Centralised orchestration and hybrid resource management for Ultra Reliable and Low Latency Communications (URLLC)

Ngoc Lam Dinh

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : MSTII - Mathématiques, Sciences et Technologies de l'Information, Informatique

Spécialité : Mathématiques et Informatique

Unité de recherche : CEA Grenoble - LETI

Orchestration centralisée et gestion des ressources hybrides pour les communications ultra-fiable et à faible latence (URLLC)

Centralised orchestration and hybrid resource management for Ultra Reliable and Low Latency Communications (URLLC)

Présentée par

Ngoc-Lam DINH

Encadrement de thèse :

Emilio CALVANESE STRINATI — Directeur de thèse
Directeur de recherche, CEA Centre de Grenoble

Mickaël MAMAN — Co-encadrant de thèse
Ingénieur de recherche, CEA Centre de Grenoble

Rapporteurs :

Jean-Marie GORCE
Professeur des Universités, INSA Lyon

Giovanni STEA
Professeur, Università degli Studi di Pisa (Italie)

Thèse soutennue publiquement le **3 Mars 2023**, devant le jury composé de :

Andrzej DUDA — Président
Professeur des Universités, Grenoble INP

Jean-Marie GORCE — Rapporteur
Professeur des Universités, INSA Lyon

Giovanni STEA — Rapporteur
Professeur, Università degli Studi di Pisa (Italie)

Marco MIOZZO — Examinateur
Docteur en sciences, CTTC (Catalunya, Espagne)

Emilio CALVANESE STRINATI — Directeur de thèse
Directeur de recherche, CEA Centre de Grenoble

Invités :

Mickaël MAMAN — Co-encadrant de thèse
Ingénieur de recherche, CEA Centre de Grenoble

Acknowledgements

Cette première page est l'occasion pour moi de remercier les personnes sans lesquelles ce travail n'aurait pu être mené à bout. La thèse est un long exercice semé d'embûches et de doutes, mais c'est aussi une expérience enrichissante en connaissances et en rencontres.

En premier lieu, je tiens à remercier M. Mickaël MAMAN pour ses conseils tout au long de ces trois années. Merci pour tes conseils, ta patience et ton expertise. Tu as toujours su me guider et trouver les astuces pour booster ma motivation. Enfin, merci à mon directeur de thèse, M. Emilio CALVANESE STRINATI, qui a su parfaitement diriger cette thèse. Tu as toujours trouvé le temps d'améliorer mon travail et de proposer des discussions pertinentes. Merci également pour votre relecture assidue de l'ensemble de mes travaux.

Je tiens également à remercier les membres de mon jury de thèse, à commencer par professeur Andrzej DUDA de Grenoble INP (France), pour avoir présidé ce jury. Par ailleurs, je tiens à remercier professeur Jean-Marie GORCE, de l'INSA Lyon (France), pour m'avoir accompagné tout au long de ma thèse en tant qu'un des membres du jury du comité de suivi et pour avoir accepté d'être un rapporteur de cette thèse. Aussi, Je voudrais remercier professeur Giovanni STEA de l'Università degli Studi di Pisa (Italie) d'avoir accepté d'examiner ma thèse en tant qu'un rapporteur. Enfin, je voudrais remercier Dr. Marco MIOZZO, du Centre Tecnològic de Telecomunicacons de Catalunya (CTTC, Espagne), pour avoir accepté d'examiner cette thèse. Je remercie sincèrement tous les membres du jury pour leurs commentaires sur le manuscrit ainsi que sur ma soutenance.

De plus, je souhaiterais remercier toutes les équipes LSHD et LCOI du CEA Leti, qui m'ont accueilli tout au long de mes travaux. Merci à tout le personnel permanent ainsi qu'au personnel non permanent, aux doctorants et aux stagiaires qui sont venus au laboratoire et ont partagé avec moi leurs échanges enrichissants.

