focusing on stringent E2E delay guarantees. Authors of [189] propose a flexible two-level scheduling framework in order to dynamically manage radio resources to meet the typical latency and reliability requirements. They focus on reducing latency for URLLC services and applies different per-slice scheduling policies over a shared RAN. [190] presents a novel framework for 5G RAN slicing by exploiting the idea of Earliest Deadline First scheduling. In particular, each slice is scheduled independently taking into account the list in order and Self Organizing Networks are used for dynamically adapting to potential SLA violations. In the context of Fog RANs, [191] formulates a delay-aware resource allocation optimization problem in order to satisfy the latency constraints of an URLLC slice. Authors prove that guaranteeing the delay performance comes at the expenses of the throughput and the queuing delay of eMBB slices.

The problem of networking, computation, and storage resources allocation to different TNTs was studied in [192] where a set of access points equipped with MEC servers are available to provide the requested resources. More specifically, the inter-dependency among different kinds of resources were taken into account to elaborate an optimal resource allocation strategy.

The study of resource allocation for V2V communication slice is investigated in [193] where the RAN slicing problem for providing two generic 5G services, namely eMBB and V2V, is addressed. The scheme was based on an off-line RL algorithm which allocates radio resources to different slices with the target of maximizing the resource utilization while ensuring the availability of resources to fulfill the requirements of the traffic of each RAN slice.

In [194], the authors proposes a network slicing based communication

for vehicular networks by creating two different slices, namely autonomous driving slice and infotainment slice, over the same physical infrastructures. The slicing algorithm partitioned the vehicles into different clusters and allocate slice leaders to each cluster. Slice leaders serve its clustered vehicles with high quality V2V links providing Low latency services. At the same time Road Side Unit (RSU) provides infotainment service using Vehicle to Infrastructure (V2I) links. The performance of the proposed method was validated using an LTE-A compliant system level simulator, with enhancement of cellular V2X standard by analysing queuing latency, throughput, and the queue lengths. Simulation result showed the proposed network slicing approach attains huge improvements in terms of reliability and throughput, which is due to the utilization of high quality V2V and V2I links. However, the problem of inter-slice interference was not studied in this work.

In [195], the authors implement a mobility management scheme for V2X slicing environment. The proposed solution focused mainly on resource borrowing between slices, aiming at reducing the handover call dropping probability.

A lot of effort from standardization bodies is going into the exploitation of the 5G slicing framework for enhanced V2X services. Specifically, whether and how to correctly map enhanced V2X services in different slices and which enhanced V2X features would require or benefit from explicit network slicing support are still being discussed by 3GPP [196].

AI-based methods represent powerful instruments for solving typical technical problems of interest for Academia and Industry working on mobile communication systems. In particular, DL, RL, and DRL have been widely adopted for channel estimation and resource allocation problems [57], [64], [197], [198]. In this context, even in the presence of perfect traffic estimation, evaluating the optimal RRM setting is a very difficult task owing to the random nature of the radio conditions, requiring the use of optimization tools with unmanageable computational complexity. Alternatively, DL and RL/DRL offer low-complexity and effective solutions for RRM in communication and computing systems [54], [199]. DL can extract important features from data and model its high-level abstractions, avoiding manual description of a data structure [54], [57]. For this peculiarity, DL architectures have been successfully proposed in channel estimation. For example, Long Short-Term Memory (LSTM) networks perform the prediction of RAN resource usage by a network slice [200] and the prediction of future CQI values in a data-driven RAN slicing framework with URLLC and eMBB slices [201]. Furthermore, an

encoder-decoder structure based on Convolutional Neural Network (CNN) is presented in [202] for estimating the traffic of slices deployed at Cloud - Radio Access Network (C-RAN), MEC, and core datacenters.

The time-varying wireless channel largely impacts the optimal decision-making process for resource allocation problems. Differently from traditional solutions that require to rerun the algorithms every time the environment changes, RL and DRL methods fit for these challenges [57]. A slice admission strategy based on RL is presented in [203] for a flexible RAN. As alrerady reported, Q-learning is also adopted in [193] to handle RAN slicing, supporting an eMBB and a V2X slice on the same RAN infrastructure.

In [204], LSTM is incorporated into the actor-critic DRL algorithm for an intelligent resource management of RAN slicing. Deep Q-Network [205], [206] and its modified versions [207], [208] are exploited for slice management in RAN. Specifically, the contribution in [207] entails a Generative Adversarial Network-powered Deep distributional Q-Network for demand-aware resource allocation, while RB allocation to multiple slices is optimized in [208] by exploiting a method called Ape-X, that uses distributed learning in the Deep Q-Network (DQN) with multiple actors. Cooperative multi-agent deep Q-learning jointly solves the RAN slicing and computing task offloading problem in [209]. Moreover, by jointly optimizing radio and computation resources in the context of RAN network slicing, the utility maximization problem formulated as a Markov Decision Process (MDP) is solved in [210] through the Deep Deterministic Policy Gradient (DDPG) algorithm, that combines the DQN and the actor-critic approach. Similarly, the contributions in [211] and [212] extend the DDPG algorithm to obtain an optimal RAN slicing policy, by minimizing the long-term system cost in the context of vehicular networks and both the long-term QoS of services and spectrum efficiency of slices, respectively. Q-learning [213], Deep Q-learning [214], and a distributed DRL strategy based on the Advantage Actor Critic (A2C) algorithm [215] also assist network slicing involving both RAN and core network.

DL can also support DRL-based resource allocation methods. In particular, the compression of high-dimensional CQI information, obtained through an autoencoder, is exploited in a DQN-based framework in [198]. This valuable contribution aims at optimizing computation offloading in the large-scale MEC system, but it does not focus on network slicing problem. Autoencoders are also adopted in the core network slicing context. In particular, the framework proposed in [50] firstly entails an autoencoder-based

classifier, which is used by the IP to distribute TNTs' virtual network slicing requests with similar characteristics to its different agents. Then, an autoencoder-based compression module extracts the key features of the virtual network requests. In a privacy-oriented approach, the compressed representation of features is fed into a DDPG-based model for resource pricing, advertising, and motivating TNTs to request resources in a load-balanced manner. Therefore, virtual network slicing is accomplished in a distributed, TNT-driven, and privacy-oriented manner: after compressing the features of requests, TNTs compute their own virtual network embedding schemes independently and distributedly, according to the resource information (i.e., the available resources and their prices) advertised by the DRL agent.

4.2 Architecting RAN Slicing for Latency Sensitive Services

The inherent requirements of users demanding latency sensitive services, as well as their high mobility, put further constraints on the RAN slicing problem. Indeed, it is evidently clear that forming and handling network slices with severe latency and reliability guarantees for mission-critical applications is a dismaying effort. Current scientific literature, as it stands, does not adequately account for the fact that the delay and reliability requirements of latency sensitive services call for unconventional, distributed, and scalable slicing approaches. Furthermore, recent research on RRM for URLLC [67], [189], [216]–[219] has made great strides towards a more efficient scheduling of these services. Nevertheless, these studies have ignored to take into account a common infrastructure shared by multiple TNTs.

Moreover, it is equally important to acknowledge the possibilities offered by MEC and virtualization. In fact, thanks to the use of the servers installed at the edge of the network , MEC can guarantee extremely low latency and bandwidth efficiency, differently from traditional centralized cloud computing [220]. The innate ability to collect information about users and network could also be exploited to provide personalized services to users and third-parties, which represents a fascinating possibility for TNTs.

In essence, a virtualized application platform is built over physical hardware resources provided by the host machine that mounts the MEC server within the C-RAN. This application platform is capable of offering a series of middleware services accessible to the applications running on the MEC

server through Application Programming Interfaces (APIs). In particular, they include *Infrastructure Services*, which manage the communication between application and service and between the various applications, and also allow the visibility of services offered by the MEC platform, *Radio Network Information Services*, that collect and provide information about users and cell, and *Traffic Offload Function*, which manages the routing of traffic to and from applications. From the preceding discussion, it is clear that MEC servers deployment in the RAN is one of the best solutions for RAN slicing, guaranteeing both a strong versatility to extremely time-variant conditions and easiness of interactions between all the parties involved in the slicing operations.

It is even more true that the combination of different technology emerges as a foundation for supporting reliable and low-latency services. Indeed, there several promising approaches that are beneficial for URLLC as well, especially with the advent of resource-intensive and safety-critical applications. First, there are Non Orthogonal Multiple Access (NOMA) techniques, which utilize the superposition coding principle to multiplex multiple users on the same time/frequency resources. As a consequence, they are able to effectively lessen latency by supporting far more users than conventional orthogonal-based approaches, hence reducing both queueing delays and the need for more spectrum. Second, already standardized 5G NR features are remarkably convenient to improve the performance of the over-the-air transmission delay. In particular, shorter TTIs can be achieved by operating at larger subcarrier spacings, hence exploiting the flexibility of the NR interface. Moreover, small scheduling units such as mini-slots yield substantial latency reduction. Third, the complex RRM problem could be addressed by means of a distributed machine learning framework. Specifically, essential data is distributedly saved across several interconnected nodes, and the problem is then solved collectively, eventually involving end devices as well [221]. Such a distributed mechanism is mandatory since a centralized intelligent node is evidently inadequate for addressing the URLLC requirements. A possible solution could be to locate the intelligent framework in a RAN controller, which then forwards the decisions to the actual base stations. Finally, many more examples are possible: ultra-lean design, grant-free transmissions, flexible TDD configurations, device-to-device communication, delay-budget reporting, etc..

Starting from the valuable methodologies and solutions available in the current state of the art, this Section sheds some important basis for the design

of a comprehensive architecture enabling RAN slicing for latency sensitive services. Specifically, it presents a high-level architecture jointly tackling the problems related to latency sensitive services and network slicing by means of a MEC system, while posing particular attention to design criteria, system components, and their baseline interactions. The conceived architecture grounds its roots on the Platform as a Service (PaaS) paradigm: it is assumed that the IP is the only entity allowed to manage radio resources and network slices. Nevertheless, the overall design suitably fits with the Infrastructure as a Service (IaaS) paradigm as well, where virtual mobile operator have the chance to control radio resources and network slices.

4.2.1 The Proposed Architecture

Although network slicing is expected to open new business models for all the interested parties, it is of the utmost importance to emphasize that roles must not change. For instance, IPs and third-party vertical industries that request mobile slices should not be aware of TNTs' most valuable information, i.e., business strategies, subscriber numbers, etc.. At the same time, although with the necessary exceptions, TNTs are not authorized to precisely comprehend the procedures and algorithms implemented by the IPs, as well as the internal functioning of the provided apparatuses. Based on these premises, here is envisioned an architecture able to advance the current mobile network slicing context, favoring collaboration while keeping key requirements and distinctive interactions among all the parties involved clearly distinct. As a consequence, the IP is the only entity allowed to implement resource allocation. In addition, for simplicity reason, the network of a single IP is considered in the subsequent analysis. However, it is important to note that the proposed architecture can be easily scaled for multiple IPs.

4.2.1.1 Overview

A general network scenario constituted by a New Generation Service Oriented Core and a C-RAN is considered. In particular, C-RAN architecture is used on the RAN side in order to effectively implement real time functions, as well as RAN slicing, on-demand deployment of resources, and flexible coordination. Moreover, the C-RAN is equipped with the Mobile Cloud Entity (MCE), i.e., the logical entity of central control and management for

the C-RAN. MCE can implement functions requiring high real-time performance and computing load — e.g., access network scheduling, link adaptation, power control, interference coordination, retransmission, modulation and coding — based on different service requirements and resource configurations. According to the IP implementation, C-RAN can have different levels of control of radio functions, from an overall control of the gNBs to the complete Radio Remote Heads capabilities. Clearly, RAN is composed of numerous gNBs settled in different geographic areas. Nearby gNBs are then grouped together in a C-RAN cluster, a spatially isolated and self-sufficient logical network partition. One or several MEC servers are then bootstrapped to each C-RAN cluster and connected to one or more RAN controllers supporting the underlying physical infrastructure.

In order to automatically generate network slicing services according to the specified requirements, two different entities, namely IP Subsystem and TNT Subsystem, are envisaged to interact with each other. In this scenario, the IP dynamically leases computing resources for granting TNT Subsystems to be virtualized on MEC, according to the *PaaS* paradigm. A high-level overview of this architecture is depicted in Figure 4.1.

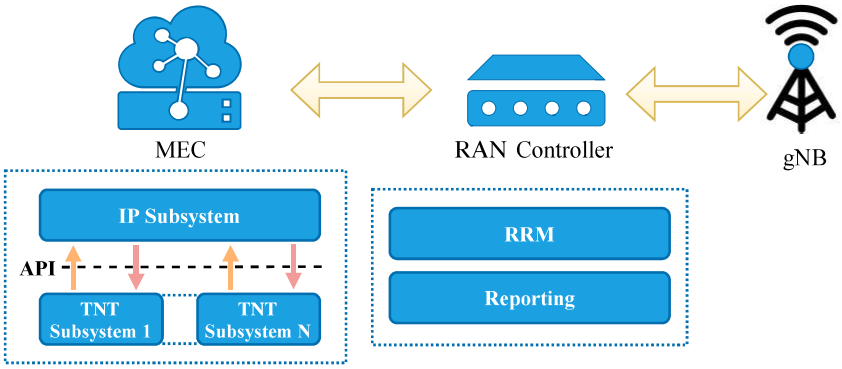


FIGURE 4.1: Functions and interactions between the elements of the proposed architecture.

It is important to note that a different *IaaS*-based architecture, in which the IP only leases its resources and the TNTs are in charge of developing their own virtualized subsystems, although coming at a much more convenient architectural simplicity, is able to provide less control over the slice creation, as well as minor coordination between the different TNTs, which is crucial in the RAN slicing context, as repeatedly stated. This is the main reason why

the proposed architecture is built on top of the *PaaS* paradigm. However, it is important to remark that the envisioned network scenario is flexibly suitable to both *IaaS* and *PaaS* architecture deployments.

4.2.1.2 IP Subsystem

In essence, it is in charge of handling the creation of different RAN slices according to the control directives coming from the TNT Subsystems, which are first translated in order to be effectively managed. Here, the IP provides specific APIs for the submission of slice requests. The complex directives are then turned into simpler records containing all the important parameters — e.g., QoS requirements, frame structure, transmission configuration, etc.. The resource requirements are carefully checked in order to determine whether a slice should be admitted or not. Admission control is thus performed on a per-slice basis by taking into account the SLAs with the TNTs as well as the reporting information from the gNBs. To this end, the IP Subsystem directly communicates with the controllers managed by the C-RAN sending network-level RAN performance requirements necessary to create and/or enforce already present RAN slices.

The design criteria regarding which parameters the TNTs and IP are allowed to share, as well as the level of control the TNTs can have on the C-RAN are critically important, especially for latency sensitive services. In fact, the variability of the service requirements may call for feedback messages between the IP and the TNTs Subsystems (or toward third-party verticals) leading to a slice renegotiation, which must happen on a very small time scale. For instance, after having formed a latency sensitive slice, it may happen that suddenly the channel conditions have changed so much, or that the MEC server is overloaded, that no enforced RRM algorithm can guarantee the satisfaction of the constraints provided by the resource allocation policy established for that slice at the time of its creation.

4.2.1.3 TNT Subsystem

TNT Subsystems mainly formulate slice requests by specifying both general information (e.g., the time duration of the slice, type of services to be provided) and high-level control directives in order to successfully address the requirements for the requested slice. Requests are forwarded to the IP Subsystem through the provided APIs. In addition, each TNT Subsystem has to

continuously monitor customer requirements based on current network status in order to check whether SLA violations happen and to properly reconfigure the parameters included in the slice requests. Based on the previous behavior of the IP Subsystem, if multiple slice requests need to be forwarded, it is in charge of the assignment of inter-slice priorities according to the required SLAs. For this purpose, it is necessary to be informed in advance and to have the faculty both to renegotiate the slice and to decide to interrupt other flows, if they have less priority.

Handling of RAN slice requests submitted by third-party entities should be supported as well. Vertical industries subscribe a specific plan with TNTs who are then completely in charge of managing the sliced networks according to the agreed decisions.

4.3 DRL-Aided RAN Slicing Enforcement for Latency Sensitive Services

Edge Intelligence (EI) is considered one of the most powerful enabling technology for RAN slicing enforcement. By leveraging the native capabilities of both MEC and Artificial Intelligence, it promises to simplify the large-scale data acquisition, predict the incoming agglomerated per-slice traffic, and efficiently support resource allocation, management, orchestration, and network automation [222].

At the time of this writing, several AI-based solutions to anticipate future offered loads in mobile networks have been extensively studied [200], [202], [223]. However, estimating the traffic only represents a partial step for the optimal slice resource allocation problem. Even in the presence of perfect traffic estimation, in fact, evaluating the optimal RRM is a very difficult task owing to the random nature of the radio conditions. Furthermore, online optimization algorithms require relatively powerful computing devices for real-time applications, which raises monetary concerns or unmanageable computational complexity. They also often require substantial problem-specific knowledge for building them. The inherent requirements of latency sensitive services, as already reported, put further constraints on this problem and call for unconventional and scalable slicing approaches. For these reasons, RL has been recently investigated as a low-complexity and effective solution for RRM in communication and computing systems [224], [199]. Here, a RL agent can generate (near-) optimal control actions on the basis of the

reward feedback from interactions with the environment. Instead of simply optimizing the current reward in a greedy manner, the RL agent can take a long-term goal into account. Thus, RL appears to be particularly suited for RRM problems when the optimum is very difficult to know, only a reward associated with a given policy is available, and the reward/loss cannot be related, through a closed-form expression, to the actions [225].

Following these premises, this Section presents a RAN slicing solution that exploits DRL-based EI to serve latency sensitive applications. Note that unlike other state-of-the-art studies [199], [202], [206], [208], [223], [224], and in line with the architecture presented in subsection 4.2.1,the conceived methodology considers the openness of the network to third parties, hence encouraging TNTs to directly take control of the resources and drive the slice enforcement without relying on an IP-centralized solution. Section 4.3.2 considers a use-case based on autonomous-driving to show the effectiveness of the proposal against baseline methodologies.

4.3.1 The proposed RAN Slicing with EI

The IP determines the amount of resources that can be used by the TNTs for each slice. The TNTs, in turn, should adapt in real-time their requests according to their own users' requirements, avoiding expenses due to the issue of resources overbuying. As a consequence, the generation of *slice requests*, i.e., when a TNT defines its needed slice configuration to the IP, and the *slice dynamic enforcement*, i.e., the adaptation of the slice allocation policy to the time-varying RAN environment, are the solutions of local optimization problems, which have to be solved in real-time and whose decisions have to be executed instantaneously to reduce any latency of the system [48].

The reference scenario is shown in Figure 4.2.

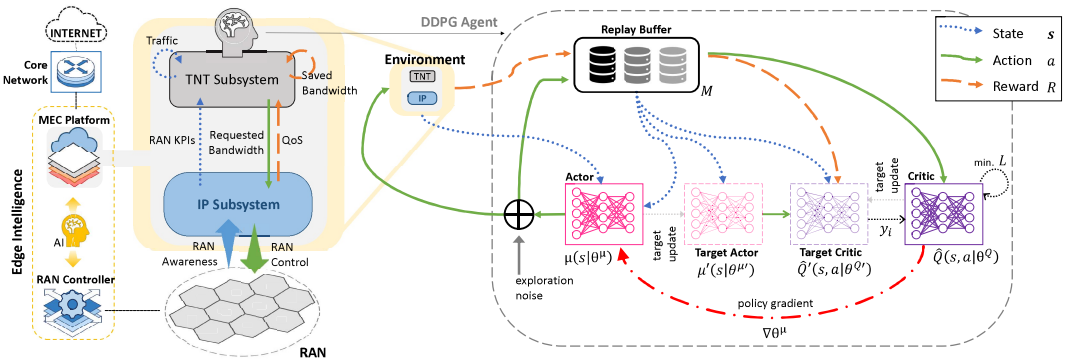


FIGURE 4.2: Reference architecture with block diagram of the DDPG algorithm.

As for the specific RAN slicing enforcement strategy, each slice is assigned a given radio resource pool across a cluster of interfering cells in a given service area. The number of RBs is dynamically determined and requested by the TNT on the base of the pieces of information it has access to.

This scenario can be described as a discrete-time stochastic control process modelling a classical MDP, where the cellular system is the *environment*, whose *state* $\mathcal{S}$ is represented by the radio conditions of the nodes and the amount of incoming traffic, the *reward* R is the efficiency of resource utilization subject to QoS constraints and the *set of actions* $\mathcal{A}$ is the bandwidth allocated to mission critical slices. However, the optimal RRM solution cannot be known because of the non-convex nature of the problem and for the fact that the TNTs have only partial knowledge of the underlying RAN information. Indeed, it is worth noting that the details of the radio interface, e.g., the adopted numerology, the scheduling policy, the packet fragmentation rules, and so on, are fully in charge of the IP and are not known by the TNT agents, which have only a limited knowledge of the radio link conditions of their users [98], [226].

Besides, the reward that is of interest for the TNT is often a QoS parameter, e.g., the latency and the packet loss ratio for mission-critical users, whose relationship with the allocation decision, e.g., the amount of allocated spectrum, is very hard to establish. Moreover, $P^{a}_{s,s'}$, which is the transition probability from the state s to the state s' given the action a and $R(s,a)$, i.e., the average reward R in the state s given the action a, are unknown since they depend from the cellular environment, whose dynamic is not predictable.

In this context, RL emerges as the perfect tool to address the RAN slicing problem. Nevertheless, the dimensions of states and actions are huge or possibly infinite, therefore Q-learning approaches are uneffective. As a consequence, one of the most effective way to deal with the problem is through model free RL and, in particular, with function approximation of the action value function $Q(s,a)$ given by neural networks [225].

As illustrated in Figure 4.2, the TNT agent is trained with the DDPG algorithm, which is known to be suitable for dealing with continuous states and actions [225], [227]. It is an actor-critic, model-free, off-policy DRL method which computes an optimal policy that maximizes the long-term reward [57], [228]. Thanks to an actor-critic method, a DDPG agent concurrently learns a Q-function and an optimal policy that maximizes the long-term reward. Here, the idea is to evaluate the Q function through an approximation $\hat{Q}(s,a|\theta^{Q})$ and to represent the policy through another approximation

$(s|\theta\ )$. In particular, θ and θ^Q are the parameters of the actor and critic neural networks, respectively. In addition, two copies of the actor and critic, that is the target networks, are used to improve the stability of learning the action-value function, since target values are constrained to change slowly. The target critic is identified by $\hat{Q}'(s,a)$ and $\theta^{Q'}$, while $'(s)$ and θ' are related to the target actor.

The update of the actor and critic networks occurs with the gradient descent method. Specifically, the critic's θ^Q is updated by minimizing the loss L:

$$L = \frac{1}{M}\sum_{i=1}^{M}\left(y_i - \hat{Q}(s_i, a_i|\theta^Q)\right)^2, \tag{4.1}$$

where M is the number of experiences sampled from the experience replay (i.e., where the agent stores each of its experiences during training) [229], $y_i = R_i + \gamma\hat{Q}'(s_i,\ '(s_i'|\theta\ )|\theta^{Q'})$ is the Q function target approximated through bootstrapping [225], γ is the future reward discount factor [225], and s_i' represents the next observation.

In turn, the actor's θ is updated by following the sampled policy gradient to maximize the expected discounted reward:

$$\nabla_\theta\ J \approx \frac{1}{M}\sum_{i=1}^{M}\nabla\ _{(s_i)}\hat{Q}\left(s_i,\ \ (s_i)|\theta^Q\right)\nabla_\theta\ \ (s_i|\theta\ ), \tag{4.2}$$

where J is the environment start distribution as defined in the *policy gradient theorem* [225].

The *action* $a \in \mathcal{A}$ is the amount of bandwidth requested every allocation period to the IP. The *state* $\mathbf{s} \in \mathcal{S}$ is a vector of some Key Performance Indicators related to the RAN as well as traffic information. The state can be either directly computed by the TNT (e.g., the agglomerated slice traffic) or communicated by the IP and is used to determine the amount of bandwidth requested for the next period. The *reward* $R(s,a)$ takes into account the amount of bandwidth the TNT saves with respect to the maximum bandwidth as well as some other QoS indicators. The TNT action is thus dynamically chosen on the base of the available observations (state) with the goal of maximizing a discounted average future reward. In particular, $a \in [0.1, 0.9]$, i.e., the *action* is a continuous value between 10% and 90% of the maximum bandwidth allocated to the TNT. The *state* is defined as:

$$\mathbf{s} = (l, r, d, o) \tag{4.3}$$

where l is the total per-slice agglomerated traffic to be sent, r is the average rate of the user experiencing the worst channel conditions (averaged over the allocation period), d is the maximum delay experienced by the users of the slice, and o represents the number of QoS outages happened in the episode. The *reward* R is computed as: $R = 1 - a$, if the QoS requirement is satisfied, or $R = -1$ otherwise. In other words, the less the bandwidth requested by the TNT while satisfying the target QoS requirement, the higher the reward. Note that the choice of the *reward* R in a RL problem is subject to empirical considerations: a good reward function should capture the essence of the problem.

4.3.2 Performance Evaluation

To evaluate the performance of the proposed model in a realistic environment, a specific use-case based on autonomous-driving is considerd. Since this work is focused on a latency-sensitive scenario, it is assumed that the TNT slice requests must be always accepted by the IP, i.e., neither an admission control nor a resource allocation negotiation policy is enforced. The service level agreement between the latency-sensitive TNT and the IP sets a maximum amount of bandwidth to be used in each cell and, by providing for a unitary cost associated with each bandwidth resource, enforces *pay for what you get* mechanisms to prevent from over-provisioning the TNT.

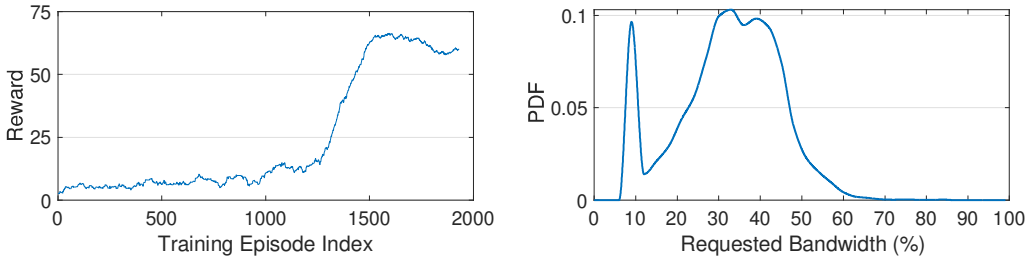
Without loss of generality, this work focuses on a single cell scenario, i.e., the effect of inter-cell interference is not taken into account. On the other hand, the proposed framework leverages on the capability of the DRL agent to predict the mutual interactions of the involved nodes in determining the actual system performance and, accordingly, it is naturally suitable to encompass a multi-cell scenario, provided that the state variables include some interference related parameters, e.g., the mutual position of the nodes.

Channel modeling considers the 3GPP UMa scenario [104], whose path loss and lognormal shadowing are implemented. The data rate of each active link is then derived based on the Shannon capacity formula. The work focuses in the following on the downlink case. Similar considerations and results can be obtained for the uplink case.

The scenario contains one single macro Base Station and a single TNT subsystem, which is assumed to provide autonomous driving services. In

the considered setting, vehicles use their own sensors (e.g., HD camera, LiDAR), as well as sensor information from other vehicles, to perceive the environment and obtain a 3D model of the world around them. The main QoS requirement of the slice is a maximum experienced packet delay of 5 ms, which is half the maximum value of latency envisioned for the High Definition Sensor Sharing, which is one of the main Autonomous Driving use cases [230]. The packet length is assumed to be fixed and equal to 32 bytes (as per the ITU-IMT2020 Urban Macro–URLLC usage scenario). The number of slice subscribers, i.e. the autonomous vehicles, is modeled according to real mobility traces[1] The TNT is allocated a maximum bandwidth of 10 MHz, organized into slots of 1 ms, according to the 5G NR numerology with $\Delta f = 15$ KHz. The MAC scheduling strategy enforced by the IP is the Throughput to Average scheduling, in order to guarantee a minimum level of service to every user, hence reaching a high fairness index. The DDPG agent performs its actions every allocation period of 1 s.

Figure 4.3(A) shows the running average (with a window length of 100 episodes) of the reward during the training process of the agent. The figure shows that the proposed DRL approach allows to converge to a bandwidth occupancy of around 35% (65% of bandwidth left to other usages). During an initial exploration phase, the agent is not able to address the QoS requirement, hence the low reward. After approximately 1200 training episodes, rewards begin to grow, since the algorithm successfully learned how to satisfy the latency constraint.



(A) (B)

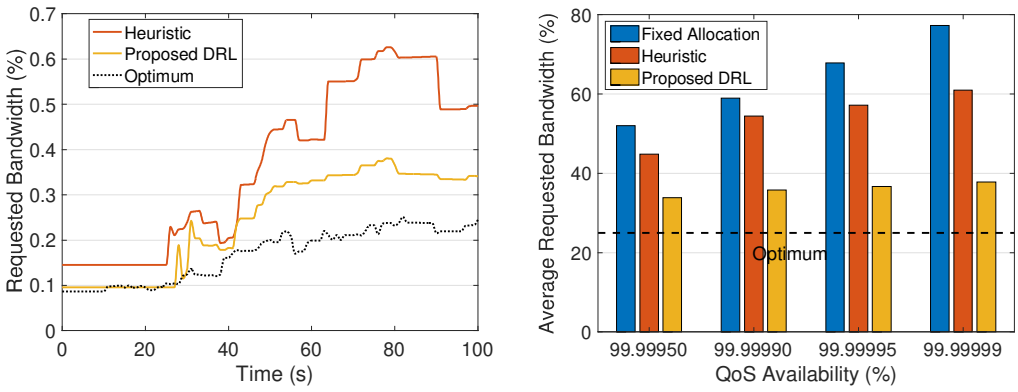
FIGURE 4.3: (a) Reward during the training phase of the agent. (b) Probability Density Function of the actions taken by the agent.

[1]M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser, "CRAWDAD dataset epfl/mobility (v. 2009-02-24)", https://doi.org/10.15783/C7J010, Feb 2009.

Figure 4.3(B) shows the probability density function of the bandwidth requested by the DRL agent of the TNT. Samples are related to 10000 independent simulations. On the one hand, it demonstrates how the agent effectively learned to perform a variety of actions, i.e., it learned to dynamically adapt to the environment. On the other hand, it shows that the agent usually tries to request as low bandwidth as possible, hence indicating a well-engineered reward function. Please note that this figure and the following are obtained by running the agent obtained at the end of the learning phase over the dataset considered for the testing phase. In this way, it is possible to assess the capability of the proposed DRL approach to generalize the proposed control strategy to every possible data traffic and radio channel conditions.

To better demonstrate the importance of DRL, the simulation results are compared with the following methods: Fixed Allocation, in which the TNT requests always the same amount of bandwidth; *Heuristic strategy*, characterized by a perfect prediction (i.e., ideal) of the incoming traffic and a bandwidth request that is directly proportional to the incoming traffic at each step; *Optimum allocation*, in which at each step the minimum bandwidth allowing to fulfill the slice QoS requirements is determined through iterative adjustment. Clearly, this last approach is unfeasible in a real system, although it can be easily simulated.

Figure 4.4(a) shows the bandwidth requested by the TNT during a representative test episode.



(A) (B)

FIGURE 4.4: (a) Bandwidth requests in a representative test episode. (b) Average bandwidth satisfying a given QoS availability.

It clearly illustrates how the agent learned to request an amount of bandwidth close to the optimum, by taking into account only the state variables.

In other words, the agent is dynamically adapting to the changes occurring in the environment. Furthermore, it is of the utmost importance to highlight how the proposed DRL solution outperforms the heuristic approach. In other words, even though the prediction of the incoming traffic is accurate, it is not sufficient to guarantee an optimal bandwidth request. As a matter of fact, it is necessary to take into account what actually happens in the RAN to accomplish such a decision. For instance, it is clear that the incoming traffic grows substantially after 60 s. However, it is reasonable to assume that general radio channel conditions improve as well, therefore it is not strictly necessary to claim more bandwidth.

Figure 4.4(b) shows the bandwidth requested by the TNT to ensure a certain level of QoS availability, i.e., the probability associated with the main QoS requirement being satisfied. Specifically, the actions taken by both the trained DRL agent and the heuristic are successively weighted to obtained different behaviors. The results are then averaged over 10000 independent simulations. The most noticeable feature is that the proposed DRL mechanism always outperforms the other strategies. Even though requested bandwidth always grows with more stringent requirements on the QoS Availability probability, the DRL agent requests up to 50% less bandwidth compared to the fixed allocation. Moreover, the variation of the bandwidths requested by the TNT DRL agent are incredibly smaller, confirming how the agent learned a near-optimal allocation strategy starting from the limited information available.

4.4 TNT-Driven RAN Slicing Enforcement based on Pervasive Intelligence

With the ever-increasing network complexity due to resource sharing among multiple entities, it would be essential to pervasively adopt AI [197], [231]. The advent of programmable networks and network virtualization, as well as the easier large-scale data acquisition, indeed promoted the allocation, management, and orchestration of network resources through AI techniques [54], [57], [64], [197], [199]. Most of the contributions in this field, which employs AI-based methods for channel estimation and to manage network slicing in the core network and RAN, proposes centralized solutions based on Deep [200]–[202] and RL/DRL [193], [203]–[215], where the network status is fully observable. However, as it should be clear by now, in the business vision of

TABLE 4.1: Comparison among this work and the other contributions adopting AI-based techniques for the management of network slicing.

Contributions	AI-based techniques		Network Slicing			
	DL	RL/ DRL	Core Network	RAN	TNT-driven	Network status compression
[200], [201]						
[202]						
[193], [203]–[212]						
[213]–[215]						
[50]						
This work						

network slicing, TNTs are decoupled from the IP and they can only have a partial vision of the network status to preserve privacy [37], [46], [48]. The current state-of-the-art approaches do not handle these aspects, by presenting slicing enforcement schemes driven directly by the IP. To bridge this gap, this Section proposes a TNT-driven RAN slicing enforcement scheme based on Pervasive Intelligence, that can be fully implemented in the business and privacy-oriented vision of network slicing. According to the Pervasive Intelligence paradigm and starting from the outcomes of the preliminary contribution presented in the previous Section, both IP and TNTs exploit AI to accomplish their tasks. This is perfectly in line with the pervasive intelligent endogenous design of future generations of mobile networks [197]. Specifically, the IP exploits a DL method based on a convolutional autoencoder, which compresses the information on network resources and connectivity and shares the actual (but hidden) network status with the TNTs by preserving privacy. In turn, each TNT exploits the resulting hidden knowledge of the network status in a DRL agent based on the DDPG algorithm in order to dynamically adapt bandwidth requests according to its own users' requirements. Finally, the IP employs the outcomes of the DDPG algorithm to effectively enforce the network slice in the RAN. Thus, even if each TNT does not fully know network resources and conditions information, the bandwidth requested for offering services and respecting a given QoS constraint (i.e., target Service Availability) could be optimally allocated according to the *Pay for What You Get* paradigm: the less the bandwidth, the more the TNT savings, while avoiding the radio resources over-provisioning.

Table 4.1 summarizes the goals and methodologies followed by the reviewed contributions adopting AI-based techniques for the management of

network slicing. It emerges that no contributions in the current scientific literature jointly exploit DL and DRL for a TNT-driven RAN slicing enforcement scheme, as proposed in this work in order to dynamically adapt bandwidth requests according to users' requirements of TNTs, without fully knowing the network status.

The efficiency of the devised TNT-driven RAN slicing enforcement scheme based on Pervasive Intelligence is investigated for the eMBB and Remote Driving use cases, by using computer simulations with real and conceivable network and QoS settings in compliance with ITU and 5G specifications [102], [232], [233]. The comparison with conventional resource allocation methods, corresponding to the optimal, random, and dynamic (i.e., proportional to the users' requests) allocation of bandwidth, demonstrates that the proposed approach ensures the best trade-off between bandwidth savings and bandwidth over-provisioning, while always ensuring the target Service Availability.

4.4.1 The Proposed Scheme

The IP has a comprehensive view of the network and it can access data not natively accessible for TNTs. As already said, indeed, the IP subsystem is the only entity able to retrieve information from the RAN. The methodology presented here assumes that the TNT subsystem may control its slice based on the information carried out by CQI feedbacks. However, it is not reasonable to suppose that the IP subsystem forwards all the collected CQI feedbacks to each TNT subsystem. Otherwise, privacy requirements and business roles of the IP and TNTs would be compromised, and the communication overhead at the network edge would be unnecessarily high [201]. To solve these issues, according to the Pervasive Intelligence paradigm, the IP subsystem processes the collected CQI feedbacks through DL and exposes a compressed vision of the RAN situation to the TNT subsystems. This task is performed through an autoencoder and represents one of the main novel ideas presented in this work. Specifically, by discarding irrelevant information and reducing the dimensionality of data, the autoencoder is used to generate a feature learning representation of CQI feedbacks, without requiring the knowledge of data distribution nor the explicit identification of a certain structure [234]. As a result, by coding and compressing the CQI information, it is possible to preserve privacy (because TNTs cannot reconstruct original CQI indexes) and to limit network complexity (because of reduced information exchanged with

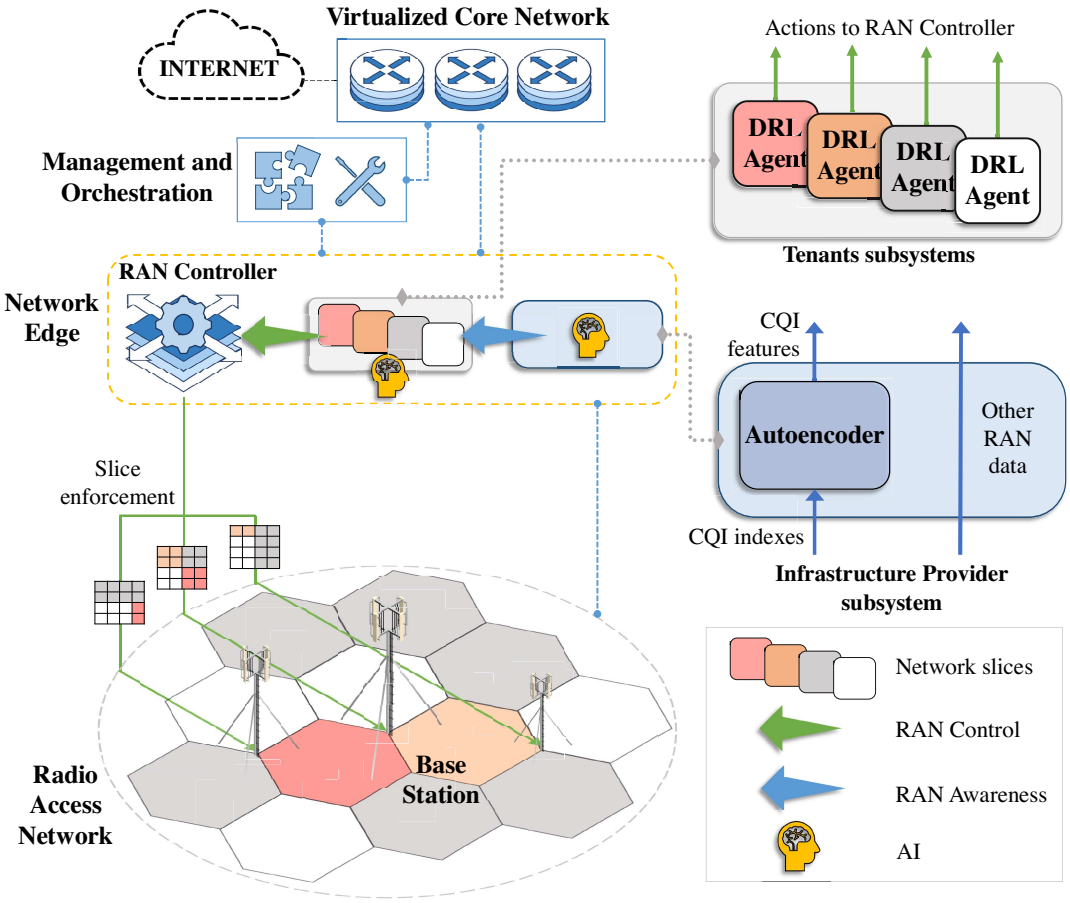


FIGURE 4.5: The reference architecture.

TNTs subsystems). Indeed, the IP subsystem sends the hidden and compressed CQI feedbacks to the TNT subsystem.

Then, the TNT subsystem further processes the received data (also in this case, through specific AI algorithms falling into DRL, as discussed hereafter) and supplies instructions for the successful handling of its RAN slice. It is important to remark that the TNT subsystem cannot manage RAN slices directly. However, any action is controlled (first) and implemented (then) by the IP. For this reason, the TNT subsystem sends the aforementioned instructions to the RAN controller, which decides to accept/deny them, allocates RAN resources to the slice and enforces the slicing policy on the available spectrum. The requests that the TNT subsystem may submit to the RAN controller include bandwidth allocation, variation of radio resource scheduling algorithms, HARQ configurations, channel coding schemes, power control strategies, multicast/broadcast activation, or beam management [135], [235]. Also, these requests must be issued in real-time, to successfully meet end-users requirements under the current RAN conditions.

The conceived slice enforcement strategy allows the TNT subsystem to estimate, in real-time and slot by slot, the number of radio resources to allocate to the controlled slice. Thanks to the *Pay for What You Get* paradigm, only the required amount of bandwidth is allocated so that the radio resources over-provisioning is avoided. At the same time, however, it is requested that allocation decisions must be executed instantaneously to reduce communication latency and avoidable expenses [48]. Accordingly, the developed solution definitively employs DRL for supporting the slice enforcement strategy. TNTs subsystems act as DRL agents that process the features extracted by the autoencoder (and provided by the IP subsystem, as illustrated before), and optimize their actions. Since DRL provides autonomous decision-making, the resulting system also ensures a high scalability level. Indeed, TNTs subsystems can make observations and obtain the best policy locally without exchanging information among each other. This reduces communication overheads and also improves the security and robustness of the networks [57].

4.4.1.1 Design of the Autoencoder used by the IP subsystem

The autoencoder is a particular Artificial Neural Network (ANN) implementing two key functionalities: the encoder generates the corresponding feature learning representation of input data, while the decoder provides a reconstruction of the input data, starting from the aforementioned feature learning representation.

The input data are spatial snapshots (i.e., matrices) related to the CQI indexes of mobile users, namely $\mathbf{Y} \in \mathbb{R}^{K \times L}$, where K and L are the chosen numbers of rows and columns of the snapshot, respectively. Note that K and L are design parameters. As depicted in Figure 4.6, the investigated encoder is made of three chained 2-dimensional convolutional layers to extract spatial correlations of CQI indexes snapshots [236].

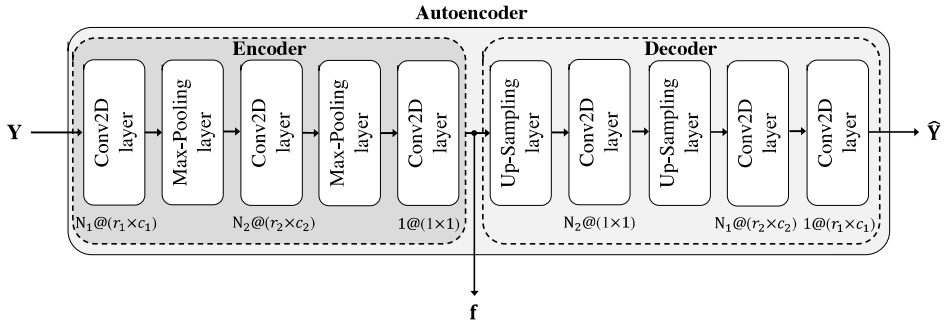


FIGURE 4.6: Architecture of the adopted convolutional autoencoder.

Each layer comprises a set of filters which are convolved with the CQI indexes snapshot for extracting the features of a certain input region and with the Rectified Linear Unit (ReLU) activation function. Then, two pooling layers follow each convolutional layer to perform down-sampling (i.e., max-pooling picks the maximum value) of intermediate representations, for complexity reduction and overfitting mitigation [54], [237]. The first and the second convolutional layers use N_1 filters $(r_1 \times c_1)$ and N_2 filters $(r_2 \times c_2)$, respectively. Note that N_1 and N_2 represent the number of filters, while $(r_1 \times c_1)$ and $(r_2 \times c_2)$ describe the dimensions of filters, where r_1, c_1 and r_2, c_2 represent the number of rows and columns of the first and the second convolutional layers. In addition, the filters can have diverse strides $[v_1\ h_1]$ and $[v_2\ h_2]$ (where v_1, v_2 and h_1, h_2 represent the vertical and the horizontal step size for the first and the second convolutional layers). Then, a channel-wise normalization with ch_1 channels and ch_2 channels per element is performed for the first and the second convolutional layers, respectively. Indeed a typical operation in CNN is channel normalization for rescaling each channel (whose number determines the depth of the snapshot) into the range of [0,1] thus avoiding vanishing gradients [238]. By receiving as input the snapshot of CQI indexes $\mathbf{Y} \in \mathbb{R}^{K \times L}$, the encoder generates the corresponding feature learning representation vector, namely $\mathbf{f} \in \mathbb{R}^{F}$, with F depending on $(K, L, r_1, c_1, r_2, c_2)$. The output of the encoder, i.e., the features $\mathbf{f}$ extracted for each input snapshot, is obtained by the third convolutional layer with 1 filter (1×1) and then it is given to TNT subsystems as input for the DRL agents.

Finally, the decoder provides the reconstruction of the CQI indexes, namely $\hat{\mathbf{Y}} \in \mathbb{R}^{K \times L}$, starting from the aforementioned feature learning representation $\mathbf{f}$. By going backwards to input reconstruction, the decoder makes use of two up-sampling layers, corresponding to the two max-pooling layers in the encoder [54], [237], and three convolutional layers, with the ReLU activation function, except for the output layer, which uses the sigmoid activation function [54]. The convolutional layers employ N_2 filters (1×1), N_1 filters $(r_2 \times c_2)$, and 1 filter $(r_1 \times c_1)$, respectively.

All the CQI indexes stored in $\mathbf{Y}$ are normalized within the range [0,1] to accelerate the training convergence [239]. The autoencoder uses weights that are properly configured during the training phase and iteratively updated for each mini-batch of the dataset in order to minimize the Mean Square Error

(MSE) loss function. Formally, the MSE loss function is defined as [236], [240]:

$$MSE = \frac{1}{B}\sum_{b=1}^{B}\sum_{k=1}^{K}\sum_{l=1}^{L}\left(\hat{y}_{b,k,l} - y_{b,k,l}\right)^2, \tag{4.4}$$

where B represents the mini-batch size, $y_{b,k,l} \in \mathbf{Y_b}$, and $\hat{y}_{b,k,l} \in \hat{\mathbf{Y}}_\mathbf{b}$.

The encoder is the key building block of the presented DL architecture because it generates the compressed CQI indexes (i.e., the CQI features $\mathbf{f}$) [234] to be shared with the DRL framework. The decoder, instead, is only used herein to reconstruct the CQI indexes and to evaluate the performance of the designed encoder. Other than this analysis, it will not be employed by the DRL framework.

4.4.1.2 Design of the DRL Agents used by the TNT subsystems

As already mentioned, policies based on *Pay for What You Get* paradigm are used by the IP to prevent over-provisioning a TNT. In other words, the IP associates a unitary cost to each bandwidth resource and determines a maximum amount of bandwidth to use in each cell. The role of the DRL agent of each TNT subsystem is to reserve the minimum amount of bandwidth in each cell to satisfy its QoS requirements, so as to avoid resource over-provisioning. Therefore, the TNT subsystem places its bandwidth allocation requests expressed as a fraction of the maximum available bandwidth within a fixed allocation period.

The *action* $a \in \mathcal{A}$ is the ratio between the amount of bandwidth the TNT requests to the IP every allocation period and the maximum allowable bandwidth. It is a continuous value between 0 and 1 (i.e., 0% and 100% of the bandwidth made available to the TNT).

The *state* $\mathbf{s} \in \mathcal{S}$ is a vector defined as in the following:

$$\mathbf{s} = (\mathbf{u}, \mathbf{f}, \sigma) \tag{4.5}$$

where $\mathbf{u} \in \mathbb{N}^W$ is the number of users per sector (W is the number of cell sectors in the system, i.e., portions of the cell served by one of the W co-located base stations), $\mathbf{f} \in \mathbb{R}^F$ represents the feature learning representation, and $\sigma \in \mathbb{R}$ is the communication Service Availability throughout each episode. In more detail, the component of the feature learning representation $\mathbf{f}_i, \forall i = 1, 2, ..., F$ are the features on radio channel conditions extracted by the

autoencoder. The communication Service Availability σ is defined as the percentage value of the amount of time the TNT service is delivered according to the agreed QoS, divided by the amount of time the TNT is expected to deliver the service [232].

In this study, the reward should take into account the amount of bandwidth the TNT subsystem saves with respect to the maximum bandwidth B, provided that the QoS constraints can be satisfied. To elaborate, the *reward* R is computed as:

$$R = \begin{cases} 1 - a, & \text{if the target Service Availability } \sigma \text{ is guaranteed;} \\ -1, & \text{otherwise.} \end{cases} \tag{4.6}$$

Thus, the less the bandwidth requested by the TNT, the higher the reward. The strategy is adopted to terminate the training episode when $R = -1$, i.e., as soon as the TNT subsystem is not providing the service with the target availability.

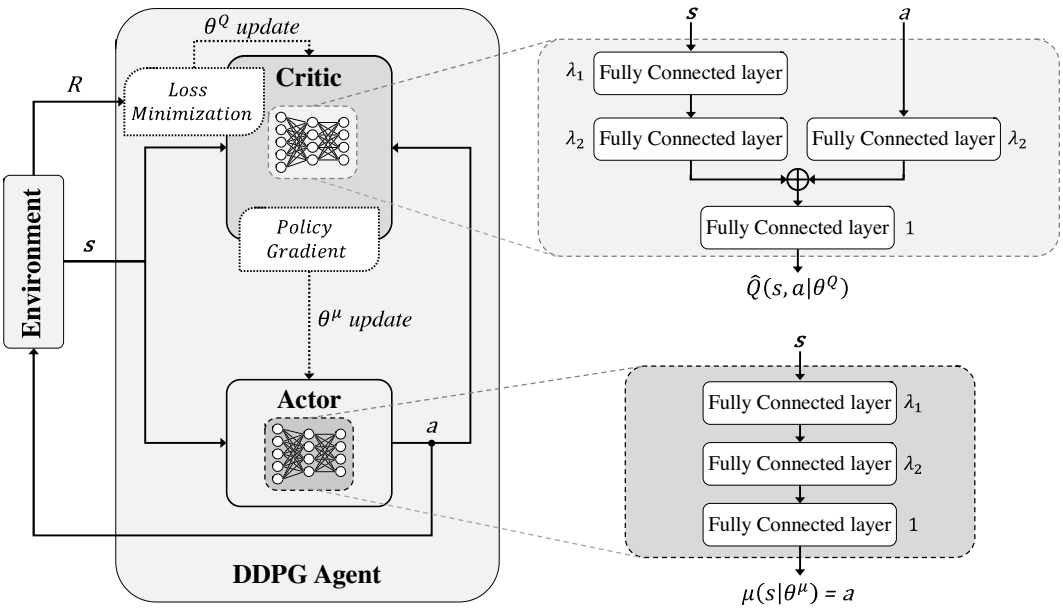


FIGURE 4.7: Architecture of the adopted DDPG algorithm.

Following the consideration reported in the previous Section, the DDPG algorithm, is considered, as illustrated in Figure 4.7. To further elaborate, the *state* **s** passes through the first and second fully-connected layers of critic and actor neural networks with λ_1 and λ_2 neurons, respectively, and ReLU activation function. Then, as shown in the right part of Figure 4.7, the actor network provides the *action* $(s|\theta\ ) = a$ as output by using a fully-connected layer with 1 neuron and hyperbolic tangent (i.e., tanh) activation function.

The *action a* is also received as input by the critic network and it passes through a fully-connected layer with λ_2 neurons. After adding the processed state, the *expected cumulative long-term reward* $\hat{Q}(s, a|\theta^Q)$ is obtained by the critic network through a fully-connected layer with 1 neuron and ReLU activation function.

4.4.2 Performance Evaluation

The performance of the conceived TNT-driven RAN slicing enforcement scheme based on Pervasive Intelligence is evaluated through computer simulations. To this aim, a system-level simulator of a mobile system is developed in MATLAB, based on the ITU's methodology recommendation [102]. The tool specifically models the downlink transmission. However, similar considerations and results can be obtained for the uplink case. A given number of base stations is placed in a regular grid, following a hexagonal layout. All cell sites consist of 3 sectors, where a configurable number of UEs, is dropped independently with a uniform distribution. The UEs, which have a fixed and identical speed with a randomly distributed direction, are attached to the base station able to ensure the highest SINR. All the links between base stations and UEs in the system are simulated with dynamic channel properties, taking into account the wrap-around configuration in the network layout. The implemented channel modeling considers inter-site interference and large-scale parameters, i.e., path loss, shadow fading, and LOS/NLOS propagation condition, according to the ITU guidelines [102] (please see Chapter 2 for further details on channel modeling).

The shadow fading is modeled as a log-normal random variable, with standard deviation set to 4 dB and 6 dB for LOS and NLOS propagation conditions, respectively. At the application level, the full-buffer traffic model (where the queue depths are assumed to be infinite) is implemented. The user-experienced data rate is derived through the Shannon theorem. Finally, the MAC scheduling strategy enforced by IP's base stations is Round Robin.

To evaluate the compliance of the developed simulator with 3GPP specifications, Figure 4.8 shows the ECDF of the wideband SINR experienced by the mobile users adopting the developed MATLAB simulator. The reported curve demonstrates that the simulator is well-calibrated according to the 3GPP Phase 1 NR MIMO system-level calibration for multi-antenna systems [241].

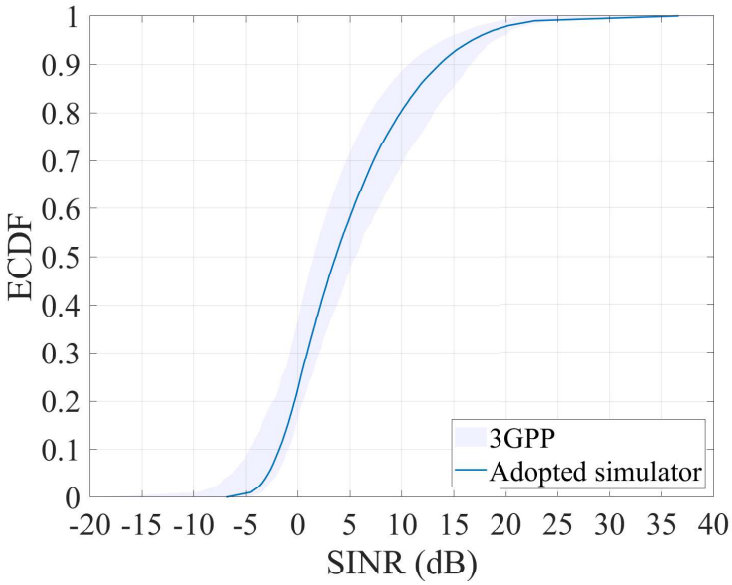


FIGURE 4.8: ECDF of the wideband SINR of the developed simulator with respect to 3GPP Phase 1 dense-urban (macro-layer) system-level calibration for multi-antenna systems.

Without loss of generality, the TNT subsystems are assumed to operate two types of network slices, i.e., eMBB and Remote Driving slices, with real and conceivable network and service settings. Of course, the whole scheme can be applied to each type of slice and scenario by properly adapting the related parameter settings.

For the eMBB scenario, the speed and density of UEs are set to 3 km/h [102] and 2000 UE/km^2 [232], respectively, and the downlink target rate and the Service Availability are set to $\mathcal{T} = 50$ Mbps [232] and $\sigma = 90\%$, respectively, according to the ITU Dense Urban eMBB deployment [102]. For the Remote Driving scenario, in line with the ITU Urban Macro URLLC deployment [102], the speed and density of UEs are set to 30 km/h [102] and 1200 UE/km^2 [230], respectively, and the downlink target rate and the Service Availability are set to $\mathcal{T} = 400$ kbps [230] and $\sigma = 99\%$, respectively. Then, the full-buffer traffic model is implemented for both the scenarios [102]. Table 4.2 summarizes the parameter settings.