Un autre remerciement est dedié à ma famille, qui m'a toujours soutenue : merci à mon papa pour ton soutien inconditionnel et pour m'avoir donné le désir de poursuivre mes études. Tu es parti bien trop vite et sans toi rien n'aurait été possible. Merci également à ma mère et à mon petit frère, dont je suis très fière.

Enfin et surtout, merci à ma compagne, Huong, pour tout. Merci pour ta patience pendant ma thèse et pour ton énorme confiance. Passer du temps avec toi est toujours merveilleux et tu es une épouse extraordinaire. Je suis plus qu'heureux de prévoir de passer le reste de ma vie avec toi !

Abstract

During the past decades, with the significant proliferation of Fifth Generation (5G) networks, Ultra Reliable and Low Latency Communications (URLLC) has been becoming indispensable to support time-critical applications where data transmission in an error-prone wireless medium must be bounded by low-delay and ultra-reliability requirements. However, the phenomenal growth of data traffic coupled with the heterogeneous Quality of Service (QoS), are posing unprecedented challenges. It suggests the paradigm shift from reactive and centralised networks towards proactive, low latency, ultra-reliable and decentralised network designs which allow more flexible, scalable and low complexity deployments of resource management solutions. In this thesis, we focus on the refinement of fundamental trade-offs between latency, reliability and other Key Performance Indicators (KPIs) such as radio resource efficiency and network throughput in guaranteeing URLLC communications In the first part of our work, we focus on centralised resource management to ensure latency and reliability of URLLC communication in dynamic scenarios. First, we prove that proactive resource allocation is essential to cope with the uncertainty of the wireless medium and to improve latency at the cost of potentially low allocation efficiency. When the traffic source rate is further considered in addition to the rapidly changing wireless channel, we propose an adaptive allocation algorithm based on Lyapunov's optimization for two-queue state (data queue and transmission queue) management and to improve allocation efficiency while fulfilling the requirements of URLLC communications. Furthermore, under highly critical perturbation system, we extend this adaptive allocation framework towards adaptive and reliability-aware one to improve the reliability of wireless communication without sacrificing low latency communication and high allocation efficiency. The performance gains of our algorithms are then evaluated with the system-level simulator NS3 and validated with experiments based on OpenAirInterface (OAI) hardware where full-stack radio protocols are considered. In the second part of our work, we propose a novel hybrid radio resource management using Grant-Based (GB) and Grant-Free (GF) access schemes, where limited radio resources are exploited for multi-user communication in the UL direction. Leveraging Multi Agent Reinforcement Learning (MARL), we provide two different algorithms to optimize a global network objective in terms of latency, reliability and network throughput: Multi-agent Deep-Q Learning (MADQL) and Multi-agent Deterministic Policy Gradient (MADDPG). MADQL is executed in a fully distributed manner so that each associated user (agent) learns its optimal action-value function through its communication via GB or GF access with the gNb, relying only on its local observation. On the other hand, MADDPG is introduced as a semi-distributed system with the associated gNb function as a global observer (critic), criticising the actions of each associated agent (actor) in the network. By leveraging centralised training and decentralised execution, we achieve a shared goal better than the MADQL algorithm. Then, through a system level simulation, we show the gain of our approach to efficiently manage radio resources.

Keywords: Ultra Reliable and Low Latency Communications (URLLC), Lyapunov's Optimization , centralized/hybrid resource management, Multi Agent Reinforcement Learning (MARL), semi-distributed system.