4.4.2.1 Performance of the Autoencoder used by the IP subsystem

The autoencoder, used by the IP subsystem to compress the network status, leverages data related to the radio channel conditions. Specifically, a dataset generated by the implemented MATLAB simulator is used. For both

TABLE 4.2: Scenarios.

	eMBB	**Remote Driving**
Network deployment	ITU Dense Urban eMBB [102]	ITU Urban Macro URLLC [102]
UE speed	3 km/h [102]	30 km/h [102]
UE density	2000 UE/km^2 [232]	1200 UE/km^2 [230]
Downlink target rate $\mathcal{T}$	50 Mbps [232]	400 kbps [230]
Service Availability σ	90%	99%
Traffic model	Full-buffer [102]	

TABLE 4.3: Performance of the different configurations of convolutional autoencoders.

Configuration											**Performance**	
Filters										**Ch.-wise norm.**	**Training RMSE**	**N. of trainable param.**
Number		Dimension				Stride						
N_1	N_2	r_1	c_1	r_2	c_2	v_1	h_1	v_2	h_2	ch_1		
200	100	3	3	1	3	4	4	1	1	2	1.2012	123202
200	100	3	2	1	6	4	4	1	1	2	1.2032	243202
200	100	3	2	1	5	4	2	3	2	2	0.1277	203202
200	100	3	2	1	5	4	2	3	2	3	0.1384	203202
300	150	3	2	1	5	4	2	3	2	2	0.1295	454802
400	200	3	3	1	5	3	3	1	1	2	0.1237	808802
200	**100**	**3**	**3**	**1**	**5**	**3**	**3**	**1**	**1**	**2**	**0.1188**	**204402**
200	100	3	3	1	5	3	3	1	1	3	0.1195	204402

the scenarios reported in Table 4.2, the adopted dataset consists of 10000 realizations reporting the CQI indexes. Each realization is a snapshot with $K \times L = 3 \times 30 = 90$ CQI values for each base station: if the number of attached UE is greater than 90, only the worst 90 values are included; if the number of attached UE is less than 90, an appropriate padding is performed.

Different configurations of convolutional autoencoders, characterized by different values of parameters listed in Section 4.4.1.1, are investigated for identifying the suitable configuration to be used in the DRL framework. In particular, different numbers N_1 and N_2, dimensions $r_1 \times c_1$ and $r_2 \times c_2$, and strides $[v_1\ h_1]$ and $[v_2\ h_2]$ of filters for the first and the second convolutional layer are analyzed (please see Table 4.3 for further details). Also, the channel-wise normalization is performed with diverse ch_1 channels for the first convolutional layer, while ch_2 for the second convolutional layer is omitted because it is always set equal to 1. Note that the dimension of the feature learning representation vector $\mathbf{f}$ is the same for all the configurations in Table 4.3, that is $F = 6$.

The training set, whose performance metrics are listed and evaluated in Table 4.2, the validation set, and the test set consist of 70%, 15%, and 15% of the adopted dataset, respectively. The training phase, during which weights are iteratively updated in order to minimize the MSE loss function, is done with 100 epochs (i.e., complete passes through the training data [242] such that each example has been seen once) for all the evaluated configurations of convolutional autoencoders. The Adam optimization [243], with a learning rate equal to 0.01, is used to iteratively update the network weights. The performance is investigated in terms of training Root Mean Square Error (RMSE) and the number of trainable parameters. The RMSE represents the root of the MSE, as defined in (4.4), and allows a better understanding of resulting values. The number of trainable parameters measures the complexity of selected learning architectures: the higher the number of parameters, the higher the complexity level. Note that the RMSE gives the reconstruction performance of the whole autoencoder, even if the CQI reconstruction is not the focus of this work. However, if an autoencoder is able to well reconstruct the input, it means that its single blocks (i.e., encoder and decoder) have high performance.

Obtained results are reported in Table 4.3 for all the evaluated configurations of convolutional autoencoders. The second-last configuration, which is highlighted in Table 4.3, represents a good trade-off of performed loss and complexity. As a consequence, the rest of the presented work considers this configuration as the best one of the proposed autoencoder used by the IP subsystem. Then, its compressed CQI feature learning representation is passed to the DRL agents employed by the TNT subsystems.

Once the best autoencoder configuration is selected, the convergence analysis evaluates the performance of the DL architecture as a function of the number of epochs considered during the training phase. Figure 4.9 shows the autoencoder loss as a function of the number of epochs for the training set and the validation set. The reported curves confirm that the selected convolutional autoencoder fastly converges to stable values, without underfitting nor overfitting after the training phase, and does not need a long training period.

Finally, Figure 4.10 reports the reconstruction errors on the test set with the relative frequency.

It is evident that the selected configuration of the convolutional autoencoder reconstructs data with very high accuracy during the test phase. In fact, the reconstruction with an overestimation/underestimation of more than 2

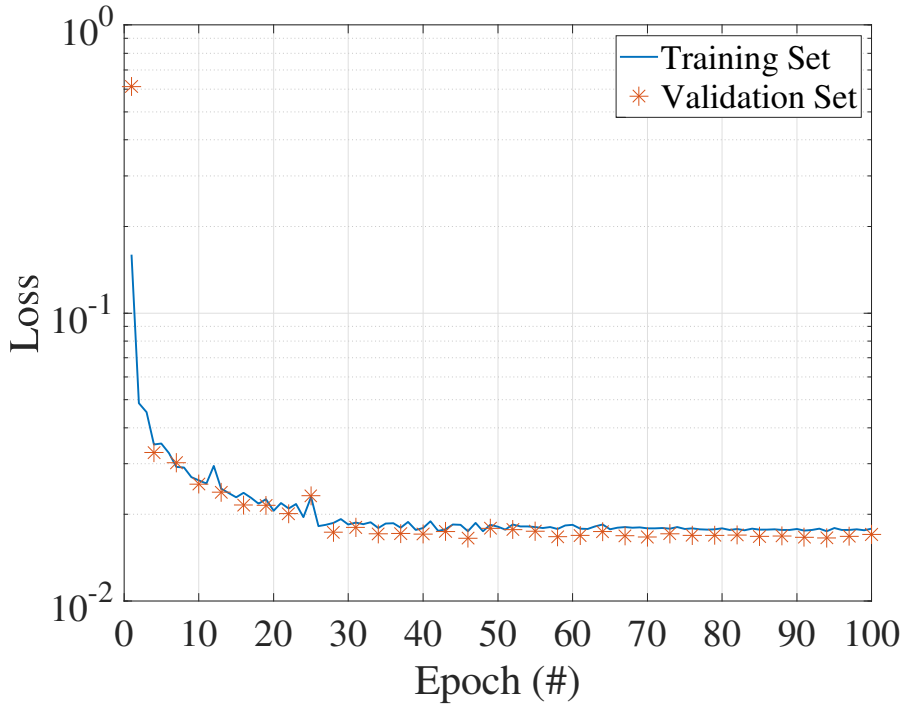


FIGURE 4.9: Autoencoder loss (i.e., MSE) vs number of epochs.

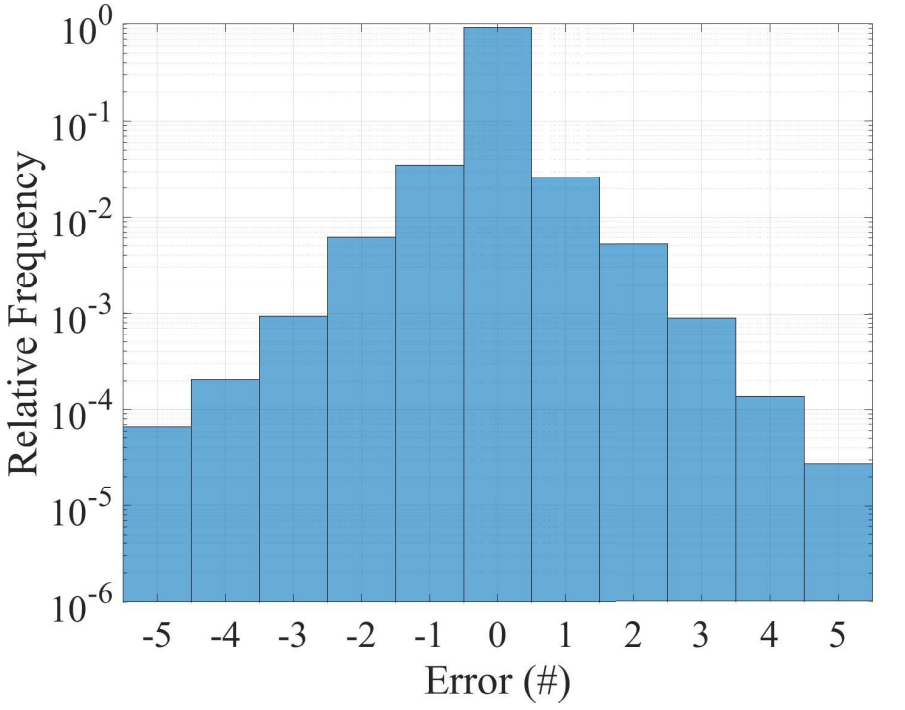


FIGURE 4.10: Relative frequency of the reconstruction errors on the test set.

CQI indexes occurs with a relative frequency always lower than 0.01.

4.4.2.2 Performance of the DRL Agents used by the TNT subsystems

The performance of the DRL agents used by the TNT subsystems is evaluated through the calibrated simulator. Specifically, a DDPG algorithm is implemented by TNT subsystems. As anticipated, two different TNT subsystems are taken into account. They are assumed to provide eMBB services and Remote Driving services, whose target Downlink rate and Service Availability are reported in Table 4.2). Each episode lasts 1 s, i.e., the DRL agent performs its actions every second. Each TNT subsystem may request a maximum bandwidth B of 100 MHz for every considered sector. As for the actor and critic neural networks, the learning rate is set equal to 0.001 and 0.0001, respectively; the number of neurons is set to $\lambda_1 = 2000$ and $\lambda_2 = 1500$.

The number of training epochs, each corresponding to 100 training episodes, is set equal to 50.

Figure 4.11 shows the achieved reward as a function of the number of training epochs for the two analyzed scenarios. Each point is the average reward obtained during the related epoch (i.e., 1 epoch = 100 episodes).

The reported curves confirm that both the DRL agents fastly learn the policy from the state: the average reward in eMBB and Remote Driving scenarios converges to stable values after 20 and 15 training epochs, respectively. Thus, 50 training epochs are sufficient for convergence.

To deeply analyze the performance of the proposed approach, the proposed approach based on the DDPG algorithm is compared with different conventional resource allocation methods, which are implemented with the same parameter settings:

- *Genie*, that corresponds to the optimal allocation of bandwidth for each slice, i.e., the minimum amount of bandwidth that guarantees $\sigma = 100\%$ as Service Availability. It is important to note that the bandwidth, in this case, is determined through iterative adjustments during simulations. Therefore, the Genie approach is unfeasible in actual deployments;

- *Random*, i.e., the bandwidth allocated to each slice is randomly chosen between 10% and 90% of the maximum bandwidth B.

- *Heuristic*, which represents the dynamic allocation of bandwidth. Specifically, the bandwidth is proportional to the highest number of mobile users in a cell (that is a piece of information available in the

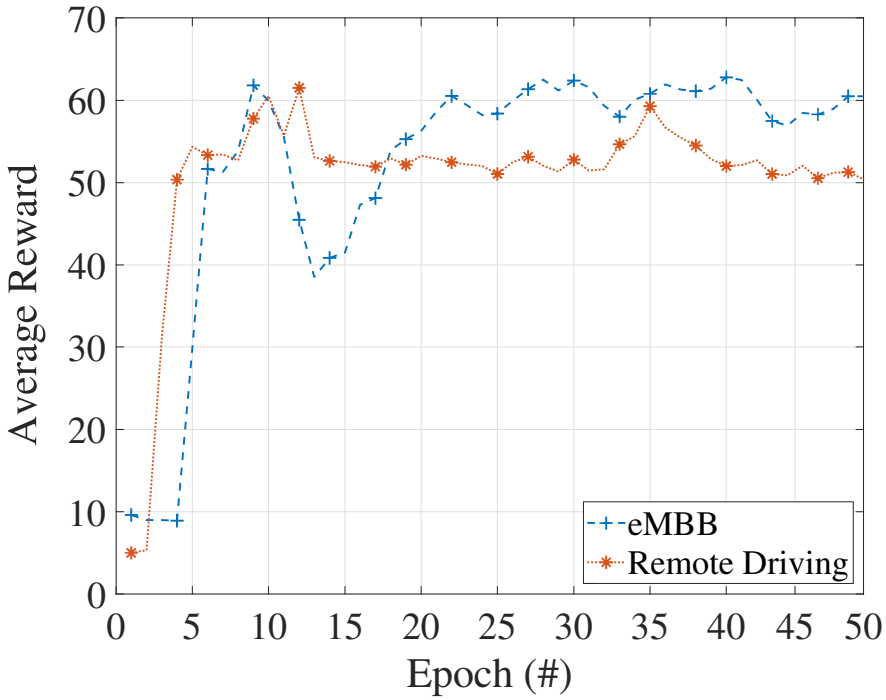


FIGURE 4.11: Average episode reward vs number of epoch (with 1 epoch corresponding to 100 training episodes) for eMBB and Remote Driving scenarios.

state **s**). In particular, for each scenario (i.e., eMBB and Remote Driving use cases), the action a is computed at each step according to the following:

$$a = \min\left\{\left(\max_{1\leq i\leq W} u_i \cdot \frac{\mathcal{T}}{B\log_2(1+SINR_p)}\right), 1\right\} \tag{4.7}$$

where u_i is the number of UEs in the $i-th$ sector, $\mathcal{T}$ is the downlink target rate, and $SINR_p$ is a specific percentile p of the SINR distribution. In the following, $p = 5\%$, namely the cell-edge SINR [244], and $p = 50\%$, namely the median SINR [244], are considered. Thus, the Heuristic approach allocates a bandwidth which is proportional to the maximum number of UEs per sector with two different choices for the proportionality factor: $p = 5\%$ represents a worst-case situation calibrated for mobile users at the cell edge, whereas ($p = 50\%$) is calibrated for median users.

The performance is investigated in terms of Episode Availability Indicator $\mathcal{E}$ and bandwidth saved with respect to the *Genie* (that represents the optimal bandwidth for 100% of communication Service Availability σ). The Episode

TABLE 4.4: Episode Availability Indicators $\mathcal{E}$ for the analyzed approaches

	$\mathcal{E}$ (%)	
	eMBB	**Remote Driving**
Random	74	0
Heuristic ($p = 5\%$)	100	32
Heuristic ($p = 50\%$)	100	0
DDPG	100	100

Availability Indicator $\mathcal{E}$ is defined as:

$$\mathcal{E} = \frac{1}{N_E} \sum_{i=1}^{N_E} \epsilon_i \cdot 100 \tag{4.8}$$

where N_E is the number of test episodes and ϵ_i is related to the target Service Availability σ, that is:

$$\epsilon_i = \begin{cases} 1, & \text{if the Service Availability } \sigma \text{ is guaranteed} \\ & \text{in the } i-th \text{ test episode, } \forall i; \\ 0, & \text{otherwise.} \end{cases} \tag{4.9}$$

Therefore, the Episode Availability Indicator $\mathcal{E}$ is the percentage value of the number of test episodes the service of the TNT subsystem is delivered according to the agreed Service Availability, divided by the total number of test episodes. Note that the total number of test episodes is set to 500.

Table 4.4 reports the Episode Availability Indicators $\mathcal{E}$ performed by the analyzed approaches for eMBB and Remote Driving scenario, respectively.

As a first observation, it is worth noting that the proposed approach based on the DDPG algorithm always guarantees the 100% of Episode Availability Indicator $\mathcal{E}$, i.e., it allows to always provide the service with 90% and 99% Service Availability for eMBB and Remote Driving cases, respectively. Specifically, in the eMBB scenario, the Episode Availability Indicator $\mathcal{E}$ equal to 100% can also be obtained by both the *Heuristic* approaches (with $p = 5\%$ and $p = 50\%$), while it is never achieved by the *Random* approach. In the Remote Driving scenario, only the proposed solution based on the DDPG algorithm has the Episode Availability Indicator $\mathcal{E}$ equal to 100%, which far exceeds those of the other approaches.

Figure 4.12 shows the percentage of bandwidth savings performed by the analyzed approaches for both scenarios. In the eMBB scenario, the proposed

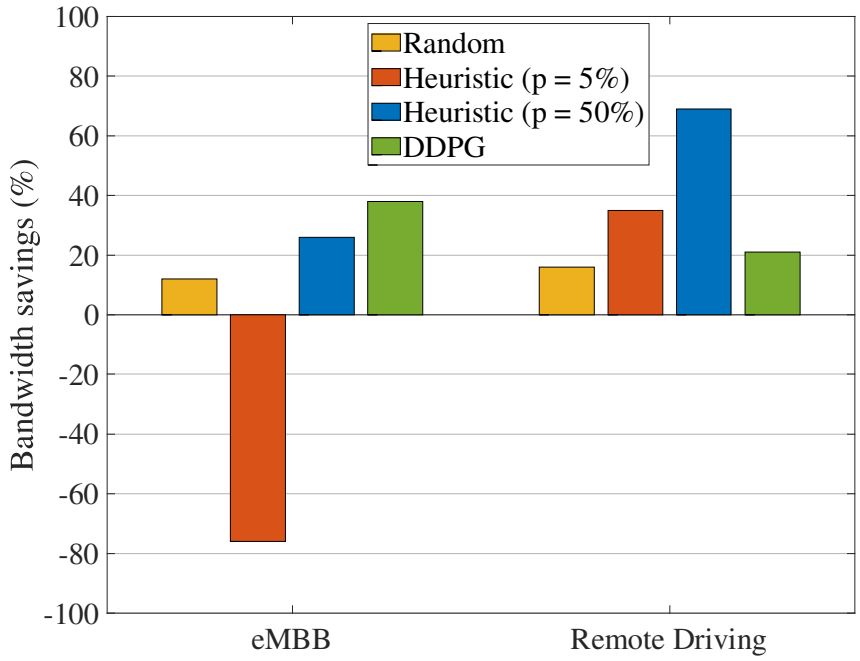


FIGURE 4.12: Comparison among different approaches with respect to the Genie in terms of bandwidth savings.

approach based on the DDPG algorithm saves the highest amount of bandwidth (i.e., around 40%) with respect to the other approaches. Note that, in this case, the *Heuristic* ($p = 5\%$) approach requires a greater amount of bandwidth than the *Genie*.

In the case of Remote Driving, the proposed approach based on the DDPG algorithm does not ensure the highest bandwidth saving: the bandwidth saving of the DDPG-based approach is the lowest one (i.e., 20%), except for the *Random* approach.

However, as anticipated, only the proposed solution based on the DDPG algorithm has the Episode Availability Indicator $\mathcal{E}$ equal to 100%. Thus, it can be considered as the winning approach also for this scenario.

To sum up, the DRL agents used by the TNT subsystems, which implement the DDPG algorithm, actually learn to save bandwidth, while always ensuring the Service Availability and avoiding the bandwidth overprovisioning in contrast to the *Genie*. Overall, the proposed approach outperforms other conventional strategies also because it can be intelligently and flexibly tuned on the required Service Availability of the TNT subsystem during training, as demonstrated by the results of both scenarios.

Thus, the bandwidth requested for offering services and respecting the target Service Availability could be optimally allocated according to the *Pay*

for What You Get paradigm.

4.5 RAN Slicing for Location-Aware V2I Communications: The Autonomous Tram Use Case

This Section envisages a RAN slicing mechanism to design a customized V2I communications service for mission-critical applications in a scenario in which the mobile users' location is available to the TNT. This study may help the design of a specific slice that can be associated with Slice/Service Type 4 i.e., V2X [245] in those situations where the service requirements are very stringent in terms of reliability and latency, thus enabling also URLLC. As a possible application scenario that falls in this general setting, the AT scenario is considered, where the trams' positions must be perfectly and timely acquired, e.g., leveraging the use of a MEC server located directly adjacent to the gNBs. In this scenario, the number of users (e.g., trams) is typically on the order of few or tens units, and, hence, the effective bandwidth needed to satisfy the QoS can be dynamically evaluated based on the knowledge of the position of each user rather than leveraging the prediction of agglomerated per-slice traffic, as done when dealing with general RAN slicing approaches, e.g., see [200], [202], [223]. In this way, it is possible to tailor the slice request and the correspondent slice enforcement to the actual needs of each user, thus minimizing the number of resources required for QoS fulfillment. The specificity of the considered scenario makes it substantially different from all the previously proposed RAN slicing mechanisms, whether they are related to vehicular communications or more general scenarios. In particular, compared with the related research works discussed in Section 4.1, the closest proposal appears to be [182], where the concept of linked bandwidth as an efficient strategy to limit the inter-slice interference is first explained. However, the knowledge of the position and the data traffic of each user allowed to elaborate a more detailed mechanism for allocating (as much as possible) the same frequency spectrum to non-interfering users and different portions of the spectrum to interfering users. This opportunity is not granted to any of the previously mentioned RAN slicing scenarios. Hence, to fully exploit the potential benefits brought by location information in creating the AT slice, the TNT is indeed responsible for allocating virtual resources to its users leveraging the knowledge of their exact positions and on some relevant RAN information. The AT TNT is then the entity in charge of guaranteeing

the required separation, isolation, independence of the AT slice with respect to the other slices.

The RAN slicing problem is mathematically formalized and a reasonably good heuristic approach for the allocation of virtual resources to the AT slice is also proposed. The slicing enforcement problem is evaluated in a realistic scenario making use of a customized version of 5G-air-simulator. Moreover, the simulator allows the establishment of the best dimensioning of the system to guarantee the required QoS in terms of throughput and latency with the minimum impact in terms of total resources required to enforce the AT slice.

Please note that although the proposed framework is tailored for the AT slice, it can be easily extended to any slicing scenario where TNT has precise knowledge of users' locations.

4.5.1 V2I communication requirements and the Autonomous Tram Use Case

Table 4.5 reports the main QoS requirements for V2I communications with specific references reported for each different service [246]) .

TABLE 4.5: QoS Requirements for V2I Communication Scenarios

Scenario	Max Latency	Reliability	Data Rate
Remote Driving [247]	5 ms	99.999%	20 Mbps
Teleoperated support [247]	20 ms	99.999%	25 Mbps
Road-side infrastructure backhaul [232]	30 ms	99.999%	10 Mbps
Processed HD map sharing [233]	100 ms	99%	4 Mbps
RAW HD map sharing[233]	100 ms	99%	47 Mbps
Hazard event collection[233]	20 ms	99.9%	300 bytes/message
Voice Communication (for rail operational purposes) [246]	100 ms	99.9%	100-300 kbps
Rail safety critical video communication [246]	10 ms	99.9%	10-30 Mbps
Rail very critical data communication [246]	10 ms	99.9999%	0.1-1 Mbps

As a possible application scenario falling in this general setting, in the following the AT scenario is considered. To elaborate, the automotive industry is strongly committed to the development of autonomous cars, buses, and trucks. Such an effort can boost at the same time the development of autonomous urban transportation systems. The implementation of autonomous or remote driving systems poses challenging requirements since these services are exposed to several critical uncontrollable events such as pedestrians, vehicles, and obstacles, demanding a much higher level of situational awareness and more dynamic interaction. In this respect, the concept of Grade of Automation, defined by the Union Internationale des Transports Publics [248] for Automatic Trains Operation has inspired the SYSTRA model, which defines six levels to qualify the Levels of Automation, from no automation to fully autonomous where the tram drives itself without any onboard agent [249].

In this complex scenario, fusing the data from vehicle sensors with the information coming from the infrastructure, as well as from the city environment, can significantly increase the situational awareness of vehicles. For these reasons, the use of edge and cloud computing resources available across the city can provide a very good situational awareness and resiliency properties due to its high performance and flexibility capabilities, contributing to safety. Besides, for operation purposes, each vehicle must provide a precise and accurate measurement of its location. In addition, a data fusion algorithm running on a MEC server is needed to merge data from multiple and heterogeneous sensors.

Figure 4.13 reports the main modules that are necessary to carry out the AT use case [250]. It is put in evidence that AT is not based on one single technology, but rather it is a highly complex system that consists of many sub-systems which can be grouped into two major components: algorithms, including sensing, perception, and decision, and the platform system, including the operating system and the hardware components. To further elaborate, preparatory functions of the future ATs include Autonomous Positioning, Obstacle Avoidance [251], and Remote Tram Control [252]. The Autonomous Positioning system must provide a precise and accurate measurement of the tram location along the rail track based on sensor data by transmitting a few kbps of data within a stringent latency constraint (less than 10 ms). The Obstacle Avoidance system utilizes computer vision to detect, classify and track objects that can potentially become tram's obstacles. Accordingly, the required throughput for Obstacle Avoidance is much

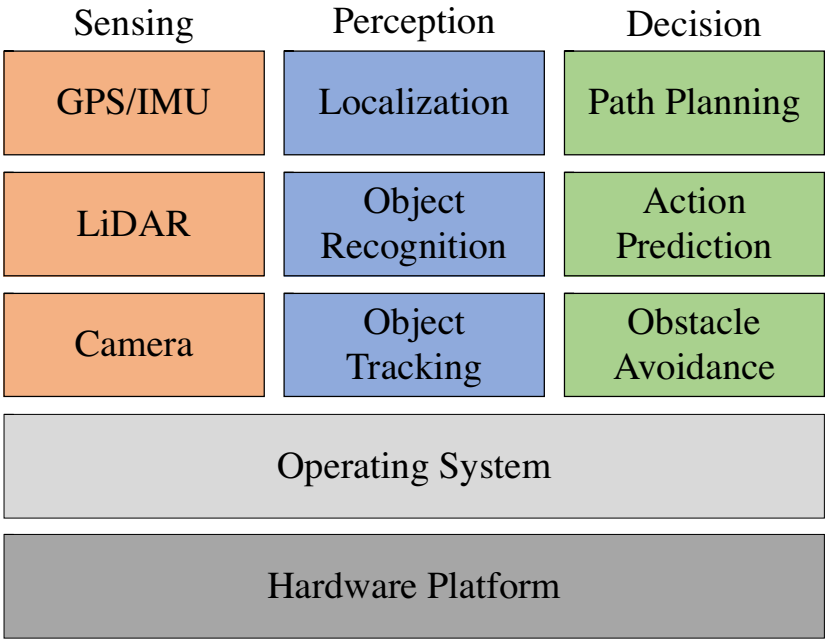


FIGURE 4.13: Autonomous Tram Modules

higher than Autonomous Positioning, while the latency requirement can be slightly relaxed. Finally, the Remote Tram Control is an important step towards autonomous driving, both in terms of the technical requirements, such as on-board safety systems to manually intervene or brake in case of malfunctioning, high throughput and low latency in communications, and complex control algorithms, as well as in terms of public acceptance of vehicles not directly manually driven. Indeed, Remote Tram Control can support full autonomy in case of downgraded scenarios when autonomy is no more available and the tram has to park aside and stop. In this case, the driver should go onboard to recover the vehicle. With remote driving, tram recovery will be done from remote and the requirements are even more challenging because raw data (video, sensors data, etc) have to arrive at the control room in real-time. The KPIs are measured in terms of the E2E latency which is determined by the time that takes to transfer a given piece of information unidirectional from a source to a destination, measured at the communication interface, from the moment it is transmitted by the source to the moment it is successfully received at the destination, and reliability which is the percentage value of the amount of sent network layer packets successfully delivered to a given system entity within the time constraint required by the targeted service, divided by the total number of sent network layer packets [246].

Accordingly, the fundamental communication service for the purposes of the AT service is represented by safety-critical video communication (as

reported in Table 4.5, [246]), where screen rendering is performed at the edge and the uplink emerges as the most important part of the link.

4.5.2 System Model and Problem Formulation

In the following, a V2I cellular communications scenario is considered, where the infrastructure is composed of RSUs co-located with gNBs to provide the 5G network communications capabilities together with RSU functions [253]. More specifically, RSUs are stationary infrastructures supporting V2X applications that can exchange messages with other entities supporting V2X applications. It is a logical entity that combines V2X application logic with the functionality of a gNB (referred to as gNB-type RSU[254]). Hence, the cellular environment includes a RAN controller and a MEC located at the edge of the Core Network, i.e., in the proximity of a cluster of 5G gNBs providing coverage to the service area.

Since the considered ATs scenario belongs to the class of mission-critical services, it is still reasonable to assume that the TNT slice resource allocation requests must be always accepted by the IP. Nevertheless, the already discussed *pay for what you get* mechanisms can be still utilized by the IP to prevent over-provisioning the AT TNT. One of the fundamental goals of the AT service is the estimation of the exact position of the users, which must be computed in real-time by the AT application running on the MEC. Moreover, owing to the regularity and predictability of vehicle movements, the TNT can timely prepare the slice requests based on coverage and performance properties of each gNB, e.g., whether or not a gNB covers a given area of interest.

As for the specific RAN slicing strategy, the resources are virtually allocated by the TNT leveraging the exact knowledge of users' position.

This work considers the uplink of a sectorized cellular system in which the gNBs are equipped with third-order sector antennas, i.e., the cells are divided into 3x120 degrees sectors as shown in Figure 4.14.

Let $\mathcal{B} = \{1, 2, \ldots, B\}$ be the set of sectors and $\mathcal{M} = \{1, 2, \ldots, M\}$ the set of active users in the service area. The total bandwidth is partitioned into a set $\mathcal{C} = \{1, 2, \ldots, C\}$ of virtual frequency resources referred to as RBPs which are managed by the TNT and which correspond to the amount of bandwidth W to be allocated to each user. The exact physical bandwidth necessary to fulfill the QoS requirements will, in general, depend on several factors, e.g., the interfering conditions and the adopted physical layer procedures such as the presence of Massive MIMO or Coordinated MultiPoint among nearby

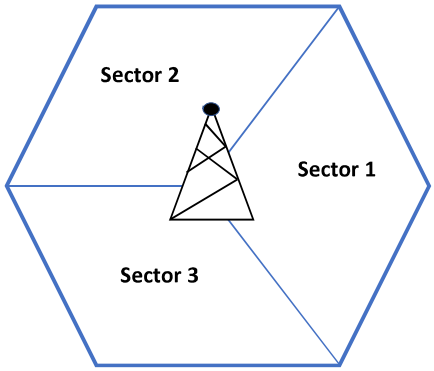


FIGURE 4.14: A gNB with three sectors

gNBs, the channel propagation models, etc. Hence, IP is in charge of the mapping between a virtual RBP and the actual physical bandwidth, which has full control of the physical layer, with the constraint that different RBPs must correspond to non-overlapping portions of the available spectrum. The virtual-to-physical mapping problem will be investigated in the following Section making use of realistic simulations. The sequel focuses on the optimal slice request problem at the TNT.

Let $(m) \in \mathcal{B}$ represents the sector to which the m-th user is associated. The sector association procedure is carried out by the IP which communicates its decisions to the TNT making use of the IP-TNT APIs (see Figure 4.1). Another information that is assumed to be available at the TNT is the position of the involved gNBs.

As for the slice request optimization problem, which is fully under the control of the TNT, it is carried out making use of the knowledge of the exact positions of the users. To elaborate, the TNT evaluates a user-specific adjacent matrix $\mathbf{Y} = y_{m,b}$, with $m \in \mathcal{M}$ and $b \in \mathcal{B}$ and a pair-wise adjacent matrix $\mathbf{Z} = z_{m,m'}$ with $m, m' \in \mathcal{M}$. More specifically, $y_{m,\ (m)} = 0$ and $y_{m,b} = 1$ if sector b interferes with user m or $y_{m,b} = 0$ otherwise. Moreover, $z_{m,m} = 0$ and $z_{m,m'} = 1$ if user m' interferes with user m, or $z_{m,m'} = 0$ otherwise. These matrices are dynamically evaluated on the basis of the positions of the users. As an example, if user m associated to sector $b = \ (m)$ is close to the border with sector $b' \neq b$, a reasonable choice is to set $y_{m,b'} = 1$. Conversely, if user m is close to its serving gNB, one could set $y_{m,b'} = 0$, for all $b \in \mathcal{B}$. On the other hand, $z_{m,m'}$ will depend on the reciprocal positions of users m and m'. An illustrative example is shown in Figure 4.15. Note that the matrix $\mathbf{Z}$ is not necessarily symmetric and $y_{m,\ (m')}$ is in general different from $y_{m',\ (m)}$.

TABLE 4.6: List of Symbols

Symbol	Remarks
$\mathcal{B}$	Set of sectors $\{1, 2, \ldots, B\}$
$\mathcal{M}$	Set of active users in the service area $\{1, 2, \ldots, M\}$
$\mathcal{C}$	Set of Resource Block Pools (RBPs) $\{1, 2, \ldots, C\}$
m	Index of user
b	Index of sector
c	Index of RBP
(m)	Sector of the m-th user
$\mathbf{Y}$	User-specific adjacent matrix with elements $y_{m,b}$
$\mathbf{Z}$	Pair-wise adjacent matrix with elements $z_{m,m'}$
$x_{m,c}$	Allocation variable $\{0, 1\}$
W_{tot}	Total amount of bandwidth reserved for the slice
$\bigvee$	Logical OR operator
R_b	Total number of reserved resources in sector b
(m)	Reordering function for user m
$\mathcal{M}$	Reordered set
$\beta(\quad)$	Objective function corresponding to order $\mathcal{M}$
L_m	Linked Resources for user m
$\overline{z'}$	Logical NOT of z'
$\mathcal{M}_{\ 0}$	Initial order
$_{i,j}$	Permutation of $_{\ 0}$ obtained by exchanging the i-th and j-th position of $_{\ 0}$
I_t	Total number of iterations
R_0	Cell Radius
$d(m)$	Distance of user m from center of it's serving sector
$\mathcal{A}_m$	Interference area for user m
R_m	Radius of interference area for user m from the center of serving cell
α_I	Distance of interference area from the center of serving cell
$_I$	Angle of interference area from the center of serving cell
$\mathcal{S}_b$	Set of two interfering sectors of the same cell for sector b
$\mathcal{N}_b$	Set of four interfering sectors to sector b delimited by its sector beam
α_T	Threshold distance to determine the potential interference sectors [0,1]
W_η	Bandwidth per sector
	Reliability

The slice requests formulated by the TNT are expressed in terms of RBP allocations. To this aim, let $x_{m,c} \in \{0, 1\}$ be the allocation variable such that $x_{m,c} = 1$ if RBP $c \in \mathcal{C}$ is allocated to user m and $x_{m,c} = 0$ otherwise. Such allocation decisions are passed from the TNT to the IP through the available APIs, which in turn enforce them through the reservation of a correspondent physical bandwidth to each user m in the serving sector (m), with $m \in \mathcal{M}$. In this case, the total amount of bandwidth reserved for the considered slice is equal to $W_{tot} = MW$ and no mechanism is adopted to protect the users

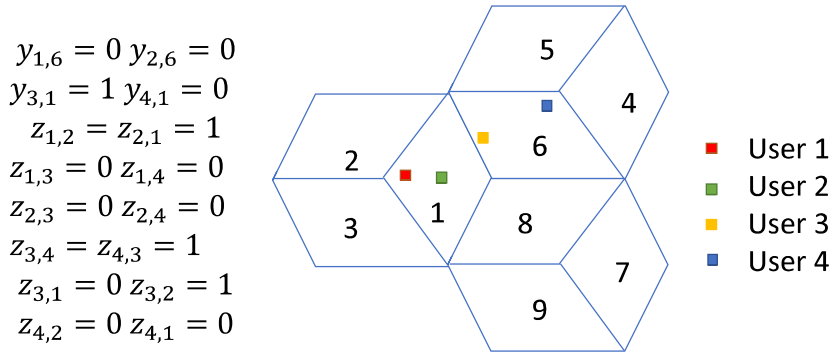


FIGURE 4.15: Illustrative example with 9 sectors: user 1 is in sector 1 and it is close to its gNB, user 2 is in sector 1 and it is in the middle of the sector, user 3 is in sector 6 and it is close to the border with sector 1, user 4 is in sector 6 and it is close to its gNB.

from inter-slice interference., e.g., through the reservation of additional bandwidth. Accordingly, the only mechanism for controlling inter-slice interference is given by the possibility of sharing the same bandwidth among users of the same slice in adjacent (interfering) gNBs. Such mechanism has already been investigated in [182] where the intra-slice shared bandwidth is denoted by linked bandwidth. To better clarify, consider again the example in Figure 4.15. A possible solution is given by $x_{1,1} = 1$, $x_{2,2} = 1$, $x_{3,1} = 1$ and $x_{4,2} = 1$, in which the bandwidth allocated to users 1 and 3 is linked, and, hence, user 3 does not receive any interference from sector 1 despite $y_{3,1} = 1$. Hence, the situation where the resource c is linked for users m and m' in sector $b = \ (m')$ is expressed by the condition $y_{m,\ (m')} x_{m,c} x_{m',c} = 1$. Note that, in the example of Figure 4.15 the bandwidth allocated to users 1 and 3 could be used in sector 8 by different slices' users, thus producing non-negligible inter-slice interference. Accordingly, this proposed approach is denoted byRS-NOISP. For notational convenience, let $\mathbf{Z}'$ with $z'_{m,m'} = z_{m,m'} \bigvee z_{m',m}$ be a symmetric matrix, where $\bigvee$ is the logical OR operator. Accordingly, $z'_{m,m'}$ indicates whether users m and m' can be allocated the same RBP. Then, the RS-NOISP problem can be formulated as:

$$\max_{\mathbf{x}} \sum_{c,m,m'} y_{m,\ (m')} x_{m,c} x_{m',c} \tag{4.10}$$

subject to:

$$\sum_{c} x_{m,c} = 1 \quad m \in \mathcal{M} \tag{4.10.a}$$

$$\sum_{c,m,m' \neq m} z'_{m,m'} x_{m,c} x_{m',c} = 0 \tag{4.10.b}$$

$$x_{m,c} \in \{0,1\} \tag{4.10.c}$$

where the objective function corresponds to the amount of linked bandwidth, constraint (4.10.a) ensures that all the users receive one RBP while constraint (4.10.b) guarantees that none of the adjacent users receive the same RBP (i.e., intra-slice interference is avoided).

In order to provide a higher inter-slice isolation level, this work proposes a mechanism based on bandwidth over-provisioning denoted by RS-ISP. In this case, the same resource allocated to user m on sector b is reserved in all its adjacent sectors. To elaborate, if $x_{m,c} = 1$, then the same RPB c is reserved in all sectors b such as $y_{m,b} = 1$. Note that, if two users have Linked Resources, such resources can be reserved only once. Mathematically, this means that the number of reserved resources in sector b is $R_b = \sum_{c} \bigvee_{m=1}^{M} y_{m,b} x_{m,c}$. In this setting, an optimal RS-ISP strategy aims at minimizing the total bandwidth reserved for the considered service, i.e., the RS-ISP problem can be formulated as:

$$\max_{\mathbf{x}} -\sum_{b} \sum_{c} \bigvee_{m=1}^{M} y_{m,b} x_{m,c} \tag{4.11}$$

subject to (4.10.a), (4.10.b) and (4.10.c)

In this second case, the TNT sends to the IP the outcome $\mathbf{x}$ of problem 4.11 and the adjacent matrix $\mathbf{Y}$, so that the IP may perform the slice enforcement accordingly.

4.5.3 Heuristic solution for RS-NOISP and RS-ISP problems

Problems (4.10) and (4.11) belong to the class of Mixed-Integer Non-Linear Programming problems that combine the combinatorial difficulty of optimizing over discrete variable sets with the challenge of handling nonlinear functions. In particular, in the RS-ISP case, which will be shown to represent the best approach for the problem at hand, the objective function is strongly nonlinear, i.e., it is neither linear nor polynomial. To circumvent the difficulty, a heuristic approach is proposed for achieving a reasonably good solution with limited complexity.

The proposed approach, referred to as Pairwise Reordering Improvement (PRI) scheme, is based on a two-fold procedure: (*i*) find a feasible solution according to a given *good* initial sequential order; (*ii*) iteratively improve the initial order considering a pairwise reordering approach. To elaborate, consider a reordering function of the set $\mathcal{M}$, where $(m) \in \mathcal{M}$ is the user index occupying position m and the reordered set is denoted by $\mathcal{M}$. For any given order $\mathcal{M}$, the feasible solution can be found by sequentially assigning the first available RBP to each user and the objective functions (4.10) and (4.11) can be evaluated accordingly. For ease of notation, $\beta(\ )$ denotes the objective function corresponding to order $\mathcal{M}$ without distinction between RS-NOISP or RS-ISP.
The initial order $\mathcal{M}_{\ 0}$ is established basing on the amount of possible Linked Resources for each user, evaluated as $L_m = \sum_{m'} y_{m,\ (m')} \overline{z'}_{m,m'}$, where $\overline{z'}$ is the logical NOT of z'. The procedure for establishing the initial order is illustrated in Algorithm 2. The order is obtained by adding to $\mathcal{M}_{\ 0}$, at each iteration, the index $\tilde{m}$ with maximum $L_{\tilde{m}}$ and all its linked users. Hence, the remaining set $\mathcal{Q}$ is obtained by subtracting the set of users already inserted in $\mathcal{M}^{(0)}$.

AS for the pairwise reordering part of the heuristic solution, let $_{i,j}$ be a permutation of $_0$ obtained by exchanging the i-th and j-th position of $_0$, i.e.:

$$\begin{aligned} {}_{i,j}(i) &= {}_0(j) \\ {}_{i,j}(j) &= {}_0(i) \\ {}_{i,j}(k) &= {}_0(k) \ \ k \neq \{i,j\} \end{aligned} \tag{4.12}$$

Algorithm 2: Procedure for establishing the initial order

Initialize: $\mathcal{M}_{(0)} = \emptyset$, $\mathcal{Q} = \mathcal{M}$;
while $\mathcal{Q} \neq \emptyset$ **do**
  Evaluate: $L_m, \forall m \in \mathcal{Q}$;
  $\tilde{m} = \arg\max_{m \in \mathcal{Q}} L_m$;
  $\mathcal{T} = \left\{ i \in \mathcal{Q} : y_{\tilde{m},(i)} \overline{z'}_{\tilde{m},i} = 1 \right\}$;
  $\mathcal{P} = \tilde{m} \bigcup \mathcal{T}$;
  $z'_{m,m'} = 1$, for all $m \in \mathcal{P}$, $m' \in \mathcal{Q}$; $\mathcal{M}^{(0)} = \mathcal{M}^{(0)} \bigcup \mathcal{P}$;
  $\mathcal{Q} = \mathcal{Q} \setminus \mathcal{P}$;

In addition:

$$\mathcal{C} = \{i, j\} \quad \text{such as } y_{i,(j)} \overline{z'}_{i,j} = 0 \tag{4.13}$$

i.e., i and j are not linked. Then, the best reordering $_{i^*,j^*}$ is evaluated as:

$$\{i^*, j^*\} = \arg\max_{\{i,j\} \in \mathcal{C}} \beta\left(\ _{i,j}\right) \tag{4.14}$$

Finally, if $\beta\left(\ _{i^*,j^*}\right) = \beta\left(\ _{i,j}\right)$ the search is stopped, otherwise set $_0 =\ _{i^*,j^*}$ and run another iteration for a maximum of I_t iterations. Note that the complexity of the PRI scheme is upper bounded by $\mathcal{O}\left(I_t M(M-1)/2\right)$, i.e., it increases with M^2.

To assess the effectiveness of the proposed heuristics, it is carried out the simulation of a cellular network in which M users are randomly deployed over a cluster of 7 three-sectors cells with radius R_0. Each user m is associated with a serving sector (m) based on its position. The distance of user m from the center of the serving sector is denoted by $d(m)$. Hence, each tram m is associated to an interference area $\mathcal{A}_m$ surrounding the serving sector such as $z_{m,m'} = 1$ if $m' \in \mathcal{A}_m$, and $z_{m,m'} = 0$ otherwise. In particular, $\mathcal{A}_m$ is given by an arc of the circumference with radius $R_m = \alpha_I d(m)$ and angle $2\Theta_I$ centered in the serving sector, where α_I and Θ_I are two design parameters.

An illustrative example is shown in Figure 4.16. Note that in the case of ideal cell sectorization, it would be sufficient to set $\Theta_I = \pi/3$ since no interference can be received outside this angle. However, some degree of overlap among sectors cannot be avoided in practice, and hence a precautionary approach can be recommendable, as shown in the next Section.
As for $\mathbf{Y}$, two sets of sectors which may potentially create interference to sector b are considered. The first set is denoted by $\mathcal{S}_b$ and is given by the two

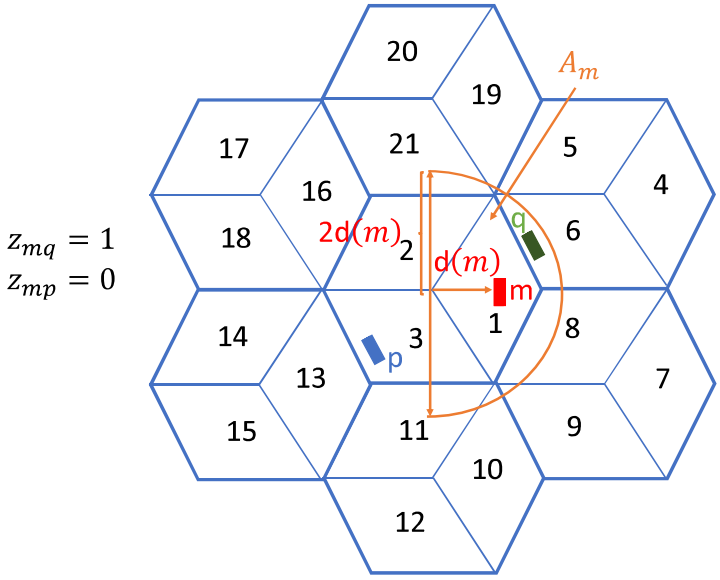


FIGURE 4.16: An illustrative example with $\Theta_I = \pi/2$ and $\alpha_I = 2$.

sectors of the same cell to which sector b belongs. As an example, in Figure 4.16 there is $\mathcal{S}_1 = \{2,3\}$, $\mathcal{S}_4 = \{5,6\}$, etc. The second set is denoted by $\mathcal{N}_b$ and it is composed by the 4 adjacent sectors to sector b lying in the region delimited by its sector beam. As an example, in Figure 4.16, $\mathcal{N}_1 = \{5,6,8,9\}$, $\mathcal{N}_6 = \{2,1,8,7\}$, etc.

To elaborate, it is worth noting that the positions of potentially interfering users of different slices are not known, and, hence, they could be located everywhere in the interference area of user m. Accordingly, $y_{m,b} = 1$ for all $b \in \mathcal{S}_{(m)}$. As for the interference from adjacent sectors, it is reasonable to assume that its effect depends on $d(m)$. As a case in point, if user m is very close to its gNB, i.e., $d(m)$ R_0, the interference from adjacent sectors can be neglected. Accordingly, for all $b \in \mathcal{N}_{(m)}$, $y_{m,b} = 1$ if $d(m) \geq \alpha_T R_0$, and $y_{m,b} = 0$ otherwise, where $0 \leq \alpha_T \leq 1$ is a design parameter.

Figures 4.17 and 4.18, report the performance metric as a function of M obtained through the proposed PRI scheme with $I_t = 5$ in the RS-NOISP and RS-ISP cases, respectively. More specifically, Figure 4.17 reports the number of Linked Resources (LR), while Figure 4.18 reports the total number of allocated resources normalized to the number of sectors (TR). The results have been obtained considering $\alpha_I = 2$, $\Theta_I = \pi/2$ and $\alpha_T = 0.5$. The PRI approach is compared with three alternatives, namely, a single iteration approach in which the allocation is performed based on the initial order (No-PRI), a single-shot random-ordering mechanism (Random), and a Monte Carlo (MC) approach with $N_o = 100000$ independent random orders.

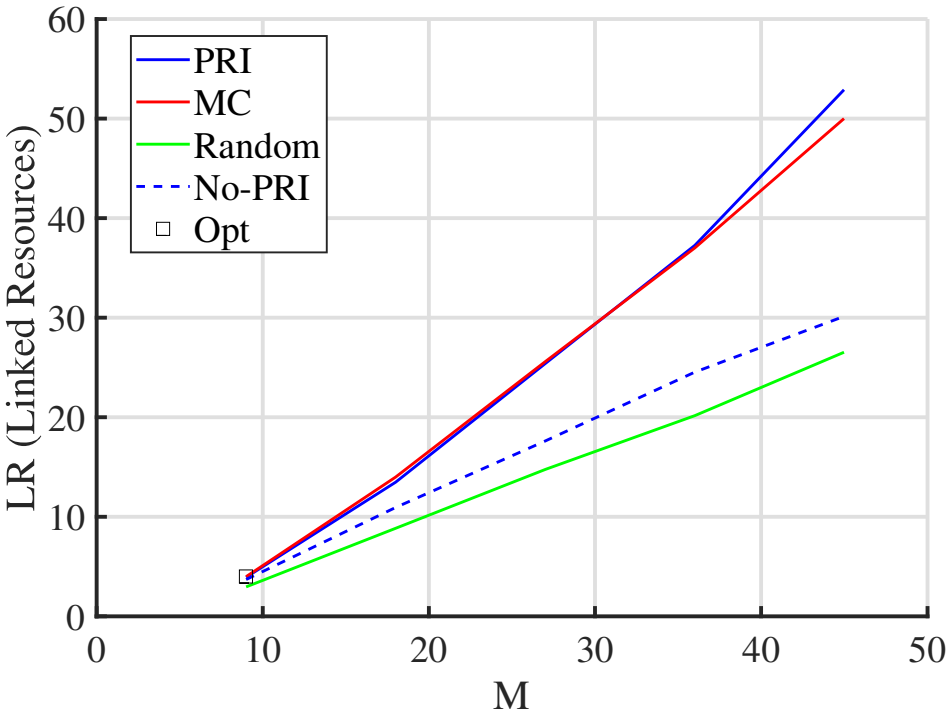


FIGURE 4.17: RS-NOISP case: Linked Resources for the four considered schemes vs the number of ATs M.

When the number of possible orders $M!$ is comparable to or lower than N_o the MC is expected to reasonably approach the optimum exhaustive search. Figures 4.17 and 4.18 also report the results obtained by the optimal exhaustive search (Opt) in the $M = 9$ case, which confirm this expectation. Unfortunately, owing to complexity, it is impossible to get a comparison for higher M values. From Figures 4.17 and 4.18, it is evidenced that the proposed mechanism allows to noticeably outperform the Random and the No-PRI schemes for all the considered M values and to outperform the MC approach for high M values with a remarkable complexity reduction. Indeed, for example, if $M = 45$ the complexity $\mathcal{O}\left(I_t M(M \quad 1)/2\right)$ of the PRI is of few thousands of operations which is almost two orders of magnitude lower than N_o.

4.5.4 Experimental Results

4.5.4.1 5G-air-simulator Development

In pursuance of obtaining an effective enforcement strategy for the proposed RAN slicing mechanism, 5G-air-simulator has been properly extended.

First of all, a realistic uplink interference model has been implemented, which was completely absent originally. Then, mMIMO with MRC features

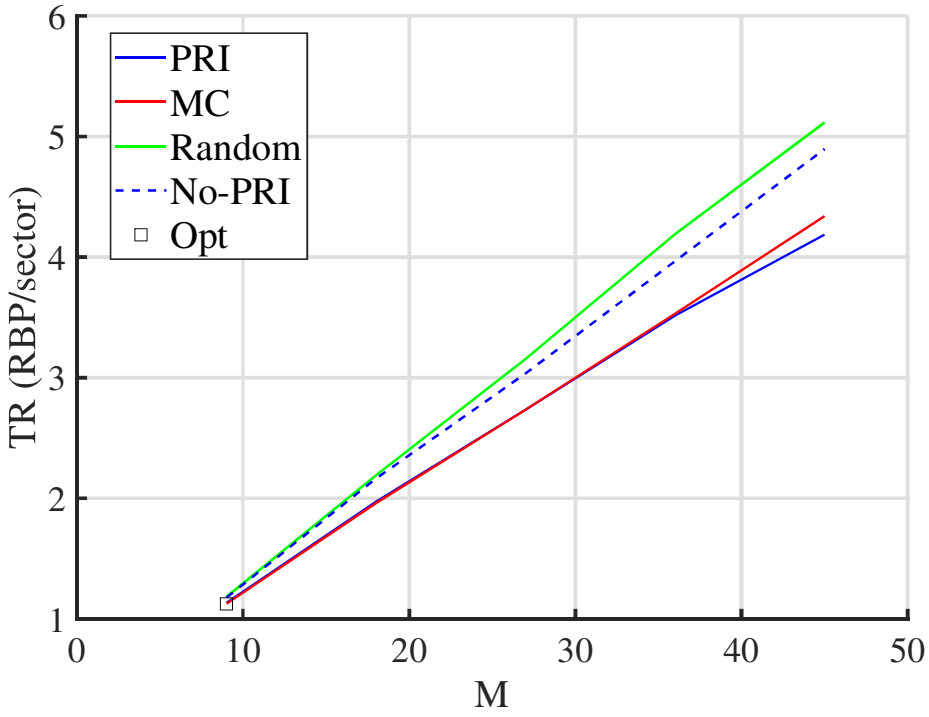


FIGURE 4.18: RS-ISP case: TR for the four considered schemes vs the number of users M.

has been brought to the uplink, in order to fully exploit 5G NR features for UE transmissions. Thanks to the MRC filtering and the new interference model, the actual SINR is measured for each RB. The final vector of SINRs is used to detect eventual physical errors in the uplink, similarly to the downlink procedure. Following the same BLER rationale in the downlink, the retransmission mechanism (i.e., ARQ at the RLC layer) has been implemented in the uplink as well.

Moreover, since multi-connectivity has the potential to enable reliable handovers without handover interruptions for user plane data, providing a form of RRC diversity [255], a DC handover module has been implemented in the form of a Twin Model handover shown in Figure 4.19, which is able to mimic a DC transmission mode where the user is connected to two different gNBs at the same time.

In particular, a Twin UE is created for each UE with the same mobility as the original UE, i.e., the twin node will always be located at the very same position as the original UE. The proposed approach is sketched in Figure 4.19. When the handover event is triggered for the UE, the Twin UE is created and a twin radio bearer for this node is created in the Target gNodeB (T-gNB). This twin radio bearer is used to transmit a replica of the original

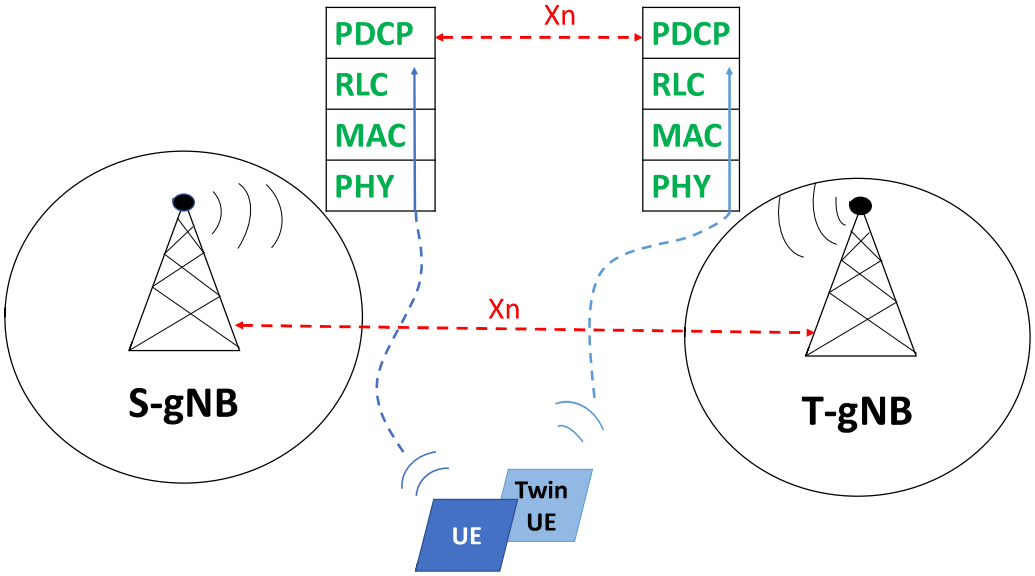


FIGURE 4.19: Schematic diagram of DC handover module in the 5G-air-simulator

packet flow towards the T-gNB, whereas the UE continues its transmissions towards the Serving gNodeB (S-gNB). Hence, there is a dual transmission until the handover is completed, when the Twin UE is deactivated. The detection and discarding of eventual duplicates are carried out at the S-gNB assuming an ideal Xn interface allowing the T-gNB to forward the service data units received from the twin nodes towards the S-gNB. Following the *bearer split on PDCP level with data duplication* mode proposed for 5G/5G DC [127], the packet is tagged as dropped only if both versions are lost at the PDCP, thus achieving a link-layer diversity of order two. It is essential to highlight that during dual transmissions the original UE and its Twin counterpart transmit with half power to model a realistic DC. Finally, the possibility of implementing DC handover is facilitated by the RS-ISP reservation mechanism described previously. Indeed, in this case, the same RBP reserved for transmission in the S-gNB is also reserved in the adjacent T-gNB. Accordingly, the DC handover can be operated on the same RBPs without requiring a timely new reservation of resources, as it would be necessary for the RS-NOISP case.