Résumé

DURANT les dernières décennies, avec la prolifération significative des réseaux 5G, les communications ultra fiables et à faible latence (URLLC) sont devenues indispensables pour prendre en charge les applications critiques où la transmission de données sur un support sans fil sujet aux erreurs doit être limitée par des exigences de faible retard et d'ultra fiabilité. Cependant, la croissance phénoménale du trafic de données, associée à l'hétérogénéité de la qualité de service (QoS), pose des défis sans précédent. Elle suggère le changement de paradigme des réseaux réactifs et centralisés vers des conceptions de réseaux proactifs, à faible latence, ultra-fiables et décentralisés qui permettent des déploiements plus flexibles, évolutifs et peu complexes de solutions de gestion des ressources. Dans cette thèse, nous nous concentrons sur le raffinement des compromis fondamentaux entre la latence, la fiabilité et d'autres indicateurs clés de performance tels que l'efficacité des ressources radio et le débit du réseau pour garantir les communications URLLC. Dans la première partie de notre travail, nous nous concentrons sur la gestion centralisée des ressources afin de garantir la latence et la fiabilité des communications URLLC dans des scénarios dynamiques. Tout d'abord, nous prouvons que l'allocation proactive des ressources est essentielle pour faire face à l'incertitude du support sans fil et pour améliorer le temps de latence au prix d'une efficacité d'allocation potentiellement faible. Lorsque le taux de la source de trafic est pris en compte en plus du canal sans fil qui change rapidement, nous proposons un algorithme d'allocation adaptatif basé sur l'optimisation de Lyapunov pour la gestion de deux files d'attente (file d'attente de données et file d'attente de transmission) et pour améliorer l'efficacité de l'allocation tout en satisfaisant aux exigences des communications URLLC. En outre, dans le cadre d'un système de perturbation hautement critique, nous étendons ce cadre d'allocation adaptative à un cadre adaptatif et sensible à la fiabilité afin d'améliorer la fiabilité de la communication sans fil sans sacrifier la communication à faible latence et l'efficacité de l'allocation. Les gains de performance de nos algorithmes sont ensuite évalués avec le simulateur au niveau du système NS3 et validés par des expériences basées sur le matériel OpenAirInterface (OAI) où des protocoles radio complets sont pris en compte. Dans la deuxième partie de notre travail, nous proposons une nouvelle gestion hybride des ressources radio utilisant les schémas d'accès Grant-Based (GB) et Grant-Free (GF), où les ressources radio limitées sont exploitées pour la communication multi-utilisateurs dans la direction UL. En nous appuyant sur Multi Agent Reinforcement Learning (MARL), nous fournissons deux algorithmes différents pour optimiser un objectif de réseau global en termes de latence, de fiabilité et de débit du réseau : Multi-agent Deep-Q Learning (MADQL) et Multi-agent Deterministic Policy Gradient (MADDPG). MADQL est exécuté de manière entièrement distribuée de sorte que chaque utilisateur associé (agent) apprend sa fonction action-valeur optimale par le biais de sa communication via l'accès GB ou GF avec le gNb, en s'appuyant uniquement sur son observation locale. D'autre part, Multi-agent Deterministic Policy Gradient (MADDPG) est présenté comme un système semi-distribué avec la fonction gNb associée en tant qu'observateur (critique) global, critiquant les actions de chaque agent (acteur) associé dans le réseau. En tirant parti de la formation centralisée et de l'exécution décentralisée, nous atteignons un objectif commun mieux que l'algorithme MADQL. Ensuite, par le cadre d'une simulation au niveau du système, nous montrons les avantages de notre approche pour gérer efficacement les ressources radio.

Mots clés : URLLC, optimisation de Lyapunov , gestion des ressources, MARL, système semi-distribué.

Publications

Patents

[P1] M. Maman, **L. N. Dinh**, E. Calvanese Strinati, *"Method to exploit latency distribution for early decision making,"* filed patent, FR2103542.

[P2] **L. N. Dinh**, M. Maman, E. Calvanese Strinati, *"Methods and apparatus for jitter-aware scheduling in wireless Time Sensitive Network communications,"* filed patent, .

International Journal Communications

[J1] M. Maman, E. Calvanese Strinati, **L. N. Dinh**, et al., *"Beyond Private 5G Networks: Applications, Architectures, Operator Models and Technological Enablers,"* EURASIP Journal on Wireless Communications and Networking, doi: 10.21203/rs.3.rs-430193/v1.

International Conference Communications

[C1] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Proactive Resource Scheduling for 5G and Beyond Ultra-Reliable Low Latency Communications,"* IEEE 95th Vehicular Technology Conference: (VTC2022-Spring, doi: 10.1109/VTC2022-Spring54318.2022.9860872.