It is worth highlighting that the implementation of handover strategies (Hard/DC) is not considered as a part of the RAN slicing policy described in previous Sections. In particular, TNT is in charge of allocating virtual resources to users in the slice and it does not have any knowledge about the

handover (either DC or Hard) algorithm adopted by the IP. The implementation of DC handover is a choice of the IP and it is important to determine the most proper mapping of virtual to physical resources, as shown in the following.

Finally, the proposed resource allocation mechanism for slicing enforcement has been integrated into the simulator. To this aim, a specific Autonomous Tram - Network Slice Selection Assistance Information (AT-NSSAI) is used to identify the AT slice and is associated with the radio bearers of each AT user. Hence, at every predefined time interval of 100 ms, the RAN controller running on the MEC evaluates the new reservation policy based on the TNT's directives. To this aim, a physical bandwidth of W Hz is associated with each non-overlapping RBP. This information is handled by the Service Data Application Protocol at each gNB which associates to each uplink AT-NSSAI radio bearer an indicator of the physical RBs reserved for it. Finally, a dedicated MAC instance is considered for AT users which performs MAC scheduling based on the reserved resources, while the *free* spectrum is shared among non-Autonomous Tram (NAT) users.

4.5.4.2 Test Scenario

The test scenario is built upon the ITU Urban Macro - URLLC scenario[109]. However, a larger network is simulated to encompass a more realistic AT scenario. In particular, the service area is composed of 21 cells (63 sectors) with a radius of 500 m. As for the number of UEs belonging to the AT slice type (denoted by AT UEs), a worst-case scenario with a tram for each site is taken into account. Indeed, it is reasonable to assume unlikely to have a higher density of AT UEs in urban environments. Hence, 21 AT UEs are randomly deployed over the service area, moving in a rectilinear direction with a speed of 30 km/h. Such users are assigned RBPs according to the allocation mechanism for slicing enforcement described previously. To provide a realistic scenario including extra-slice interference, 63 always active NAT users are deployed in the same service area. Such users are allocated the whole free spectrum, i.e, the spectrum not reserved for the AT slice.

According to [256], the maximum output power for Class 1, Class 1.5, Class 2, and Class 3 5G UEs operating in NR FR1 are 31 dBm, 29 dBm, 26 dBm, and 23 dBm respectively without taking into account the tolerance. In particular, the maximum power class for the UL MIMO in the closed-loop

spatial multiplexing scheme is 29 dBm which is achieved via dual transmission. Therefore, the power of AT users is set to 28 dBm to model a realistic scenario. Moreover, this assumption is quite reasonable due to the fact that trams are huge and can accommodate large power supply compared to normal UEs like cell phones which have limited energy resources and need frequent battery recharge. As for the power settings for NATs, they are taken from the references [109], [256].

Finally, as for the three-sector coverage, the fan-beam antenna system reported in [257] is considered, which is compliant with the ETSI EN 301 215-2 Class CS 2 requirements.

Table 4.7 summarizes the simulation parameters considered for the following discussions. Note that the other simulation parameters have been taken from the ones already shown in Chapter 2.

TABLE 4.7: Simulation Parameters for Considered Scenario

Parameter	Value
Operating frequency	3.7 GHz (channel model [109])
Number of sites	21
Cell radius	500 m [109]
Number of sectors per site	3 [109]
Simulation time	60 s
Number of transmitting beams	2
Number of receiving beams	32
bandwidth	200 MHz
Sub-carrier spacing	30 kHz
AT UEs transmitting power	28 dBm
NAT UEs transmitting power	23 dBm [109]
MIMO Transmission Layers	2
gNB height	25 m [109]
Number of AT UEs	21
Number of NAT UEs	63
UL Scheduler	Max Throughput
Traffic model (AT UEs):	Video streaming at 15.1 Mbps
Traffic model (NAT UEs):	Full-Buffer
AT UEs' Speed	30 km/h [109]
NAT UEs' Speed	3 km/h [109]
AT UEs Antenna Height	3.5 m
NAT UEs Antenna Height	1.5 m [109]
Reservation Periodicity	100 ms
handover Threshold	3 dB
Number of simulation seeds	250
RLC Re-transmission (AT UEs)	5 ms
Max delay (AT UEs)	10 ms
Re-transmission threshold (AT UEs)	16

4.5.4.3 Numerical Results

As for the virtual to physical resource mapping, simulations are run for different values of W to experimentally assess the best slice enforcement strategy. Hence, the reliability of AT UEs is evaluated, intended as the fraction of network layer packets successfully delivered within the time constraint required by the targeted service, that is 10 ms in the considered

scenario (i.e., rail safety-critical video communication, see Table 4.5). For a fair comparison between RS-NOISP and RS-ISP, let the required bandwidth per sector be W_η, intended as the total bandwidth needed to implement the AT slice divided by the number of sectors. To clarify, in the RS-NOISP case, the total bandwidth is simply W multiplied by the number of AT UEs, i.e., $W_\eta = 21W/63 = W/3$ in the considered setting. On the other hand, owing to additional spectrum reservation, in the RS-ISP case the total bandwidth is increased and W_η will depend on the adopted reservation strategy, which is in turn determined by the values of $_I$, α_I, and α_T. The following Figures report W_η versus for different slicing strategies. To reduce the set of possible parameters combinations, the interference area has a radius that is two times the distance from the serving gNB (i.e., $\alpha_I = 2$). Moreover, 4 possible values of $_I$ are considered, namely, $_I = \pi/3$, $_I = \pi/2$, $_I = 2\pi/3$ and $_I = \pi$ for both RS-NOISP and RS-ISP cases, and two possible values of α_T, namely, $\alpha_T = 0$ and $\alpha_T = 0.5$ for the RS-ISP case.

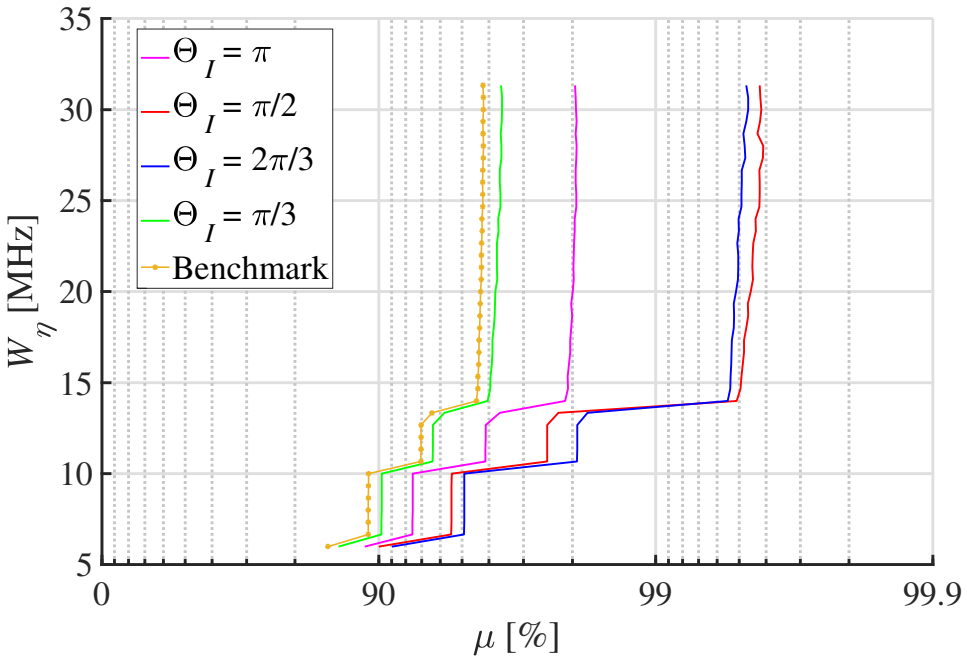


FIGURE 4.20: Performance of RS-NOISP

Figure 4.20 reports the performance of the RS-NOISP strategy for four considered values of $_I$. For comparison purposes, it also reports as a benchmark the results obtained through the approach proposed in [182]. As already discussed, indeed, [182] is the most closely paper to this work. In particular, it proposes a slicing mechanism aimed at maximizing the linked

bandwidth without considering the locations of the involved users. Hence, a comparison with [182] allows highlighting the importance of the location information in devising the slicing mechanisms proposed here. These results are obtained considering hard handover. The curves reveal a clear trade-off between intra- and extra-slice interference. To better understand, $_I = \pi/3$ represents the worst case in terms of intra-slice interference since in this case the same RBPs can be always allocated to AT UEs located in different sectors of the same cell. For the same reason, there is the maximum possible linked bandwidth for AT UEs, and, hence, this is the best case in terms of inter-slice interference. On the opposite, there is the $_I = \pi$ case. It is possible to observe that the best trade-off between the four situations is given by the two intermediate cases, and in particular, $_I = \pi/2$ is the best solution. It is also shown that the proposed approach allows outperforming the benchmark despite that, due to the residual inter-slice interference that is present in all cases, the reliability cannot be increased above a given threshold of 99.7 %. This reliability factor can be achieved at the expense of consuming 1/6 of the total available bandwidth. To sum up, the RS-NOISP case does not allow to fulfill the QoS requirements reported in Table 4.5. These results confirm that if the QoS constraints are strict, there is a need for an additional mechanism to protect against inter-slice interference.

Figure 4.21 reports the performance of the RS-ISP strategy for the 6 combinations of $_I$ and α_T. These results have been obtained considering DC handover. For comparison purposes, Figure 4.21 also reports the performance of the best RS-NOISP case. It is shown that the best performance is obtained in the case $_I = \pi/2$ and $\alpha_T = 0.5$, at least in achieving a reliability value higher than 99%. In particular, such a combination allows achieving the required reliability of 99.9 % reported in Table 4.5 with a spectrum usage of nearly 1/12 of the total available bandwidth.

Lastly, to put in evidence the effect of DC-handover with respect to hard handover, Figure 4.22 reports the performance of the best RS-ISP and RS-NOISP cases with hard handover and compare them with the DC handover case. It is shown that DC-handover allows to clearly outperform hard handover. On the other hand, for high-reliability values, the RS-ISP scheme with hard handover performs clearly better than RS-NOISP, and, in particular, it allows to achieve the required 99.9% reliability with a spectrum usage of

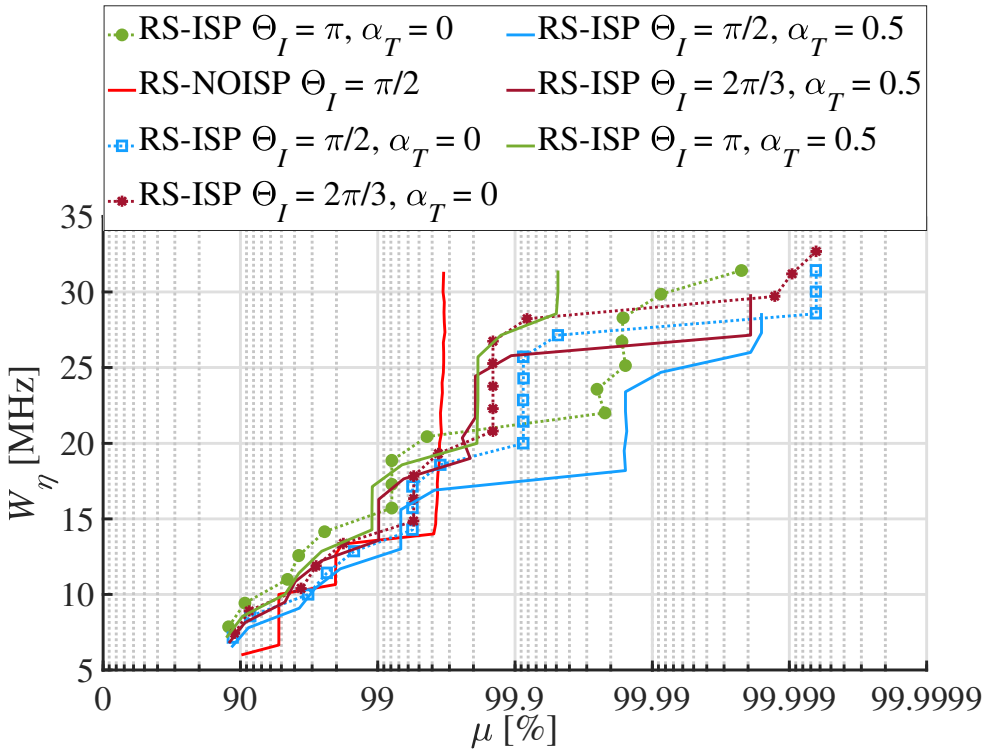


FIGURE 4.21: Performance of RS-ISP

nearly 1/10 of the total available bandwidth. In other words, the DC handover further helps in reaching the target reliability values while requiring less bandwidth.

To sum up, the simulator allowed the establishment of the best system dimensioning for guaranteeing the required QoS with the minimum amount of resources required to enforce the AT slice, aiming at providing guidelines in enforcing the considered AT scenario.

4.6 Slice Management for Pervasive In-Home Healthcare using Cascaded WLAN-FWA

Pervasive healthcare envisages continuous and ubiquitous monitoring of physiological signals and vital parameters while improving the living conditions of patients at their homes. Especially for patients in the most critical conditions, e.g., patients suffering from severe epilepsy, continuous monitoring is crucial for effective life-saving interventions in case of emergency, and to prolong life expectancy. For example, up to a third of all premature deaths worldwide are either directly or indirectly attributed to epilepsy [258]. Particularly, Sudden Unexpected Death in EPilepsy (SUDEP) is a direct cause

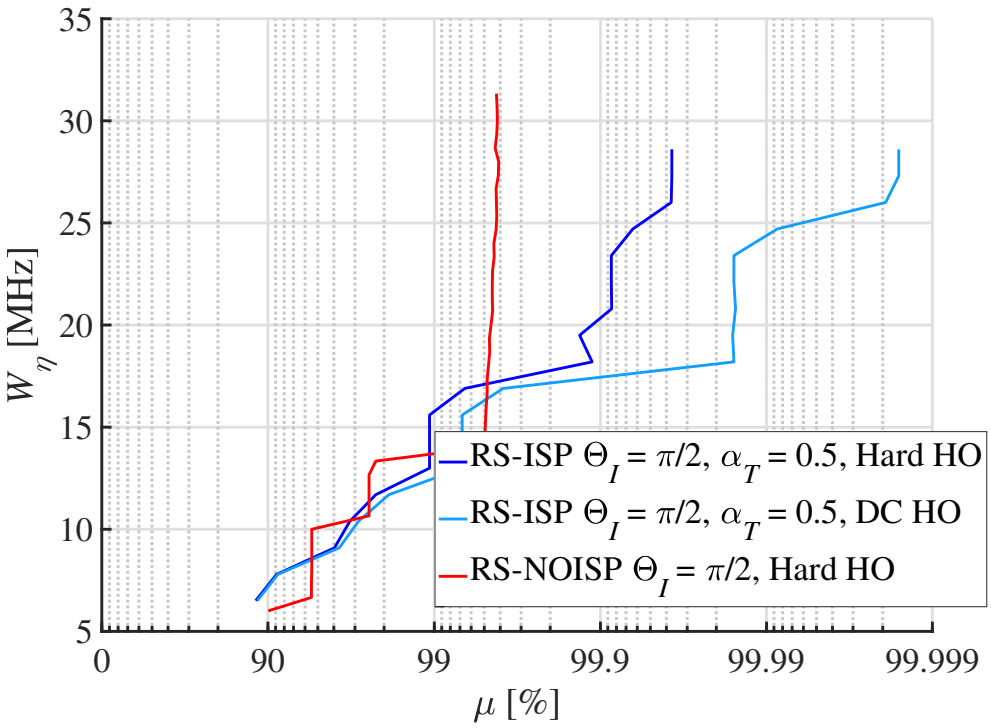


FIGURE 4.22: Performance of Hard handover with respect to DC handover

of death occurring for 1 2 every 1 000 severe epileptic patients per year, and it is estimated to occur in one every 2 000 5 000 Generalized Tonic-Clonic (GTC) seizures, a particular kind of severe convulsive seizure [258]. It has already been shown that continuous daily and overnight monitoring of patients can reduce the frequency of seizures and provide immediate protective mechanisms, thus reducing the risk of SUDEP. The state of the art in clinical monitoring of epileptic patients is performed by specialized caregivers, with the help of a 3D video camera and an ElectroEncephaloGraph (EEG) that measures brain activity. However, the cost of such continuous supervised monitoring has been estimated in the order of thousands of dollars per seizure [258]. Therefore, autonomous decision-support systems for epilepsy management in smart homes represent promising assisted-living solutions for the near future. Thus, in this work, a healthcare solution is envisaged for both continuous monitoring and emergency handling in severe epilepsy. However, the proposed system could be applied also to patients sharing a similar need for pervasive healthcare, e.g., affected by cardiovascular or chronic diseases, or doing rehabilitation at home.

Current cutting-edge communication technologies, including IEEE

802.11ax WLAN and 5G Fixed Wireless Access (FWA) solution, offer a broadband backhaul connectivity even in suburban areas and can support pervasive healthcare at large scale, otherwise not available with networks of the previous generations. Unfortunately, baseline implementations do not provide any priority to healthcare messages over other applications in case of life-threatening events. Network slicing, as applied to WLAN [259] and to heterogeneous architectures exploiting WLAN technology at the radio interface, and to 5G components in the core network [260], fulfills QoS constraints through traffic isolation (e.g., by assigning virtual resources under a common communication infrastructure). This approach, however, has been scarcely investigated in the healthcare context so far, especially to tackle life-threatening events.

In this Section, a new network solution for indoor healthcare monitoring is proposed, in particular for epileptic patients. The novel architecture is composed of various elements. First, a WLAN and a cellular network are cascaded, where IEEE 802.11ax is used in-home and 5G-enabled FWA links transfer them to a remote hospital: this solution is flexible and particularly suitable to serve remote areas, where a fiber link is not available. Second, in order to support both regular monitoring and emergency handling, two new slice types are introduced and the network slicing concept is extended to WLAN. Third, it is proposed to use an enhanced router with local computing capabilities, which is still controlled by the cellular network. The latter, integrated with mobile edge computing resources, makes the resulting architecture more flexible and powerful. Indeed, the local computation capabilities can be exploited to trigger health-related alarms and dynamic network slicing in case of emergency and to provide resource scheduling of both healthcare traffic and other promiscuous everyday communication services. Lastly, the performance of the resulting architecture is demonstrated and compared to baseline solutions.

4.6.1 State of the Art on Epilepsy Management

Recently, other architectures have been presented for continuously managing patients in critical conditions, e.g., severe epileptic patients, at their homes. Most of them employ wearables and portable devices to collect vitals and brain signals, as well as context information, i.e., the patient's location.

The most common solution includes also a MEC server for data analysis using AI algorithms: e.g., in [261], the authors propose an architecture based on LTE and SDN, where an edge gateway is assisted by AI in the localization of the epileptic foci in the brain and delivers effective real-time brain stimulus regulating the epileptic activity while mitigating symptoms. DL algorithms running on a MEC server have been advocated to support the early prediction of epileptic seizures. Although promising, these solutions still lack a realistic in-field deployment (e.g., multiple users, longer distances).

In [262] a two-hop monitoring architecture is proposed, which collects 3D accelerometer traces and the heart rate through a smart bracelet. The latter sends data via Bluetooth low energy (BLE) to a smartphone, which acts as a local gateway to the Internet via WLAN. The minimum E2E latency is 175 ms when serving a single user.

In [263], the authors propose a cloud-based seizure prediction architecture including a Wireless Body Area Network (WBAN), a GPS-based localization, and a localization software hosted in an Amazon elastic compute cloud instance. Several AI-based algorithms have been tested in this study to detect and predict seizures from a low-cost wireless EEG headset. However, the architecture covers a short-range area that can not be considered as a realistic scenario. In [264], edge computing is also proposed to deliver real-time alarms and improve user interaction during emergencies in other IoT-healthcare scenarios, e.g., for preventing falls of elderly people and in mobile healthcare units.

Interestingly, the *HealthEdge* project [265] suggests to prioritize two different types of traffic, i.e., *human behaviour* and *health emergency*, and to decide whether to pre-process data in a MEC server or to send them directly to the cloud. Task scheduling in [265] has been implemented at the edge workstation with benefits on both the bandwidth utilization and the total task processing time. In that case, the edge scheduler directs traffic to either the MEC or the cloud, solely based on the patient's physiological data and no other context information, e.g., current network traffic, is taken into account. At the same time, no strategy for dynamic switching between the two types of traffic has been investigated.

Overall, the literature has not yet investigated the use of slice types specifically designed for e-health applications, and solutions for epilepsy management (or similar scenarios as considered in this Section) are not available. Still, network slicing resource management, admission control, and traffic prioritization in the RAN are relevant problems [66].

4.6.2 Requirements Definition

The case targeted in this Section encompasses, at the same time, two healthcare services: regular monitoring with mild communication requirements in terms of latency, packet drop, and data rate, and emergency handling with strict requirements, activated during life-threatening events. Figure 4.23 shows the data collection setup of this particular use case.

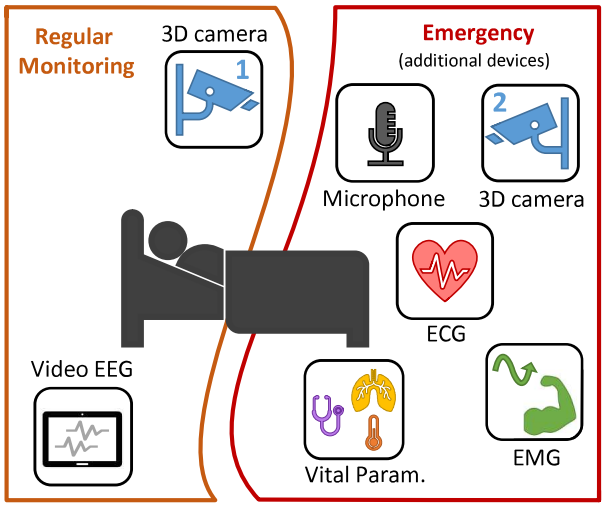


FIGURE 4.23: Data collection setup in case of severe epilepsy management, both during regular monitoring and emergency handling.

In *regular monitoring*, the state-of-art EEG-video acquisition setup [258] is considered, with recording from a 3D camera and 30 EEG channels. During an *emergency*, it is assumed that an alarm has been triggered based on abnormal EEG-video data and the interaction with the patient is extended by adding a 3D camera, a speaker, 3-leads ECG, 2 bipolar electromyography (EMG) channels, a pulse oximeter to measure peripheral oxygen saturation (SpO_2), and a system to acquire the most important vital parameters. This provides the remote specialized clinicians with a better understanding of the situation which, in turn, highly improves the accuracy in detecting SUDEP for a more effective and early intervention.

Table 4.8 summarizes the service requirements, taking into account QoS metrics typically used in the design of 5G systems [232], [266]. It is important to highlight that the values reported in Table 4.8 refer to the communication delays expected during the run-time phase of a network slice instance. Moreover, the additional latency related to the activation and configuration of a new slice is experienced only once. Also, the latter is not correlated to

TABLE 4.8: Communication Requirements of Monitoring Devices

Data type	Sub-type (no. channels)	End-to-end latency ms	Jitter ms	Survival Time ms	Data rate (aggregated)
Regular Monitoring (standard video EEG)					
Multimedia	3D camera 1	150	30	180	10 Mbps
Electrophysiology	EEG (30)	250	25	175	1 Mbps
Emergency Monitoring (additional data)					
Multimedia	3D camera 2	150	30	180	10 Mbps
	Speaker	150	25	175	220 kbps
Electrophysiology	ECG (3)	250	25	275	0.5 Mbps
	EMG (4)	250	25	275	0.5 Mbps
Optics	SpO_2	250	25	275	0.5 Mbps
Vitals	Temperature	250	25	275	100 kbps
	Blood pressure	250	25	275	100 kbps
	Heart rate	250	25	275	100 kbps
	Respiration rate	250	25	275	100 kbps

the survival time in the run-time phase (see Table 4.8) and it is expected to be much smaller than that.

Beyond a high data rate, healthcare monitoring also needs robustness (indicated by the survival time), which is specifically targeted by 5G networks.

4.6.3 Reference Architecture and Proposed Slice Management

In the considered scenario, patients' smart homes are covered by an IEEE 802.11ax WLAN connected to the Internet by an FWA over a 5G cellular network. As a matter of fact, 5G FWA is the most promising alternative, from the techno-economic point of view, to close the digital divide, for urban and suburban areas as well. For healthcare purposes, IEEE 802.11ax devices collect physiological and environmental data in the patient's home while 5G-enabled FWA links transfer them to the remote hospital's cloud servers.

Figure 4.24 depicts the proposed connectivity architecture.

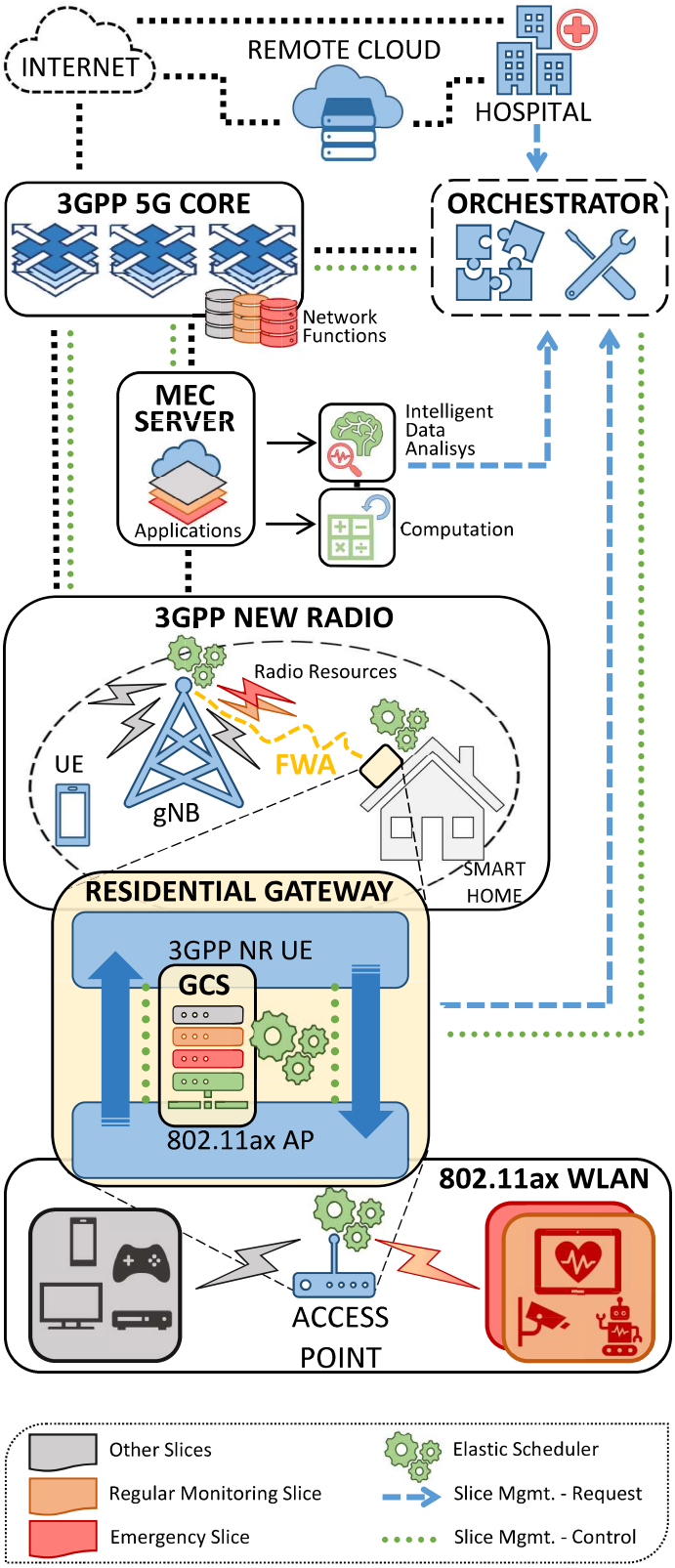


FIGURE 4.24: High-level overview of the proposed architecture.

As both WLAN and FWA networks are also used for other communication purposes in the smart home, no dedicated network deployment is necessary. Still, in order to protect the new services, they are mapped into two new slice types: a *regular monitoring slice-type* when the patient's conditions are stable and an *emergency slice type* for communications during the emergency. The two slices are defined for the cascaded WLAN-FWA networks: therefore, the WLAN router must support the in-home network slicing and should be equipped with a Gateway Computing Server (GCS).

Since network slice types are different in their nature and in their use of a subset of resources available at the radio interface, it is possible to tailor, for each of them, a customized radio resource management scheme that meets their specific requirements. The architecture proposed in our work fully exploits this key capability and natively assumes to implement customized radio resource management schemes for each network slice type.

4.6.3.1 Cellular Network

The cellular network consists of a 3GPP 5G system and an orchestrator entity, which handles the configuration of the 5G network, e.g., it controls the slice life cycle, instantiates slice resources, and assigns traffic routing policy and flow priorities. The MEC server at the edge of the network implements advanced applications for monitoring, classifying, and predicting patients' behaviors, and it supports the orchestrator in the real-time management and configuration of slices and resources.

Among UEs, there are some Residential Gateways (RGs) providing broadband connectivity to smart homes through FWA links. In particular, an RG acts as a gateway between each wireless connected device in the house and the external 5G network. In other words, the RG includes a 3GPP 5G UE and an IEEE 802.11 Access Point (AP). Besides, our architecture includes a novel GCS in the RG, in order to both offload computing task for the STAtions (STAs) and manage the interaction at the 5G-802.11 interface. The GCS plays the role of MEC at an even more local level, providing computational capabilities even closer to the end-user applications, still remaining under the full control of the operator and the service provider. Being at the border between the WLAN and the cellular network, it both responds quicker to in-home events and has privileged access to both WLAN and cellular network resources, including MEC, e.g., for computations on larger databases. MEC and GCS can also share the workload needed to detect anomalies in the EEG monitoring, and to trigger the activation of the emergency slice.

4.6.3.2 Wireless Local Area Network

IEEE 802.11ax is chosen as the WLAN standard since, among other interesting features, it provides centralized scheduling [267]. Furthermore, this work aims at extending the concept of network slicing also in the IEEE 802.11ax network. Indeed, the proposed architecture enables the creation and the dynamic control of network slices spanning from the 5G network to WLANs. A slice-oriented approach provides isolated and independent resources (e.g., radio resources in both WLAN and 5G radio networks, computing resources at the edge, dedicated network functions in 5GC) to each network slice. This significantly extends the possibilities of legacy cellular technologies for realizing QoS differentiation. On one hand, as previously mentioned, each network slice type can benefit from a tailored management scheme of its resources, which may efficiently increase the performance with respect to a one-size-fits-all mechanism. On the other hand, applications requiring AI-based heavy computations with stringent time constraints, e.g., advanced healthcare services, can be offloaded to GCS and MEC.

In addition to the slice management in the cascaded networks, this work proposes an *elastic* radio resource scheduling which operates on both radio technologies, in a distributed and coordinated manner.

4.6.4 Network Slicing Solution

Under stable patient conditions, data collected by the biometric sensors inside the smart home reach the nearest MEC server through the *regular monitoring* slice. The patient's health-related data are analyzed and processed in the MEC server. When an emergency occurs, the RG should be enabled to transmit and receive data on the *emergency* slice. In this case, not only *emergency* data is massively collected to better formulate a diagnosis, but also *control* data could be sent from the MEC to the patient's home, e.g., to alert local caregivers.

4.6.4.1 Dynamic Slicing

As the *emergency* slice type, i.e., a high-priority slice type, is rarely instantiated, it may be inefficient to leave it active, thus penalizing the lower-priority traffic and wasting computing resources on the MEC server. Instead, a *dynamic slicing* approach is considered, wherein a slice can change its type in correspondence of certain events. In this context, a *regular monitoring* slice type is instantiated to carry physiological patient data and environmental

measurements. However, upon a relevant event, e.g., a significant change of any vital parameter or any other life-critical event, the existing slice is promoted to the *emergency* type. This change can be triggered either by the device or by the network. In this latter case, for instance, a certain slice's application, which is virtualized on MEC, may detect an anomaly in regularly monitored vital parameters exploiting AI algorithms. Then, when critical events occur, the MEC server should reach the orchestrator to force the change of slice type to *emergency*. Indeed, for resource management, the proposed architecture still relies on the orchestrator of the 5G network (shown also in Figure 4.24), as from the standard. Note that letting the network change the slice type allows for the use of legacy equipment not directly capable of requesting the setup of new slices, as the slice type is set by the network. Interestingly, *dynamic slicing* provides efficient management not only for slice activation but also for its deactivation, when an emergency is solved. Finally, it is important to highlight that the existing literature on dynamic network slicing considers the dynamic creation of new slices as limited to the current 3GPP paradigm, i.e., for slowly reactive systems (as extensively discussed in the very first Sections of this Chapter).

4.6.4.2 Elastic Resource Scheduling

In order to meet the QoS requirements on an E2E basis, the RG is envisioned to control the scheduling of the WLAN, thus enforcing the slice requirements inside the smart home, too. As a matter of fact, the WLAN component is typically unaware of the 5G network slicing. Indeed, both AP and STAs' knowledge of the network is limited to the WLAN, hence ignoring how different slices are configured in the 5G segment. It is thus necessary that the GCS implements a mapping function that translates the requirements of a specific slice/service type into a WLAN service class and efficiently schedules and manages the traffic flows within the smart home. For instance, packets of the emergency slice are assigned a higher priority, in order to ensure reliable and low-latency services in critical situations. In this way, the network slicing should be enforced for the cascaded IEEE 802.11ax and 5G networks.

The slice QoS requirements are satisfied by a novel *elastic* radio resource scheduling policy. Thanks to the GCS, the radio resource management policy handles different queues within the RG, one for each slice. Moreover, the radio resource allocation must satisfy the requirements on both the maximum E2E latency and the survival time (see Table 4.8). For this reason, within both the IEEE 802.11ax and 5G networks, the scheduling algorithm distributes the

radio resources taking also into account the queuing delay across the cascaded network. Therefore, the proposed solution requires the devices to communicate the experienced queuing delay, along with their buffer status report.

In the uplink, as soon as a packet is correctly received by the AP and the GCS pushes it in the related queue of the RG, its accumulated delay is tracked. Each queue is sorted according to the packet delay and expired queued packets are dropped. As a result, the accumulated delay in the two-tier segment is taken into account in scheduling resources.

Furthermore, the envisioned scheduler determines the number of radio resources requested by each flow, hence by each slice, in a given time window T. This evaluation is based on the agreed QoS parameters (e.g., average transmission rate), as well as on the channel conditions experienced by the users, and it is conducted in both networks, i.e., in the RAN and in the WLAN. When the number of requested radio resources can be satisfied in T, the scheduling follows the Modified-Largest Weighted Delay First (M-LWDF) approach. Since the delay of each packet is tracked by the RGs, the M-LWDF scheduler aims at satisfying the QoS requirements on an E2E basis. Conversely, when the number of available radio resources in T does not match the requests, an elastic resource scaling is first applied. The elastic scaling proportionally reduces the number of available radio resources for each active slice instance in T, according to the resources surplus requested by each slice. After scaling, slices are grouped into two sets, i.e., with priority and without priority. Then, in each scheduling interval, radio resources are assigned first to packets of slices with priority, and then to those with no priority.

4.6.5 A Case Study

A European suburb is modeled according to the reference FWA scenario of [268], including 1 000 households per km^2 and a grid of three-sector macro sites with an average inter-site distance of about 1 km. The network is designed to connect simultaneously up to 30% of the covered households, thus each sector serves approximately 88 households. Each sector is equipped with 64 transmit/receive antennas, working in the sub-6 GHz band and each RG has 2 transmit and 4 receive antennas. Therefore, up to 16 different spatial layers may be multiplexed on a single RB. All households also generate

uplink traffic for a generic eMBB slice according to the models in [269].

As about 1.8 million people in Europe with epilepsy are at risk of SUDEP, there are 2 patients per sector, on average, to manage. In the worst-case scenario, it is assumed one epileptic patient at high risk of SUDEP and another epileptic patient with no risk of SUDEP to be monitored at the very same time in a single sector. Healthcare traffic flows are generated according to the specifics of Table 4.8.

In each household, single-user MIMO is considered for IEEE 802.11ax, with a single spatial stream per STA. Legacy STAs are not explicitly modeled, although it is assumed that the AP reserves 30% of the time for legacy transmissions and extra signal processing delay. Based on the channel conditions, the APs and the RG select the appropriate MCS, in order to guarantee a target maximum BLER through link adaptation. Moreover, both the transport block size for 5G and the data rate for 802.11ax networks are set accordingly, following the standards. It is assumed a higher BLER value for the IEEE 802.11ax link, in order to take into account the interference and possibly busy channels, as each STA performs carrier sensing and the transmission is canceled whenever the medium is busy, resulting in transmission delays.

The proposed scheduling solution (reported in the following as *Elastic*) has been compared with:

- a solution performing no slicing at all (reported in the following as *Basic*), without any proper resource scheduling policy, i.e., proportional fair scheduling is used at all nodes;
- a network slicing solution (reported in the following as *E2E*), including slicing within the WLAN and the use of the M-LWDF scheduler for each slice, although without the elastic resource scheduling, i.e., neither traffic prioritization, expired packets management, nor resource scaling.

By comparing these solutions, it is possible to appreciate the advantages of using network slicing and the introduced elastic resource scheduling.

The impact of scheduling strategies over the E2E latency, the communication service availability, and the number of emergency slice packets meeting the required QoS, has been investigated by computer simulations. In the following, α and β are weights to further manage priorities associated with the eMBB slice and the healthcare slices, respectively: a higher value of these parameters means a higher priority of the corresponding slice.

4.6.5.1 End-to-End Latency

First, it is considered the average E2E latency (from the in-home device to the gNB), taking into account the delay introduced by the radio interfaces of the two networks for the considered slice types.

Figure 4.25 shows the average E2E latency for the various slice types, the different scheduling techniques, and two traffic loads (30% and 50% active RGs).

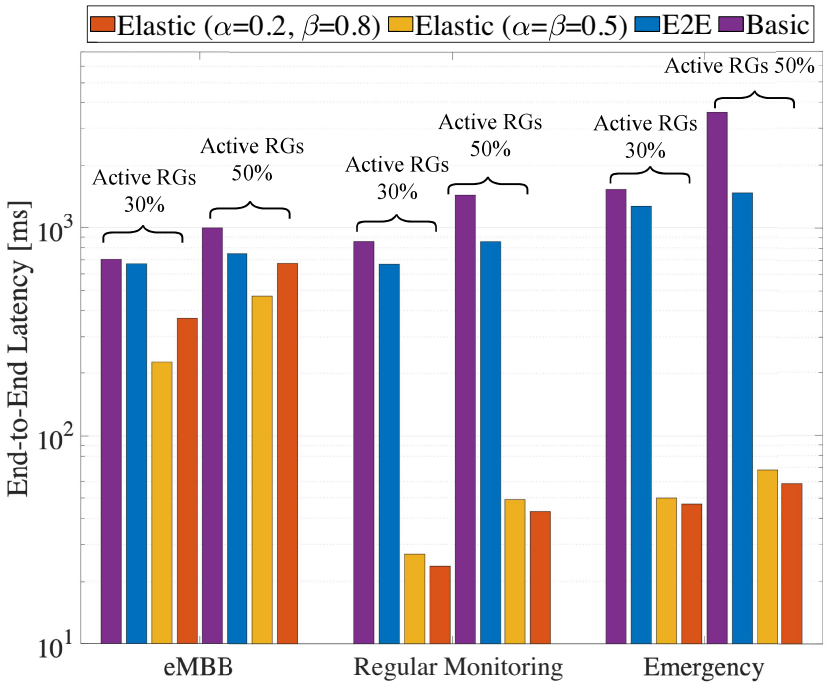


FIGURE 4.25: Average E2E latency when 30% of the RGs (low traffic load) and 50% of the RGs (high traffic load) are active.

It is noticeable that both the basic and the E2E scheduling yield a higher latency when the emergency slices are active since these scheduling approaches do not distinguish among the different traffic types and the penalty incurred by having higher loads is shared equally among all applications. Moreover, as eMBB traffic is higher than healthcare traffic, the proportional fair scheduling penalizes the regular monitoring and emergency slices. This effect is just slightly mitigated in the E2E approach that uses M-LWDF scheduling, being more sensitive to the packet latency. When elastic scheduling is considered, instead, regular monitoring and emergency traffics are served with much lower latency, as required by their specifics. In fact, the proposed approach guarantees a smaller delay than with other strategies,

by dropping expired packets. At the same time, it is also important to note that the elastic scheduling slightly reduces the latency of the eMBB slice type. Moreover, by adjusting the values of α and β, it is possible to further control the priority of the healthcare slices with respect to the eMBB slice. When considering different loads (30% and 50% active RGs), note that latency grows with the load for all slice types, when the basic and the E2E schemes are used. Instead, when the elastic solution is adopted, the latency changes only slightly for the emergency slice type, thus ensuring the required QoS anyway. This confirms the robustness of the proposed solution to the traffic load (i.e., the percentage of active RGs).

4.6.5.2 Communication Service Availability

As already reported, the communication service availability is the ratio between the time wherein the service is delivered according to an agreed QoS and the time expected to deliver it. In this scenario, the system is considered unavailable whenever a message is not received within the survival time (the sum of the E2E latency and the jitter), which is considered as the maximum acceptable delay.

Figure 4.26 shows the probability that the service availability, namely A, is larger than 0.99, thus matching the requirements of Table 4.8.

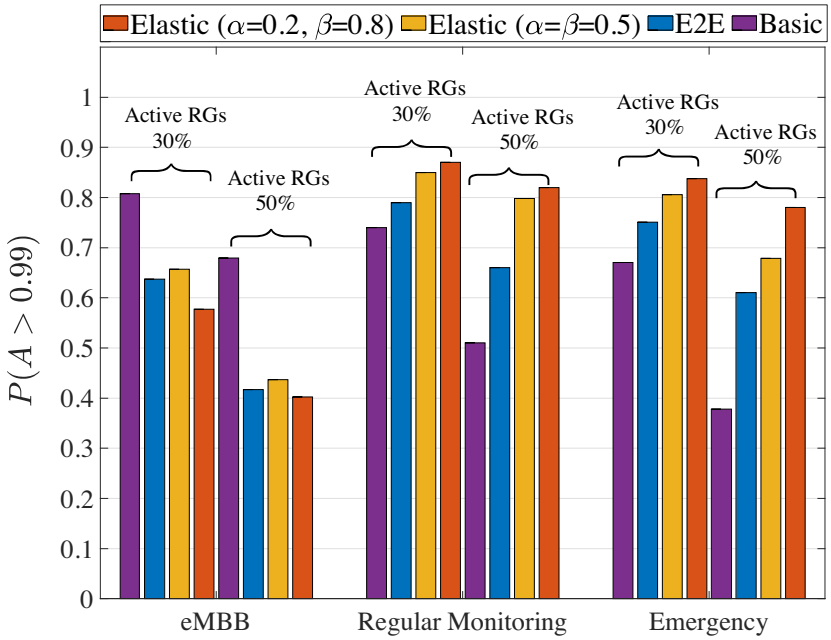


FIGURE 4.26: Probability that the communication service availability (A) is larger than 0.99 when 30% and 50% of the RGs are active (low and high traffic loads, respectively).

First, note that the availability dramatically decreases for both regular and emergency slices when the basic solution is used since the proportional fair scheduling penalizes the healthcare slices under a heavier load (50% of RGs). The E2E solution clearly provides higher availability for both healthcare slices, since the wireless networks are sliced and the M-LWDF is adopted to schedule the traffic, but it fails to provide decent performance for higher loads (50% of RGs). The proposed elastic approach, instead, ensures high availability for both healthcare slices in all load conditions (both 30% and 50% of active RGs), at the cost of reduced availability of the eMBB slice under a high load (50% of RGs), due to the limited resources of the network. It is important to highlight that the difference, in terms of performance, between the elastic and the E2E strategies, is due to, on one hand, the elastic scaling of resources, and, on the other, the expired packets dropping.

4.6.5.3 Packets meeting the QoS

To provide further insight, the percentage of the *emergency* slice packets meeting the required QoS is considered, as shown in Figure 4.27.

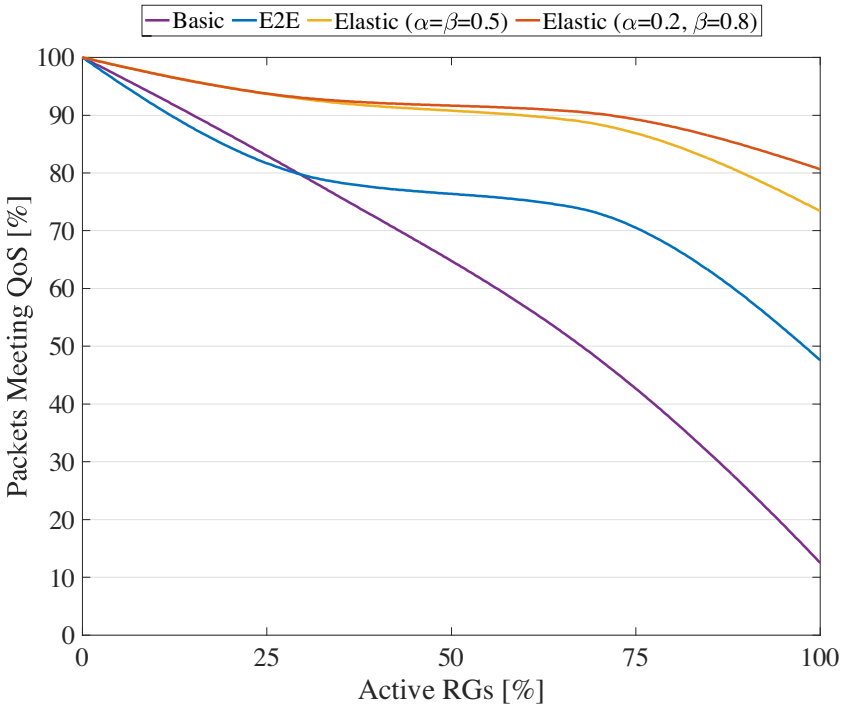


FIGURE 4.27: Percentage of the *emergency* slice packets meeting the required QoS.

Note that, when the basic scheme is used, the percentage of packets meeting the QoS requirements decreases linearly with the percentage of active RGs, being thus inadequate to support the healthcare traffic. The E2E solution yields a performance improvement, but it still has a linear decay for a small percentage of active RGs, with the percentage of packets meeting the QoS quickly dropping below 80%. Hence, also E2E is not a scalable solution. The elastic scheduling technique instead keeps the percentage of *emergency* packets meeting the QoS above 90% up to 75% of active RGs, with an overall slow decay. Even when all RGs are active, the elastic solution ensures that 80% of emergency packets meet the QoS requirement, while E2E scheme supports less than half of the emergency packets and the basic approach properly serves only 11% of the packets.

In conclusion, the proposed solution, based on new slice types and elastic scheduling, is robust to the traffic load, ensuring the required QoS for healthcare services.

Conclusions and Future Research Directions

The 5G and B5G of mobile technology have begun to revolutionize the existing wireless network, driven by the increasing demand for mobile data traffic and the tremendous growth in connectivity. These networks are characterized by embedded flexibility to optimize the network usage in order to accommodate a wide range of advanced use cases in an agile and cost-efficient manner, and for boosting the performance in terms of throughput, latency, reliability, density, and mobility. New research directions led to fundamental changes in the design of 5G and B5G cellular networks.

To this end, this work pursued the goal of presenting several cutting-edge management techniques and simulation models for 5G & Beyond RANs. Particularly, Chapter 1 described a wide range of advanced 5G services and use cases, while providing several details on both the 3GPP NR interface and the enabling technologies.

Then, Chapter 2 deeply presented an open-source simulation framework for the 5G air interface, 5G-air-simulator, as an instrument to study a number of technical components already standardized by the 3GPP, under investigation by other standardization entities, or discussed in the scientific literature. Furthermore, given both the open-source nature of the tool and its modularity, any new technical components may be integrated in order to pave the way for different research directions. As part of future research directions, technical components to be included in the 5G-air-simulator are listed (but not limited to) below. URLLC will facilitate several new services and verticals, but these applications rise tricky challenges in terms of latency, reliability, availability, and security [12], [270], hence requiring further investigations. It is important to consider the different time structures of radio resources for supporting these new services. Moreover, the introduction of pre-emption based puncturing in Schedulers is foreseen. In addition, UAVs play an important role in 5G wireless technologies [271]. On the one hand, UAVs may be leveraged in intelligent heterogeneous architecture for enhancing cells' capacity or providing network service recovery [272]. On the other

hand, thanks to the capabilities offered by the NR, it is possible to realize large-scale UAVs deployment, hence presenting new research challenges as well as opportunities [60]. For the support of UAVs in the 5G-air-simulator, it is essential to develop new 3D mobility and radio channel models [273], as well as new network deployments for the envisioned heterogeneous architectures. D2D communication also emerged as a key technical component to offload the traffic from cellular networks exploiting direct links. Sidelink communication indeed reduces the computational complexity at the base station and enhances the cell's capacity. Even though few basic D2D features have been standardized for 4G, they are currently under discussion in 3GPP, specifically for enhanced-V2X services [274]. However, a number of challenges and research directions are still open [61]. The main modifications should be made for carrying out autonomously the operations normally controlled by the base station, such as resource allocation and subsequent data transmissions. Finally, mmWave communications, which are currently under investigation of the 3GPP [25], have become increasingly important for their ability to provide high-throughput and low-latency. They extensively exploit large antenna arrays and adaptive beamforming to achieve highly directional transmissions. However, this approach has a disruptive impact not only on the physical layer but also on the other layers of the protocol stack, hence affecting several features, such as cell search, broadcast signaling, random access, etc. [62]. For this reason, the design of mmWave 5G systems still requires considerable research efforts [77]. For the support of mmWave communications, it is important to properly integrate new channel models, as well as to add extended-bandwidth features. In parallel, future work includes a number of new supporting models in order to enhance the present technical components and enable future ones. Several enhancements may include Bandwidth Parts, a FrameManager extension to support different numerologies in different subframes, HARQ activation/deactivation per flow, new handover procedures, an extension of the NB-IoT module with downlink transmission and in-band coexistence with 5G, as well as Supplementary Uplink, and the introduction of the new 5G Service Data Adaptation Protocol (SDAP).

Chapter 3 presented the implementation of NB-IoT in 5G-air-simulator, for both traditional and NTNs, as well as a preliminary analytical model describing the random access procedure. Regarding this latter aspect, future activities could involve, on the one hand, the formulation of a complete analytical model to estimate the number of users performing the random access

procedure. On the other, either entirely new random access mechanisms or major enhancements to the current random access protocol could be implemented in the proposed simulation tool, in order to assess their performance with respect to the standard procedure. As for the integration of the NB-IoT technologies in 5G-air-simulator, future activities intend to extend the developed tool with additional features, like the support of simultaneous Single-Tone and Multi-Tone transmission schemes, more accurate channel and interference models for all the standardized NB-IoT operation modes. Moreover, future work may also extend the investigation in the context of SatCom, in order to evaluate the impact of the satellite constellation configuration on more complex network topologies, the performance of a multi-tone uplink and downlink channel configuration, as well as the analysis of the energy consumption. Finally, the proposed simulation platform could be used for conducting the performance evaluation of different application scenarios, hence embracing the huge extent of Internet of Things use cases.

Chapter 4 investigated the issues related to RAN Slicing, which notably improves the E2E performance, thanks to the flexibility introduced in NR and the virtualization of the network. Moreover, it gave the basis for the design of a comprehensive architecture enabling RAN slicing for Latency Sensitive Services, while posing particular attention to design criteria, system components, and their baseline interactions. The identification of the readiness level of the underlying technologies paves the way towards future research activities on the reference themes. The most relevant issues that affect RAN slicing for latency-sensitive services, as well as interesting research activities to address in the future, are listed in what follows. Given the 5G and B5G heterogeneous context, the architecture should take into account different time scales for slice generation. At the same time, granularity constraints in spectrum- and radio-level resource sharing are a primary concern. Appropriate APIs between IP and TNT are a key feature for boosting the performance and achieving extreme flexibility. Moreover, it is of the utmost importance to properly dimension the performance reporting between the IP and TNTs (e.g., periodical, triggered by events, or both). Indeed, too much information may produce undesirable control overhead. Security threats should be utterly reduced in the entire architecture, and specifically in the MEC servers. Finally, Net Neutrality should always be guaranteed, especially in RAN slicing contexts, which tend to prefer the most involved TNTs in a *rich-get-richer* fashion.

Overall, it should be noted that this thesis has attempted to explore only a fraction of the 5G and B5G world. Nonetheless, in conclusion, it is of paramount importance to underline the two main lessons learned. First, starting from 5G, any new communication technology needs to have sufficient flexibility in its design to be able to adapt to needs that were not anticipated at the time it was designed, and sufficient potential to enable innovation beyond the imagination of today. Second, it is important to reconsider the traditional notion of a "generational" change driven solely by advancements in radio and core technologies. Since mobile networks are expected to become even more critical as the role of communication networks expands in every aspect of society, the actual impact of future communications technologies would be far broader in scope and larger in scale, limited only by human imagination and creativity in applying these technologies for the benefit of all. Therefore, the focus of future standards and technology development should become broader in scope, but also more incremental and agile in detail. History all too eloquently teaches us that this will lead again to the urgent necessity of a brand new generation of mobile communication: 6G.

Acknowledgements

I would like to acknowledge my gratitude for my supervisor, Prof. Gennaro Boggia, and my colleague, Giuseppe Piro, for their thorough grounding in the research topics covered in this thesis and for awakening my interest in them.

Bibliography

[1] M. R. Bhalla and A. V. Bhalla, "Generations of mobile wireless technology: A survey," *International Journal of Computer Applications*, vol. 5, no. 4, pp. 26–32, 2010.

[2] ITU, "IMT Vision – Framework and overall objectives of the future development of IMT for 2020 and beyond," International Telecommunication Union (ITU), Recommendation 2083-0, 2015, ITU-R M.2083-0.

[3] E. Soltanmohammadi, K. Ghavami, and M. Naraghi-Pour, "A Survey of Traffic Issues in Machine-to-Machine Communications Over LTE," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 865–884, 2016.

[4] M. Lauridsen, L. C. Gimenez, I. Rodriguez, T. B. Sørensen, and P. E. Mogensen, "From LTE to 5G for Connected Mobility.," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 156–162, 2017.

[5] M. S. Elbamby, C. Perfecto, M. Bennis, and K. Doppler, "Toward Low-Latency and Ultra-Reliable Virtual Reality," *IEEE Network*, vol. 32, no. 2, pp. 78–84, 2018.