[C2] **L. N. Dinh**, I. Labriji, M. Maman, and E. Calvanese Strinati, *"Toward URLLC with Proactive HARQ Adaptation,"* in 2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), pp. 220–225. doi: 10.1109/EuCNC/6GSummit54941.2022.9815615.

[C3] **L. N. Dinh**, R. Bertolini, M. Maman, *"Dynamic Resource Scheduling Optimization for Ultra-Reliable Low Latency Communications: From Simulation to Experimentation,"* in 2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), sept. 2022, p. 1026-1031. doi: 10.1109/PIMRC54779.2022.9977893.

[C4] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Hybrid Radio Resource Management based on Multi-Agent Reinforcement Learning,"* accepted in 2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Gothenburg, Jun. 2023.

Extra-thesis Collaborations

[C5] F. Poli, **N.L Dinh**, V. Mannoni, and B. Denis, *"Evaluation of 5G-NR V2N Connectivity in a Centralized Cooperative Lane Change Scenario,"* IEEE 95th Vehicular Technology Conference: (VTC2022-Spring), doi: 10.1109/VTC2022-Spring54318.2022.9861012.

Contents

List of Figures

List of Tables

List of Acronyms

2G Second Generation.
3G Third Generation.
3GPP 3rd Generation Partnership Project.
4G Fourth Generation.
5G Fifth Generation.
6G Sixth Generation.

ACK Acknowledgement.
AI Artificial Intelligent.
AMC Adaptive Modulation and Coding Scheme.
AOA Angles Of Arrival.
AOD Angles Of Departure.
ASA Angle Spreads at Departure.
AWGN Additive White Gaussian Noise.

BF Beam-Forming.
BLER BLock Error Rate.
BSR Buffer Status Report.
BWP Bandwidth Part Manager.

CB Code Block.
CBLER Code BLock Error Rate.
CBS Code Block Size.
CDF Cumulative Distribution Function.
CN Core Network.
CP Cyclic Prefix.
CQI Channel Quality Indicator.
CRC Cyclic Redundancy Check.
CSI Channel State Information.

DCI Downlink Control Information.
Dec-POMDP Decentralised Partially Observable Markov Decision Process.
DL Downlink.
DM Decision Maker.
DMRS DeModulation Reference Signal.
DNN Deep Neural Network.
DQN Deep Q-Networks.
DRL Deep Reinforcement Learning.
DS Delay Spread.

E2E End-to-End.
ECR Effective Code Rate.
EESM Exponential Effective SNR Mapping.
eMBB Enhanced Mobile Broadband.
EPC Evolved Packet Core.
ESM Effective SNR Mapping.

FDD Frequency Division Duplex.

GB Grant-Based.
GCS Global Coordinate System.
GF Grant-Free.
gNb Next Generation NodeB.
GTP GPRS Tunnelling Protocol.

HARQ Hybrid Automatic Repeat reQuest.
HARQ-CC HARQ Chase Combining.
HARQ-IR HARQ Incremental Redundancy.

InF Indoor Factory.
IoT Internet-of-Thing.
IP Internet Protocol.

K Ricean K Factor.
KPIs Key Performance Indicators.

L2S Link To System.
L2SM Link To System Mapping.
LAA Licensed-Assisted Access.
LCS Local Coordinate System.
LDPC Low Density Parity Check Code.
LOS Line-Of-Sight.
LTE Long Term Evolution.

MAC Medium Access Control.
MADDPG Multi-agent Deterministic Policy Gradient.
MADQL Multi-agent Deep-Q Learning.
MANO MANagement and Orchestration.

MARL Multi Agent Reinforcement Learning.
MCS Modulation and Coding Scheme.
MDP Markov Decision Process.
MEC Mobile Edge Computing.
MIMO Multiple Input Multiple Output.
ML Machine Learning.
MMSE Minimum Mean Square Error.
mMTC Massive Machine-Type Communication.
MPC Multiple Path Component.
MR Maximum Rate.