[6] E. Dahlman, G. Mildh, S. Parkvall, J. Peisa, J. Sachs, Y. Selén, and J. Sköld, "5G wireless access: requirements and realization," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 42–47, 2014.

[7] D. Jiang, H. Wang, E. Malkamaki, and E. Tuomaala, "Principle and Performance of Semi-Persistent Scheduling for VoIP in LTE System," in *2007 International Conference on Wireless Communications, Networking and Mobile Computing*, IEEE, 2007, pp. 2861–2864.

[8] E. Hossain, M. Rasti, H. Tabassum, and A. Abdelnasser, "Evolution toward 5G multi-tier cellular wireless networks: An interference management perspective," *IEEE Wireless Communications*, vol. 21, no. 3, pp. 118–127, 2014.

[9] Patrik Cerwall, "Ericsson Mobility Report - November," Tech. Rep., Nov. 2019.

[10] A. Adhikary, X. Lin, and Y.-P. E. Wang, "Performance evaluation of NB-IoT coverage," in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, IEEE, 2016, pp. 1–5.

[11] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5G: Physical and MAC-layer solutions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, 2016.

[12] M. Bennis, M. Debbah, and H. V. Poor, "Ultrareliable and Low-Latency Wireless Communication: Tail, Risk, and Scale," *Proceedings of the IEEE*, vol. 106, no. 10, pp. 1834–1853, Oct. 2018.

[13] S. Chen, J. Hu, Y. Shi, Y. Peng, J. Fang, R. Zhao, and L. Zhao, "Vehicle-to-everything (v2x) services supported by lte-based systems and 5g," *IEEE Communications Standards Magazine*, vol. 1, no. 2, pp. 70–76, 2017.

[14] J. Harris, M. Beach, A. Nix, and P. Thomas, "The Potential of Offloading and Spectrum Sharing for 5G Vehicular Infotainment," in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, IEEE, 2016, pp. 1–5.

[15] J. Wang, J. Liu, and N. Kato, "Networking and Communications in Autonomous Driving: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1243–1274, 2018.

[16] M. Boban, K. Manolakis, M. Ibrahim, S. Bazzi, and W. Xu, "Design aspects for 5G V2X physical layer," in *Standards for Communications and Networking (CSCN), 2016 IEEE Conference on*, IEEE, 2016, pp. 1–7.

[17] L. Militano, M. Condoluci, G. Araniti, A. Molinaro, A. Iera, and G.-M. Muntean, "Single Frequency-Based Device-to-Device-Enhanced Video Delivery for Evolved Multimedia Broadcast and Multicast Services," *IEEE Transactions on Broadcasting*, vol. 61, no. 2, pp. 263–278, 2015.

[18] G. Araniti, M. Condoluci, P. Scopelliti, A. Molinaro, and A. Iera, "Multicasting over emerging 5G networks: Challenges and perspectives," *IEEE Network*, vol. 31, no. 2, pp. 80–89, 2017.

[19] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Further advancements for E-UTRA physical layer aspects," 3rd Generation Partnership Project, Tech. Rep. 36814, Mar. 2010.

[20] J. J. Gimenez, D. Gomez-Barquero, J. Morgade, and E. Stare, "Wideband Broadcasting: A Power-Efficient Approach to 5G Broadcasting," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 119–125, 2018.

[21] O. Kodheli, E. Lagunas, N. Maturo, S. K. Sharma, B. Shankar, J. F. M. Montoya, J. C. M. Duncan, D. Spano, S. Chatzinotas, S. Kisseleff, J. Querol, L. Lei, T. X. Vu, and G. Goussetis, "Satellite Communications in the New Space Era: A Survey and Future Challenges," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2020. DOI: 10.1109/COMST.2020.3028247.

[22] A. Ghosh, A. Maeder, M. Baker, and D. Chandramouli, "5G Evolution: A View on 5G Cellular Technology Beyond 3GPP Release 15," *IEEE Access*, vol. 7, pp. 127 639–127 651, 2019. DOI: 10.1109/ACCESS.2019.2939938.

[23] 3GPP, "5G; NR; Base Station (BS) radio transmission and reception (Release 15)," 3rd Generation Partnership Project, Tech. Rep. 38104, Jul. 2018.

[24] ——, "5G; NR; Physical channels and modulation (Release 15)," 3rd Generation Partnership Project, Tech. Rep. 38211, Jan. 2020.

[25] ——, "5G; NR; Overall description (Release 15)," 3rd Generation Partnership Project, Tech. Rep. 38300, Apr. 2020.

[26] A. A. Zaidi, R. Baldemair, V. Molés-Cases, N. He, K. Werner, and A. Cedergren, "Ofdm numerology design for 5g new radio to support iot, embb, and mbsfn," *IEEE Communications Standards Magazine*, vol. 2, no. 2, pp. 78–83, 2018.

[27] J. Campos, "Understanding the 5g nr physical layer," *Keysight Technologies release*, 2017.

[28] K. Pedersen, G. Pocovi, J. Steiner, and A. Maeder, "Agile 5g scheduler for improved e2e performance and flexibility for different network implementations," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 210–217, 2018.

[29] T. Inoue, "5g nr release 16 and millimeter wave integrated access and backhaul," in *2020 IEEE Radio and Wireless Symposium (RWS)*, IEEE, 2020, pp. 56–59.

[30] 3GPP, "Study on NR-based access to unlicensed spectrum (Release 16)," 3rd Generation Partnership Project, Tech. Rep. 38889, Dec. 2018.

[31] Nokia, "Key directions for Release 17," 3rd Generation Partnership Project, Tech. Rep. RP-190831, Jun. 2019.

[32] E. Au, "A short update on 3gpp release 16 and release 17 [standards]," *IEEE Vehicular Technology Magazine*, vol. 15, no. 2, pp. 160–160, 2020.

[33] Y. Li and M. Chen, "Software-defined network function virtualization: A survey," *IEEE Access*, vol. 3, pp. 2542–2553, 2015.

[34] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," *IEEE Communications Magazine*, vol. 53, no. 2, pp. 90–97, 2015.

[35] D. Kreutz *et al.*, "Software-Defined Networking: A Comprehensive Survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2014. DOI: 10.1109/JPROC.2014.2371999.

[36] M. Agiwal, A. Roy, and N. Saxena, "Next generation 5g wireless networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1617–1655, 2016.

[37] Q. Pham *et al.*, "A Survey of Multi-Access Edge Computing in 5G and Beyond: Fundamentals, Technology Integration, and State-of-the-Art," *IEEE Access*, vol. 8, pp. 116 974–117 017, 2020. DOI: 10.1109/ACCESS.2020.3001277.

[38] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A Survey on Low Latency Towards 5G: RAN, Core Network and Caching Solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3098–3130, 2018. DOI: 10.1109/COMST.2018.2841349.

[39] ETSI, "Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements," European Telecommunications Standards Institute, Group Specification (GS) V2.1.1, Oct. 2018.

[40] ——, "Multi-access Edge Computing (MEC); Framework and Reference Architecture," European Telecommunications Standards Institute, Group Specification (GS) MEC 003, Dec. 2020, V2.2.1.

[41] 3GPP, "5G; Management and orchestration; Concepts, use cases and requirements," 3rd Generation Partnership Project, Tech. Specification (TS) 28.530, Oct. 2019, V15.2.0.

[42] X. Zhou, R. Li, T. Chen, and H. Zhang, "Network slicing as a service: Enabling enterprises' own software-defined cellular networks," *IEEE Communications Magazine*, vol. 54, no. 7, pp. 146–153, 2016.

[43] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5g: Survey and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94–100, 2017.

[44] S. Zhang, "An overview of network slicing for 5G," *IEEE Wireless Communications*, vol. 26, no. 3, pp. 111–117, 2019.

[45] M. Maule, J. Vardakas, and C. Verikoukis, "5G RAN Slicing: Dynamic Single Tenant Radio Resource Orchestration for eMBB Traffic within a Multi-Slice Scenario," *IEEE Communications Magazine*, vol. 59, no. 3, pp. 110–116, 2021.

[46] A. A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines, "5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges," *Computer Networks*, vol. 167, p. 106 984, 2020. DOI: 10.1016/j.comnet.2019.106984.

[47] J. Mei, X. Wang, and K. Zheng, "An intelligent self-sustained RAN slicing framework for diverse service provisioning in 5G-beyond and 6G networks," *Intelligent and Converged Networks*, vol. 1, no. 3, pp. 281–294, 2020.

[48] S. D'Oro, F. Restuccia, and T. Melodia, "Toward operator-to-waveform 5g radio access network slicing," *IEEE Communications Magazine*, vol. 58, no. 4, pp. 18–23, 2020.

[49] Ö. U. Akgül, I. Malanchini, and A. Capone, "Dynamic resource trading in sliced mobile networks," *IEEE Transactions on Network and Service Management*, vol. 16, no. 1, pp. 220–233, 2019.

[50] X. Zhang, B. Li, J. Peng, X. Pan, and Z. Zhu, "You Calculate and I Provision: A DRL-Assisted Service Framework to Realize Distributed and Tenant-Driven Virtual Network Slicing," *Journal of Lightwave Technology*, vol. 39, no. 1, pp. 4–16, 2021.

[51] X. Zhang, W. Lu, B. Li, and Z. Zhu, "Drl-based network orchestration to realize cooperative, distributed and tenant-driven virtual network slicing," in *2019 Asia Communications and Photonics Conference (ACP)*, 2019, pp. 1–3.

[52] K. Zheng *et al.*, "Big Data-Driven Optimization for Mobile Networks Toward 5G," *IEEE Network*, vol. 30, pp. 44–51, 2016. DOI: 10.1109/MNET.2016.7389830.

[53] C. Jiang *et al.*, "Machine Learning Paradigms for Next-Generation Wireless Networks," *IEEE Wireless Communications*, vol. 24, no. 2, pp. 98–105, Apr. 2017, ISSN: 1536-1284. DOI: 10.1109/MWC.2016.1500356WC.

[54] C. Zhang, P. Patras, and H. Haddadi, "Deep Learning in Mobile and Wireless Networking: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2224–2287, Sep. 2019, ISSN: 2373-745X. DOI: 10.1109/COMST.2019.2904897.

[55] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of Machine Learning in Wireless Networks: Key Techniques and Open Issues," *IEEE Communications Surveys & Tutorials*, 2019. DOI: 10.1109/COMST.2019.2924243.

[56] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[57] N. C. Luong *et al.*, "Applications of Deep Reinforcement Learning in Communications and Networking: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133–3174, 2019. DOI: 10.1109/COMST.2019.2916583.

[58] Y. Wang, J. Xu, and L. Jiang, "Challenges of system-level simulations and performance evaluation for 5G wireless networks," *IEEE Access*, vol. 2, pp. 1553–1561, 2014.

[59] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An Overview of Massive MIMO: Benefits and Challenges," *IEEE journal of selected topics in signal processing*, vol. 8, no. 5, pp. 742–758, 2014.

[60] Y. Zeng, J. Lyu, and R. Zhang, "Cellular-connected uav: Potential, challenges, and promising technologies," *IEEE Wireless Communications*, vol. 26, no. 1, pp. 120–127, 2018.

[61] R. I. Ansari, C. Chrysostomou, S. A. Hassan, M. Guizani, S. Mumtaz, J. Rodriguez, and J. J. Rodrigues, "5g d2d networks: Techniques, challenges, and future prospects," *IEEE Systems Journal*, vol. 12, no. 4, pp. 3970–3984, 2017.

[62] S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter wave cellular wireless networks: Potentials and challenges," *arXiv preprint arXiv:1401.2560*, 2014.

[63] A. Guidotti, A. Vanelli-Coralli, M. Conti, S. Andrenacci, S. Chatzinotas, N. Maturo, B. Evans, A. Awoseyila, A. Ugolini, T. Foggi, L. Gaudio, N. Alagha, and S. Cioni, "Architectures and Key Technical Challenges for 5G Systems Incorporating Satellites," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2624–2639, 2019. DOI: 10.1109/TVT.2019.2895263.

[64] S. Zhang and D. Zhu, "Towards artificial intelligence enabled 6g: State of the art, challenges, and opportunities," *Computer Networks*, p. 107 556, 2020.

[65] A. Ahad, M. Tahir, and K.-L. A. Yau, "5G-based Smart Healthcare Network: Architecture, Taxonomy, Challenges and Future Research Directions.," *IEEE Access*, 2019.

[66] L. U. Khan, I. Yaqoob, N. H. Tran, Z. Han, and C. S. Hong, "Network slicing: Recent advances, taxonomy, requirements, and open research challenges," *IEEE Access*, vol. 8, pp. 36 009–36 028, 2020.

[67] A. Azari, M. Ozger, and C. Cavdar, "Risk-Aware Resource Allocation for URLLC: Challenges and Strategies with Machine Learning," *IEEE Communications Magazine*, vol. 57, no. 3, pp. 42–48, Mar. 2019.

[68] F. D. Calabrese, L. Wang, E. Ghadimi, G. Peters, and P. Soldati, "Learning Radio Resource Management in 5G Networks: Framework, Opportunities and Challenges," *arXiv:1611.10253 [cs]*, 2017.

[69] S. E. Elayoubi, S. B. Jemaa, Z. Altman, and A. Galindo-Serrano, "5g ran slicing for verticals: Enablers and challenges," *IEEE Communications Magazine*, vol. 57, no. 1, pp. 28–34, 2019.

[70] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca, and J. Folgueira, "Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 80–87, 2017.

[71] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges," *Future Generation Computer Systems*, vol. 78, pp. 680–698, 2018.

[72] E. J. Kitindi, S. Fu, Y. Jia, A. Kabir, and Y. Wang, "Wireless network virtualization with SDN and C-RAN for 5G networks: Requirements, opportunities, and challenges," *IEEE Access*, vol. 5, pp. 19 099–19 115, 2017.

[73] T. K. Rodrigues, K. Suto, H. Nishiyama, J. Liu, and N. Kato, "Machine Learning Meets Computation and Communication Control in Evolving Edge and Cloud: Challenges and Future Perspective," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 38–67, 2020, ISSN: 2373-745X. DOI: 10.1109/COMST.2019.2943405.

[74] S. Cho, S. Chae, M. Rim, and C. G. Kang, "System level simulation for 5G cellular communication systems," in *Ubiquitous and Future Networks (ICUFN), 2017 Ninth International Conference on*, IEEE, 2017, pp. 296–299.

[75] I.-P. Belikaidis, A. Georgakopoulos, E. Kosmatos, I. de-la-Bandera, D. Palacios, R. Barco, and P. Demestichas, "5G Component Carrier Management Evaluation by Means of System Level Simulations," in *2019 European Conference on Networks and Communications (EuCNC)*, IEEE, 2019, pp. 592–596.

[76] M. Liu, P. Ren, Q. Du, W. Ou, X. Xiong, and G. Li, "Design of system-level simulation platform for 5G networks," in *Communications in China (ICCC), 2016 IEEE/CIC International Conference on*, IEEE, 2016, pp. 1–6.

[77] M. Mezzavilla, M. Zhang, M. Polese, R. Ford, S. Dutta, S. Rangan, and M. Zorzi, "End-to-end simulation of 5g mmwave networks," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2237–2263, 2018.

[78] N. Patriciello, S. Lagen, B. Bojovic, and L. Giupponi, "An e2e simulator for 5g nr networks," *Simulation Modelling Practice and Theory*, vol. 96, p. 101 933, 2019.

[79] M. K. Müller, F. Ademaj, T. Dittrich, A. Fastenbauer, B. R. Elbal, A. Nabavi, L. Nagel, S. Schwarz, and M. Rupp, "Flexible multi-node simulation of cellular mobile communications: the Vienna 5G System Level Simulator," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, p. 227, 2018.

[80] M. Han, J. W. Lee, C. G. Kang, and M. J. Rim, "5g k-simsys: Open/-modular/flexible system level simulator for 5g system," in *2018 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, IEEE, 2018, pp. 1–2.

[81] Tetcos. (2019). "NetSim-Network Simulator & Emulator." [Online; accessed 31 Oct. 2021], [Online]. Available: https://www.tetcos.com/.

[82] C.-K. Jao, C.-Y. Wang, T.-Y. Yeh, C.-C. Tsai, L.-C. Lo, J.-H. Chen, W.-C. Pao, and W.-H. Sheen, "Wise: A system-level simulator for 5g mobile networks," *IEEE Wireless Communications*, vol. 25, no. 2, pp. 4–7, 2018.

[83] S. Foni, T. Pecorella, R. Fantacci, C. Carlini, P. Obino, and M.-G. Di Benedetto, "Evaluation methodologies for the NB-IOT system: issues and ongoing efforts," in *Proc. of AEIT International Annual Conference*, IEEE, 2017.

[84] Y. Miao, W. Li, D. Tian, M. S. Hossain, and M. F. Alhamid, "Narrow Band Internet of Things: Simulation and Modelling," *IEEE Internet of Things Journal*, 2017.

[85] O. Sallent, J. Perez-Romero, R. Ferrus, and R. Agusti, "On radio access network slicing from a radio resource management perspective," *IEEE Wireless Communications*, vol. 24, no. 5, pp. 166–174, 2017.

[86] T. R. Henderson, M. Lacage, G. F. Riley, C. Dowell, and J. Kopena, "Network simulations with the ns-3 simulator," *SIGCOMM demonstration*, vol. 14, no. 14, p. 527, 2008.

[87] N. Baldo, M. Miozzo, M. Requena-Esteso, and J. Nin-Guerrero, "An open source product-oriented lte network simulator based on ns-3," in *Proceedings of the 14th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, ACM, 2011, pp. 293–298.

[88] Y. Kim, J. Bae, J. Lim, E. Park, J. Baek, S. I. Han, C. Chu, and Y. Han, "5g k-simulator: 5g system simulator for performance evaluation," in *2018 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, Oct. 2018, pp. 1–2. DOI: 10.1109/DySPAN.2018.8610404.

[89] G. Piro, L. A. Grieco, G. Boggia, F. Capozzi, and P. Camarda, "Simulating LTE cellular systems: An open-source framework," *IEEE transactions on vehicular technology*, vol. 60, no. 2, pp. 498–513, 2011.

[90] (2020). "5G-air-simulator Official Repository." [Online; accessed 31 Oct. 2021], [Online]. Available: https://github.com/telematics-lab/5G-air-simulator.

[91] S. Chen, S. Sun, Y. Wang, G. Xiao, and R. Tamrakar, "A comprehensive survey of TDD-based mobile communication systems from TD-SCDMA 3G to TD-LTE (A) 4G and 5G directions," *China Communications*, vol. 12, no. 2, pp. 40–60, 2015.

[92] F. Capozzi, G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, "Downlink packet scheduling in lte cellular networks: Key design issues and a survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 678–700, 2013.

[93] Arizona State University. (2019). "Video Trace Library." [Online; accessed 31 Oct. 2021], [Online]. Available: `http://trace.eas.asu.edu/`.

[94] R. Salami, C. Laflamme, B. Bessette, and J.-P. Adoul, "Description of itu-t recommendation g. 729 annex a: Reduced complexity 8 kbit/s cs-acelp codec," in *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, IEEE, vol. 2, 1997, pp. 775–778.

[95] J. Hoadley and P. Maveddat, "Enabling small cell deployment with hetnet," *IEEE Wireless Communications*, vol. 19, no. 2, pp. 4–5, 2012.

[96] A. Damnjanovic, J. Montojo, Y. Wei, T. Ji, T. Luo, M. Vajapeyam, T. Yoo, O. Song, and D. Malladi, "A survey on 3GPP heterogeneous networks," *IEEE Wireless communications*, vol. 18, no. 3, pp. 10–21, 2011.

[97] R. R. Roy, *Handbook of mobile ad hoc networks for mobility models*. Springer, 2011, vol. 170.

[98] 3GPP, "5G; NR; Physical layer procedures for data (Release 15)," 3rd Generation Partnership Project, Tech. Rep. 38214, Dec. 2018.

[99] E. Tuomaala and H. Wang, "Effective sinr approach of link to system mapping in ofdm/multi-carrier mobile network," 2005.

[100] C.-X. Wang, J. Bian, J. Sun, W. Zhang, and M. Zhang, "A Survey of 5G Channel Measurements and Models," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3142–3168, 2018.

[101] I. Kostić, "Analytical approach to performance analysis for channel subject to shadowing and fading," *IEE Proceedings-Communications*, vol. 152, no. 6, pp. 821–827, 2005.

[102] ITU-R, "Guidelines for Evaluation of Radio Interface Technologies for IMT-Advanced," Tech. Rep. M.2135, 2008.

[103] 3GPP, "Study on 3D channel model for LTE," 3rd Generation Partnership Project, Tech. Rep. 36873, Jun. 2015.

[104] ——, "5G; Study on channel model for frequencies from 0.5 to 100 GHz," 3rd Generation Partnership Project, Tech. Rep. 38901, Jan. 2018.

[105] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[106] 3GPP, "E-UTRA and E-UTRAN; LTE physical layer; Radio Frequency (RF) system scenarios (Release 14)," 3rd Generation Partnership Project, Tech. Rep. 36942, Mar. 2017.

[107] ——, "Simulation assumptions and parameters for FDD HeNB RF requirements," 3rd Generation Partnership Project, Tech. Rep. R4-092042, May 2009.

[108] Y. d. J. Bultitude and T. Rautiainen, "IST-4-027756 WINNER II Channel Models," Tech. Rep. D1.1.2 V1.2, Feb. 2008.

[109] ITU-R, "Minimum requirements related to technical performance for IMT-2020 radio interface(s)," ITU Radiocommunication Sector, Report M.2410, Nov. 2017.

[110] X. He, K. Niu, Z. He, and J. Lin, "Link layer abstraction in mimo-ofdm system," in *2007 International Workshop on Cross Layer Design*, Sep. 2007, pp. 41–44. DOI: 10.1109/IWCLD.2007.4379036.

[111] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "OpenAirInterface: A flexible platform for 5G research," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 33–38, 2014.

[112] I. Gomez-Miguelez, A. Garcia-Saavedra, P. D. Sutton, P. Serrano, C. Cano, and D. J. Leith, "srsLTE: an open-source platform for LTE evolution and experimentation," in *Proceedings of the Tenth ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation, and Characterization*, 2016, pp. 25–32.

[113] Ericsson, "Summary from email discussion on calibration step 1+2," 3rd Generation Partnership Project, Tech. Rep. R1-092019, May 2009.

[114] ——, "Email discussion summary on calibration step 1c," 3rd Generation Partnership Project, Tech. Rep. R1-092742, Jun. 2009.

[115] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures (Release-10)," 3rd Generation Partnership Project, Tech. Rep. 36213, Dec. 2010.

[116] J. C. Ikuno, M. Wrulich, and M. Rupp, "Performance and modeling of LTE H-ARQ," in *International ITG Workshop on Smart Antennas WSA*, 2009.

[117] Free Software Foundation, Inc. (2019). "The GNU Awk User's Guide." [Online; accessed 31 Oct. 2021], [Online]. Available: `https://www.gnu.org/software/gawk/manual/gawk.html`.

[118] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3590–3600, 2010.

[119] J. Nam, A. Adhikary, J.-Y. Ahn, and G. Caire, "Joint spatial division and multiplexing: Opportunistic beamforming, user grouping and simplified downlink scheduling," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 876–890, 2014.

[120] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive mimo for next generation wireless systems," *IEEE communications magazine*, vol. 52, no. 2, pp. 186–195, 2014.

[121] Z. Jiang, A. F. Molisch, G. Caire, and Z. Niu, "Achievable rates of fdd massive mimo systems with spatial channel correlation," *IEEE Transactions on Wireless Communications*, vol. 14, no. 5, pp. 2868–2882, 2015.

[122] X. Rao and V. K. Lau, "Distributed compressive csit estimation and feedback for fdd multi-user massive mimo systems," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3261–3271, 2014.

[123] M. S. Sim, J. Park, C.-B. Chae, and R. W. Heath, "Compressed channel feedback for correlated massive mimo systems," *Journal of Communications and Networks*, vol. 18, no. 1, pp. 95–104, 2016.

[124] J. Choi, D. J. Love, and P. Bidigare, "Downlink training techniques for fdd massive mimo systems: Open-loop and closed-loop training with memory," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 802–814, 2014.

[125] A. Adhikary, J. Nam, J.-Y. Ahn, and G. Caire, "Joint spatial division and multiplexing—the large-scale array regime," *IEEE transactions on information theory*, vol. 59, no. 10, pp. 6441–6463, 2013.

[126] FANTASTIC-5G D2.1, "Air interface framework and specification of system level simulations," Tech. Rep. 2.1, Apr. 2016.

[127] P. Marsch, Ö. Bulakci, O. Queseth, and M. Boldi, *5G System Design: Architectural and Functional Considerations and Long Term Research*. John Wiley & Sons, 2018.

[128] L. Rong, O. B. Haddada, and S.-E. Elayoubi, "Analytical analysis of the coverage of a MBSFN OFDMA network," in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, IEEE, 2008, pp. 1–5.

[129] B. Mouhouche and M. Al-Imari, "Optimization of delivery time in broadcast with acknowledgement and partial retransmission," in *Broadband Multimedia Systems and Broadcasting (BMSB), 2016 IEEE International Symposium on*, IEEE, 2016, pp. 1–5.

[130] 3GPP, "E-UTRA and E-UTRAN; Radio Resource Control (RRC); Protocol specification (Release 15)," 3rd Generation Partnership Project, Tech. Rep. 36331, Jul. 2018.

[131] D.-T. Phan-Huy, M. Sternad, and T. Svensson, "Making 5G adaptive antennas work for very fast moving vehicles," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 2, pp. 71–84, 2015.

[132] A. Grassi, G. Piro, G. Boggia, and D.-T. Phan-Huy, "A system level evaluation of SRTA-PI transmission scheme in the high-speed train use case," in *Proc. of IEEE International Conference on Telecommunications (ICT)*, Saint-Malo, France, Jun. 2018. eprint: `https://telematics.poliba.it/publications/2018/GrassiICT2018.pdf`.

[133] J. Kim, D. Munir, S. Hasan, and M. Chung, "Enhancement of LTE RACH through extended random access process," *Electronics Letters*, vol. 50, no. 19, pp. 1399–1400, 2014.

[134] A. Grassi, G. Piro, and G. Boggia, "A look at Random Access for Machine-Type Communications in 5G cellular networks," *Internet Technology Letters*, no. 1, Jan. 2018. eprint: `https://telematics.poliba.it/publications/2017/RandomAccess5G.pdf`.

[135] 3GPP, "5G; NR; Medium Access Control (MAC) protocol specification (Release 15)," 3rd Generation Partnership Project, Tech. Rep. 38321, Dec. 2018.

[136] Cisco, "The Zettabyte Era: Trends and Analysis," White Paper, Jun. 2017.

[137] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The Future of Industrial Communication: Automation Networks in the Era of the Internet of Things and Industry 4.0," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 17–27, 2017.

[138] J. Lloret, J. Tomas, A. Canovas, and L. Parra, "An Integrated IoT Architecture for Smart Metering," *IEEE Commun. Mag.*, vol. 54, no. 12, pp. 50–57, 2016.

[139] J. Guerrero-Ibáñez, S. Zeadally, and J. Contreras-Castillo, "Sensor Technologies for Intelligent Transportation Systems," *Sensors*, vol. 18, no. 4, p. 1212, 2018.

[140] P. P. Ray, "Internet of Things for Smart Agriculture: Technologies, Practices and Future Direction," *Journal of Ambient Intelligence and Smart Environments*, vol. 9, no. 4, pp. 395–420, 2017.

[141] G. Manogaran, R. Varatharajan, D. Lopez, P. M. Kumar, R. Sundarasekar, and C. Thota, "A New Architecture of Internet of Things and Big Data Ecosystem for Secured Smart Healthcare Monitoring and Alerting System," *Future Generation Computer Systems*, vol. 82, pp. 375–387, 2018.

[142] N. Vijayakumar and R. Ramya, "The Real Time Monitoring of Water Quality in IoT Environment," in *Proceedings of the International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, Coimbatore, India, 2015, pp. 1–5.

[143] X. Xiong, K. Zheng, R. Xu, W. Xiang, and P. Chatzimisios, "Low Power Wide Area Machine-to-Machine Networks: Key Techniques and Prototype," *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 64–71, 2015.

[144] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low Power Wide Area Networks: An Overview," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 855–873, 2017.

[145] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of LoRa: Long range & Low Power Networks for the Internet of Things," *Sensors*, vol. 16, no. 9, p. 1466, 2016.

[146] SIGFOX. (). "Sigfox - The Global Communications Service Provider for the Internet of Things," [Online]. Available: `https://www.sigfox.com` (visited on).

[147] 3GPP, "E-UTRA and E-UTRAN; LTE physical layer; General description (Release 13)," TS 36.201, Jun. 2016.

[148] ——, "E-UTRA and E-UTRAN; LTE physical layer; General description (Release 15)," TS 36.201, Jun. 2018.

[149] Y.-P. E. Wang, X. Lin, A. Adhikary, A. Grovlen, Y. Sui, Y. Blankenship, J. Bergman, and H. S. Razaghi, "A Primer on 3GPP Narrowband Internet of Things," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 117–123, 2017.

[150] J. Chen, K. Hu, Q. Wang, Y. Sun, Z. Shi, and S. He, "Narrowband internet of things: Implementations and applications," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 2309–2314, 2017.

[151] L. Feltrin, G. Tsoukaneri, M. Condoluci, C. Buratti, T. Mahmoodi, M. Dohler, and R. Verdone, "Narrowband IoT: A Survey on Downlink and Uplink Perspectives," *IEEE Wireless Communications*, vol. 26, no. 1, pp. 78–86, 2019.

[152] Y. D. Beyene, R. Jantti, K. Ruttik, and S. Iraji, "On the Performance of Narrow-Band Internet of Things (NB-IoT)," in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC)*, IEEE, 2017, pp. 1–6.

[153] M. Sauter, *From GSM to LTE-Advanced Pro and 5G: An Introduction to Mobile Networks and Mobile Broadband*. Wiley, 2017, ISBN: 9781119346937.

[154] 3GPP, "E-UTRA and E-UTRAN; Overall description (Release 15)," 3rd Generation Partnership Project, Tech. Rep. 36300, Jul. 2018.

[155] ——, "E-UTRA and E-UTRAN; Physical channels and modulation (Release 14)," 3rd Generation Partnership Project, Tech. Rep. 36211, Jun. 2017.

[156] ——, "Cellular System Support for Ultra-Low Complexity and Low Throughput Internet of Things (Release 13)," 3rd Generation Partnership Project, Tech. Rep. 45820, Dec. 2015.

[157] R. Ratasuk and N. Mangalvedhe and Y. Zhang and M. Robert and J. P. Koskinen, "Overview of narrowband IoT in LTE Rel-13," in *Proc. of 2016 IEEE Conference on Standards for Communications and Networking (CSCN)*, Oct. 2016. DOI: {10.1109/CSCN.2016.7785170}.

[158] X. Lin, A. Adhikary, and Y.-P. E. Wang, "Random access preamble design and detection for 3gpp narrowband iot systems.," *IEEE Wireless Commun. Letters*, vol. 5, no. 6, pp. 640–643, 2016.

[159] 3GPP, "E-UTRA and E-UTRAN; MAC protocol specification (Release 15)," 3rd Generation Partnership Project, Tech. Rep. 36321, Jul. 2018.

[160] O. Liberg, S. E. Löwenmark, S. Euler, B. Hofström, T. Khan, X. Lin, and J. Sedin, "Narrowband internet of things for non-terrestrial networks," *arXiv preprint arXiv:2010.04906*, 2020.

[161] G. Charbit, D. Lin, K. Medles, L. Li, and I. Fu, "Space-Terrestrial Radio Network Integration for IoT," in *Proc. of IEEE 6G Wireless Summit (6G SUMMIT)*, 2020, pp. 1–5. DOI: 10.1109/6GSUMMIT49458.2020.9083854.

[162] M. Conti, S. Andrenacci, N. Maturo, S. Chatzinotas, and A. Vanelli-Coralli, "Doppler Impact Analysis for NB-IoT and Satellite Systems Integration," in *Proc. of IEEE International Conference on Communications (ICC)*, 2020, pp. 1–7. DOI: 10.1109/ICC40277.2020.9149140.

[163] O. Kodheli, S. Andrenacci, N. Maturo, S. Chatzinotas, and F. Zimmer, "Resource Allocation Approach for Differential Doppler Reduction in NB-IoT over LEO Satellite," in *Proc. of IEEE Advanced Satellite Multimedia Systems Conference and the 15th Signal Processing for Space Communications Workshop (ASMS/SPSC)*, 2018, pp. 1–8. DOI: 10.1109/ASMS-SPSC.2018.8510724.

[164] ——, "An Uplink UE Group-Based Scheduling Technique for 5G mMTC Systems Over LEO Satellite," *IEEE Access*, vol. 7, pp. 67 413–67 427, 2019. DOI: 10.1109/ACCESS.2019.2918581.

[165] O. Kodheli, N. Maturo, S. Chatzinotas, S. Andrenacci, and F. Zimmer, "On the Random Access Procedure of NB-IoT Non-Terrestrial Networks," in *Proc. of IEEE Advanced Satellite Multimedia Systems Conference (ASMS) and 16th Signal Processing for Space Communications Workshop (SPSC)*, IEEE Virtual Conference, 2020.

[166] S. Cluzel, L. Franck, J. Radzik, S. Cazalens, M. Dervin, C. Baudoin, and D. Dragomirescu, "3GPP NB-IOT Coverage Extension Using LEO Satellites," in *Proc. of IEEE Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1–5. DOI: 10.1109/VTCSpring.2018.8417723.

[167] G. Aiyetoro and P. Owolawi, "Spectrum Management Schemes for Internet of Remote Things (IoRT) Devices in 5G Networks via GEO Satellite," *Future Internet*, vol. 11, no. 12, p. 257, 2019.

[168] G. Aiyetoro and P. Owolawi, "Dynamic Packet Scheduling for Internet of Remote Things (IoRT) devices in 5G Satellite Networks," in *Proc. of IEEE International Conference on Advances in Computing and*

Communication Engineering (ICACCE), 2020, pp. 1–6. DOI: 10.1109/ICACCE49060.2020.9154993.

[169] 3GPP, "Solutions for NR to support Non-Terrestrial Networks (NTN) (Release 1)," 3rd Generation Partnership Project, Tech. Rep. 38.821, 2019.

[170] Z. Qu, G. Zhang, H. Cao, and J. Xie, "LEO Satellite Constellation for Internet of Things," *IEEE Access*, vol. 5, pp. 18 391–18 401, 2017. DOI: 10.1109/ACCESS.2017.2735988.

[171] I. F. Akyildiz and A. Kak, "The Internet of Space Things/CubeSats," *IEEE Network*, vol. 33, no. 5, pp. 212–218, 2019. DOI: 10.1109/MNET.2019.1800445.

[172] C. A. Balanis, *Antenna Theory: Analysis and Design, 2nd Edition*. Wiley, 1996.

[173] ITU, "Attenuation by atmospheric gases and related effects," International Telecommunication Union (ITU), Recommendation, 2019, ITU-R P.676-12.

[174] ——, "Attenuation due to clouds and fog," International Telecommunication Union (ITU), Recommendation, 2019, ITU-R P.840-8.

[175] L. J.Ippolito, *Satellite Communications Systems Engineering: Atmospheric Effects, Satellite Link Design and System Performance*, 2nd. Wiley Publishing, 2017, ISBN: 1119259371.

[176] ITU, "Propagation data and prediction methods required for the design of Earth-space telecommunication systems," International Telecommunication Union (ITU), Recommendation, 2017, ITU-R P.618-13.

[177] A. Mahmood and S. Zafar, "Performance Analysis of Narrowband Internet of Things (NB-IoT) Deployment Modes," in *Proc. of IEEE International Multitopic Conference (INMIC)*, 2019, pp. 1–8. DOI: 10.1109/INMIC48123.2019.9022748.

[178] 3GPP, "Study on RAN Improvements for Machine-Type Communications," 3rd Generation Partnership Project (3GPP), Technical Report TR 37.868, 2011.

[179] A. Ksentini and N. Nikaein, "Toward enforcing network slicing on ran: Flexibility and resources abstraction," *IEEE Communications Magazine*, vol. 55, no. 6, pp. 102–108, 2017.

[180] D. Marabissi and R. Fantacci, "Highly flexible RAN slicing approach to manage isolation, priority, efficiency," *IEEE Access*, vol. 7, pp. 97 130–97 142, 2019.

[181] K. Samdanis, X. Costa-Perez, and V. Sciancalepore, "From Network Sharing to Multi-Tenancy: The 5G Network Slice Broker," *IEEE Communications Magazine*, vol. 54, no. 7, pp. 32–39, 2016.

[182] S. D'Oro, F. Restuccia, A. Talamonti, and T. Melodia, "The Slice Is Served: Enforcing Radio Access Network Slicing in Virtualized 5G Systems," in *Proc. IEEE Conference on Computer Communications (INFOCOM)*, Apr. 2019, pp. 442–450.

[183] S. D'Oro, F. Restuccia, T. Melodia, and S. Palazzo, "Low-Complexity Distributed Radio Access Network Slicing: Algorithms and Experimental Results," *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, pp. 2815–2828, Dec. 2018.

[184] P. Caballero, A. Banchs, G. De Veciana, and X. Costa-Pérez, "Network Slicing Games: Enabling Customization in Multi-Tenant Mobile Networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 2, pp. 662–675, Apr. 2019.

[185] S. Costanzo, I. Fajjari, N. Aitsaadi, and R. Langar, "Dynamic Network Slicing for 5G IoT and eMBB services: A New Design with Prototype and Implementation Results," in *Proc. IEEE Cloudification of the Internet of Things (CIoT)*, Jul. 2018.

[186] Y. Sun, M. Peng, S. Mao, and S. Yan, "Hierarchical Radio Resource Allocation for Network Slicing in Fog Radio Access Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3866–3881, Apr. 2019.

[187] K. Zhu and E. Hossain, "Virtualization of 5G Cellular Networks as a Hierarchical Combinatorial Auction," *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2640–2654, Oct. 2016.

[188] L. Zanzi and V. Sciancalepore, "On Guaranteeing End-to-End Network Slice Latency Constraints in 5G Networks," in *Proc. IEEE International Symposium on Wireless Communication Systems (ISWCS)*, Aug. 2018.

[189] A. Ksentini, P. A. Frangoudis, A. PC, and N. Nikaein, "Providing Low Latency Guarantees for Slicing-Ready 5G Systems via Two-Level MAC Scheduling," *IEEE Network*, vol. 32, no. 6, pp. 116–123, Nov. 2018.

[190] T. Guo and A. Suárez, "Enabling 5G RAN Slicing With EDF Slice Scheduling," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2865–2877, Mar. 2019.

[191] D. Tang, C. Hu, and T. Dang, "Delay-Aware Resource Allocation for Network Slicing in Fog Radio Access Networks," in *Proc. IEEE International Conference on Wireless Communications and Signal Processing (WCSP)*, Oct. 2018.

[192] S. D'Oro, L. Bonati, F. Restuccia, M. Polese, M. Zorzi, and T. Melodia, "Sl-edge: Network slicing at the edge," in *Proceedings of the Twenty-First International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*, 2020, pp. 1–10.

[193] H. D. R. Albonda and J. Pérez-Romero, "An efficient ran slicing strategy for a heterogeneous network with embb and v2x services," *IEEE access*, vol. 7, pp. 44 771–44 782, 2019.

[194] H. Khan, P. Luoto, S. Samarakoon, M. Bennis, and M. Latva-Aho, "Network Slicing for Vehicular Communication," *Transactions on Emerging Telecommunications Technologies*, e3652, May 2019.

[195] N. Mouawad, R. Naja, and S. Tohme, "Inter-slice handover management in a v2x slicing environment using bargaining games," *Wireless Networks*, vol. 26, no. 5, pp. 3883–3903, 2020.

[196] 3GPP, " Technical Specification Group Services and System Aspects; Study on architecture enhancements for the Evolved Packet System (EPS) and the 5G System (5GS) to support advanced V2X services (Release 16)," 3rd Generation Partnership Project, Tech. Rep. 23.786, Jun. 2019.

[197] Y. Chen, W. Liu, Z. Niu, Z. Feng, Q. Hu, and T. Jiang, "Pervasive intelligent endogenous 6g wireless systems: Prospects, theories and key technologies," *Digital Communications and Networks*, vol. 6, no. 3, pp. 312–320, 2020.

[198] F. Jiang, K. Wang, L. Dong, C. Pan, and K. Yang, "Stacked Autoencoder-Based Deep Reinforcement Learning for Online Resource Scheduling in Large-Scale MEC Networks," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9278–9290, 2020.

[199] F. B. Mismar, B. L. Evans, and A. Alkhateeb, "Deep reinforcement learning for 5g networks: Joint beamforming, power control, and interference coordination," *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1581–1592, 2019.

[200] C. Gutterman, E. Grinshpun, S. Sharma, and G. Zussman, "Ran resource usage prediction for a 5g slice broker," in *Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ACM, 2019, pp. 231–240.

[201] S. Bakri, P. A. Frangoudis, A. Ksentini, and M. Bouaziz, "Data-Driven RAN Slicing Mechanisms for 5G and Beyond," *IEEE Transactions on Network and Service Management*, pp. 1–1, 2021. DOI: 10.1109/TNSM.2021.3098193.

[202] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Perez, "Deepcog: Optimizing resource provisioning in network slicing with ai-based capacity forecasting," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 2, pp. 361–376, 2019.

[203] M. R. Raza, C. Natalino, P. Öhlen, L. Wosinska, and P. Monti, "Reinforcement learning for slicing in a 5G flexible RAN," *Journal of Lightwave Technology*, vol. 37, no. 20, pp. 5161–5169, 2019.

[204] R. Li, C. Wang, Z. Zhao, R. Guo, and H. Zhang, "The LSTM-Based Advantage Actor-Critic Learning for Resource Management in Network Slicing With User Mobility," *IEEE Communications Letters*, vol. 24, no. 9, pp. 2005–2009, 2020.

[205] B. Khodapanah, A. Awada, I. Viering, A. N. Barreto, M. Simsek, and G. Fettweis, "Slice management in radio access network via deep reinforcement learning," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, IEEE, 2020, pp. 1–6.

[206] X. Chen, Z. Zhao, C. Wu, M. Bennis, H. Liu, Y. Ji, and H. Zhang, "Multi-tenant cross-slice resource orchestration: A deep reinforcement learning approach," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2377–2392, 2019.

[207] Y. Hua, R. Li, Z. Zhao, X. Chen, and H. Zhang, "GAN-Powered Deep Distributional Reinforcement Learning for Resource Management in Network Slicing," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 2, pp. 334–349, 2019.

[208] Y. Abiko, T. Saito, D. Ikeda, K. Ohta, T. Mizuno, and H. Mineno, "Flexible resource block allocation to multiple slices for radio access network slicing using deep reinforcement learning," *IEEE Access*, vol. 8, pp. 68 183–68 198, 2020.

[209] Q. Ye, W. Shi, K. Qu, H. He, W. Zhuang, and X. Shen, "Joint ran slicing and computation offloading for autonomous vehicular networks: A learning-assisted hierarchical approach," *IEEE Open Journal of Vehicular Technology*, 2021.

[210] Z. Wang, Y. Wei, F. R. Yu, and Z. Han, "Utility Optimization for Resource Allocation in Edge Network Slicing Using DRL," in *GLOBECOM 2020-2020 IEEE Global Communications Conference*, IEEE, 2020, pp. 1–6.

[211] W. Wu, N. Chen, C. Zhou, M. Li, X. Shen, W. Zhuang, and X. Li, "Dynamic RAN Slicing for Service-Oriented Vehicular Networks via Constrained Learning," *IEEE Journal on Selected Areas in Communications*, 2020.

[212] J. Mei, X. Wang, K. Zheng, G. Boudreau, A. B. Sediq, and H. Abouzeid, "Intelligent radio access network slicing for service provisioning in 6g: A hierarchical deep reinforcement learning approach," *IEEE Transactions on Communications*, 2021.

[213] W. Guan, H. Zhang, and V. C. Leung, "Slice Reconfiguration Based on Demand Prediction with Dueling Deep Reinforcement Learning," in *GLOBECOM 2020-2020 IEEE Global Communications Conference*, IEEE, 2020, pp. 1–6.

[214] R. Li, Z. Zhao, Q. Sun, I. Chih-Lin, C. Yang, X. Chen, M. Zhao, and H. Zhang, "Deep reinforcement learning for resource management in network slicing," *IEEE Access*, vol. 6, pp. 74 429–74 441, 2018.

[215] F. Mason, G. Nencioni, and A. Zanella, "Using distributed reinforcement learning for resource orchestration in a network slicing scenario," *arXiv preprint arXiv:2105.07946*, 2021.

[216] A. Anand, G. De Veciana, and S. Shakkottai, "Joint Scheduling of URLLC and eMBB Traffic in 5G Wireless Networks," in *Proc. IEEE Conference on Computer Communications (INFOCOM)*, Apr. 2018, pp. 1970–1978.

[217] M. Alsenwi, N. H. Tran, M. Bennis, A. Kumar Bairagi, and C. S. Hong, "eMBB-URLLC Resource Slicing: A Risk-Sensitive Approach," *IEEE Communications Letters*, vol. 23, no. 4, pp. 740–743, Apr. 2019.

[218] J. Park and M. Bennis, "URLLC-eMBB Slicing to Support VR Multimodal Perceptions over Wireless Cellular Systems," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Dec. 2018.

[219] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G Wireless Network Slicing for eMBB, URLLC, and mMTC: A Communication-Theoretic View," *IEEE Access*, vol. 6, pp. 55 765–55 779, 2018.

[220] ETSI, "MEC in 5G Networks," White Paper No. 28, Jun. 2018.

[221] M. Bennis, M. Debbah, and H. V. Poor, "Ultrareliable and Low-Latency Wireless Communication: Tail, Risk, and Scale," *Proceedings of the IEEE*, vol. 106, no. 10, pp. 1834–1853, Oct. 2018.

[222] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1738–1762, 2019.

[223] A. Nikravesh, S. Ajila, C. Lung, and W. Ding, "An experimental investigation of mobile network traffic prediction accuracy," *Services Transactions on Big Data*, vol. 3, no. 1, pp. 1–16, 2016.

[224] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep reinforcement learning based resource allocation for v2v communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3163–3173, 2019.

[225] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press, 2018.

[226] 5G PPP Architecture Working Group, "View on 5G Architecture," *White Paper*, Feb. 2019, V3.0.

[227] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[228] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, "Deep reinforcement learning that matters," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[229] D. Zha, K.-H. Lai, K. Zhou, and X. Hu, "Experience replay optimization," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, International Joint Conferences on Artificial Intelligence Organization, Jul. 2019, pp. 4243–4249. DOI: 10.24963/ijcai.2019/589.

[230] 5GAA, *C-v2x use cases: Methodology, examples and service level requirements*, White Paper, 2019.

[231] M. E. Morocho-Cayamcela, H. Lee, and W. Lim, "Machine Learning for 5G/B5G Mobile and Wireless Communications: Potential, Limitations, and future Directions," *IEEE Access*, vol. 7, pp. 137 184–137 206, 2019. DOI: 10.1109/ACCESS.2019.2942390.

[232] 3GPP, "Service requirements for the 5G system - stage 1," *Tech. Spec. 3GPP TS 22.261, V16.5.0*, Sep. 2018.

[233] 5GAA, *C-v2x use cases volume ii: Examples and service level requirements*, White Paper, 2020.

[234] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, "Machine Learning in the Air," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2184–2199, 2019.

[235] 3GPP, "5G; NR; Physical layer; General description (Release 15)," 3rd Generation Partnership Project, Technical Specification (TS) 38.201, Dec. 2017, V15.0.0.

[236] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[237] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, "Mobile Encrypted Traffic Classification Using Deep Learning: Experimental Evaluation, Lessons Learned, and Challenges," *IEEE Transactions on Network and Service Management*, vol. 16, no. 2, pp. 445–458, Jun. 2019. DOI: 10.1109/TNSM.2019.2899085.

[238] Z. Dai and R. Heckel, "Channel normalization in convolutional neural network avoids vanishing gradients," *arXiv preprint arXiv:1907.09539*, 2019.

[239] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[240] S. Hashem and B. Schmeiser, "Improving model accuracy using optimal linear combinations of trained neural networks," *IEEE Transactions on neural networks*, vol. 6, no. 3, pp. 792–794, 1995. DOI: 10.1109/72.377990.

[241] 3GPP, "Evaluation assumptions for Phase 1 NR MIMO system level calibration," 3rd Generation Partnership Project, TSG RAN WG1 Meeting R1-1701824, Feb. 2017.

[242] J. G. Carney and P. Cunningham, "The epoch interpretation of learning," *IEEE Transaction on Neural Networks*, vol. 8, pp. 111–116, 1998.

[243] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. of 3rd International Conference on Learning Representations (ICLR)*, 2015, pp. 1–15.

[244] M. Shi, K. Yang, Z. Han, and D. Niyato, "Coverage analysis of integrated sub-6GHz-mmWave cellular networks with hotspots," *IEEE Transactions on Communications*, vol. 67, no. 11, pp. 8151–8164, 2019.

[245] 3GPP, "Technical Specification Group Services and System Aspects; System architecture for the 5G System (5GS); Stage 2 (Release 16)," *Tech. Rep.*, no. TR 23.501, Dec. 2019.

[246] ——, " Technical Specification Group Services and System Aspects; Mobile Communication System for Railways; Stage 1 (Release 17)," 3rd Generation Partnership Project, Tech. Rep. 22.289, Dec. 2019.

[247] ——, " Technical Specification Group Services and System Aspects; Study on enhancement of 3GPP Support for 5G V2X Services (Release 16)," 3rd Generation Partnership Project, Tech. Rep. 22.886, Dec. 2018.

[248] The International Association of Public Transport. (2021). "World report on metro automation." [Online; accessed 31 Oct. 2021], [Online]. Available: https://www.uitp.org/publications/world-report-on-metro-automation/.

[249] SYSTRA Group. (2021). "AUTOMATED AND AUTONOMOUS PUBLIC TRANSPORT: POSSIBILITIES, CHALLENGES AND TECHNOLOGIES." [Online; accessed 31 Oct. 2021], [Online]. Available: https://www.systra.com/en/automated-and-autonomous-public-transport-possibilities-challenges-and-technologies/.

[250] D. Trentesaux, R. Dahyot, A. Ouedraogo, D. Arenas, S. Lefebvre, W. Schön, B. Lussier, and H. Cheritel, "The autonomous train," in *2018 13th Annual Conference on System of Systems Engineering (SoSE)*, IEEE, 2018, pp. 514–520.

[251] Mandò, Gianluca. (2019). "ELASTIC is boosting the autonomous tram in Florence." [Online; accessed 31 Oct. 2021], [Online]. Available: `https://elastic-project.eu/media/news-and-press-releases/elastic-boosting-autonomous-tram-florence`.

[252] M.-P. Pacaux-Lemoine, Q. Gadmer, and P. Richard, "Train remote driving: A human-machine cooperation point of view," in *2020 IEEE International Conference on Human-Machine Systems (ICHMS)*, IEEE, 2020, pp. 1–4.

[253] N. Alliance, "V2x," *White Paper*, vol. 1, 2018.

[254] 3GPP, " Technical Specification Group Services and System Aspects; Enhancement of 3GPP support for V2X scenarios; Stage 1 (Release 16)," 3rd Generation Partnership Project, Tech. Rep. 22.186, Jun. 2019.

[255] H.-S. Park, Y. Lee, T.-J. Kim, B.-C. Kim, and J.-Y. Lee, "Handover mechanism in nr for ultra-reliable low-latency communications," *IEEE Network*, vol. 32, no. 2, pp. 41–47, 2018.

[256] 3GPP, "NR; User Equipment (UE) radio transmission and reception; Part 1: Range 1 Standalone (Release 16)," 3rd Generation Partnership Project, Tech. Rep. 38.101, Nov. 2020.

[257] M. B. Zeytinci, V. Sari, F. K. Harmanci^, E. Anarim, and M. Akar, "Location estimation using rss measurements with unknown path loss exponents," *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, no. 1, pp. 1–14, 2013.

[258] M. van der Lende, F. M. E. Cox, G. H. Visser, J. W. Sander, and R. D. Thijs, "Value of video monitoring for nocturnal seizure detection in a residential setting.," *Epilepsia*, vol. 57 11, pp. 1748–1753, 2016.

[259] M. Richart, J. Baliosian, J. Serrat, J.-L. Gorricho, and R. Agüero, "Slicing in WiFi networks through airtime-based resource allocation," *Journal of Network and Systems Management*, vol. 27, no. 3, pp. 784–814, Jul. 2019, ISSN: 1573-7705. DOI: `10.1007/s10922-018-9484-x`. [Online]. Available: `https://doi.org/10.1007/s10922-018-9484-x`.

[260] M. Carmo, F. S. Dantas Silva, A. V. Neto, D. Corujo, and R. Aguiar, "Network-cloud slicing definitions for Wi-Fi sharing systems to enhance 5G ultra dense network capabilities," *Wireless Communications and Mobile Computing*, vol. 2019, 2019.

[261] M.-P. Hosseini, T. X. Tran, D. Pompili, K. Elisevich, and H. Soltanian-Zadeh, "Multimodal data analysis of epileptic eeg and rs-fmri via deep learning and edge computing," *Artificial Intelligence in Medicine*, vol. 104, p. 101 813, 2020.

[262] P. M. Vergara, E. de la Cal, J. R. Villar, V. M. González, and J. Sedano, "An iot platform for epilepsy monitoring and supervising," *Journal of Sensors*, vol. 2017, 2017.

[263] S. Sareen, S. K. Sood, and S. K. Gupta, "An automatic prediction of epileptic seizures using cloud computing and wireless sensor networks," *Journal of medical systems*, vol. 40, no. 11, p. 226, 2016.

[264] M. Asif-Ur-Rahman, F. Afsana, M. Mahmud, M. S. Kaiser, M. R. Ahmed, O. Kaiwartya, and A. James-Taylor, "Toward a heterogeneous mist, fog, and cloud-based framework for the internet of healthcare things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4049–4062, 2018.

[265] H. Wang, J. Gong, Y. Zhuang, H. Shen, and J. Lach, "Healthedge: Task scheduling for edge computing with health emergency and human behavior consideration in smart homes," in *2017 IEEE International Conference on Big Data (Big Data)*, IEEE, 2017, pp. 1213–1222.

[266] G. Cisotto, E. Casarin, and S. Tomasin, "Requirements and enablers of advanced healthcare services over future cellular systems," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 76–81, 2020.

[267] "IEEE draft standard for information technology – telecommunications and information exchange between systems local and metropolitan area networks – specific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment enhancements for high efficiency WLAN," *IEEE P802.11ax/D6.0, November 2019*, pp. 1–780, Dec. 2019, ISSN: null.

[268] Ericsson AB, *Fixed Wireless Access handbook*. 2019, p. 192. DOI: https://doi.org/http://handle.itu.int/11.1002/pub/800c949d-en". [Online]. Available: https://www.itu-ilibrary.org/content/publication/pub-800c949d-en.

[269] High Efficiency (HE) Wireless LAN Task Group, "TGax simulation scenarios," Tech. Rep. 11-14-0980, Nov. 2015.

[270] L. Zanzi and V. Sciancalepore, "On Guaranteeing End-to-End Network Slice Latency Constraints in 5G Networks," in *Proc. IEEE International Symposium on Wireless Communication Systems (ISWCS)*, Aug. 2018.

[271] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, 2016.

[272] B. Li, Z. Fei, and Y. Zhang, "Uav communications for 5g and beyond: Recent advances and future trends," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2241–2263, 2018.

[273] A. A. Khuwaja, Y. Chen, N. Zhao, M.-S. Alouini, and P. Dobbins, "A survey of channel modeling for uav communications," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2804–2821, 2018.

[274] 3GPP, "5G; NR; Study on NR Vehicle-to-Everything (V2X) (Release 16)," 3rd Generation Partnership Project, Tech. Rep. 38885, Mar. 2019.