NACK Non-Acknowledgement.
NFV Network Function Virtualization.
NLOS Non-Line-Of-Sight.
NR 5G New Radio.
NR New Radio.
NS3 Network Simulator NS-3.

O-RAN Open-RAN.
OAI OpenAirInterface.
OFDM Orthogonal Frequency Division Multiplexing.
OFDMA Orthogonal Frequency-Division Multiple Access.

PDCCH Physical Downlink Control Channel.
PDCP Packet Data Convergence Protocol.
PDSCH Physical Downlink Signal Channel.
PDU Packet Data Unit.
PF Proportional Fair.
PHY Physical Layer.
PLA Planar Linear Array.
POMG Partially Observable Markov Game.
PRB Physical Resource Block.
PUCCH Physical Uplink Control Channel.
PUSCH Physical Uplink Signal Channel.

QoE Quality of Experience.
QoS Quality of Service.

RAN Radio Access Network.
RB Resource Block.
RBG Radio Block Group.
RE Resource Element.
RL Reinforcement Learning.
RLC Radio Link Control.
RMS Root Mean Square.
ROHC RObust Header Compression.
RR Round Robin.
RRC Radio Resource Control.
RRM Radio Resource Management.
RSRP Reference Signal Received Power.
RTT Round Trip Time.
RTX Retransmission.
RV Redundancy Version.

SAP Service Access Point.
SCM Spatial Channel Model.
SCS Subcarrier Spacing.
SDAP Service Data Adaptation Protocol.
SDN Software Defined Networking.
SDR Software Defined Radio.
SDU Service Data Unit.
SF Shadow Fading.
SGW/PGW Service Gateway and Packet Gateway.
SINR Signal-to-Interference-plus-Noise-Ratio.
SLS System Level Simulation.
SN Sequence Number.
SNR Signal-to-Noise-Ratio.
SR Scheduling Request.
SRS Sounding Reference Signal.

TB Transport Block.
TBLER Transport BLock Error Rte.
TBS Transport Block Size.
TDD Time Division Duplex.
TDMA Time-Division Multiple Access.
TTI Transmission Time Interval.

UCI Uplink Control Information.
UDP User Datagram Protocol.
UE User Equipment.
UL Uplink.
URLLC Ultra Reliable and Low Latency Communications.
USRP Universal Software Radio Peripheral.

V2X Vehicle-to-Everything.

ZF Zero Forcing.
ZOA Zenith Angles Of Departure.
ZOD Zenith Angles Of Departure.
ZSA Angle Spreads at Arrival.
ZSD Zenith Angle Spreads at Departure.
ZSD Zenith Angle Spreads at Arrival.

List of Symbols

$H_{u,s}(t,\tau)$	Channel impulse response between antenna u of tx and antenna s of rx.
K_1	Processing delay at UE in terms of time slot(s).
K_{cb}	maximum Code Block Size (CBS).
K_{max}	Maximum number of allowed retransmission.
L_{12}	Processing delay at gNb in terms of time slot(s).
L_{RTT}	Round Trip Time delay.
L_a	Over the air latency.
L_{e2e}	End-to-End Latency.
L_q	Queue latency.
L	Application packet size generated from sender in Bytes.
Q_m	Modulation order of the selected modulation and coding scheme index m..
R_e	Effective Coding Rate of selected modulation and coding scheme index m..
S_{rx}	PLA antennas installed at receiver side.
U_{tx}	PLA antennas installed at transmitter side.
$\Delta\boldsymbol{\Theta}(t)$	One-slot conditional Lyapunov drift function..
$\boldsymbol{\Phi}$	Chernoff bound on the probability of error.
α_a	Control parameter for RLC queue Q_1 and MAC queue Q_2.
α	Control Parameter in Lyapunov Optimization (Adaptive, Reliability-aware)..
β	Parameter to adapt Effective SINR calculation for different MCS.
$\boldsymbol{\Theta}(t)$	Vector of Queue State in Lyapunov 1.
ϵ_t	Reliability target.
$\eta_{R,eff}$	Resource Efficiency when R parallel retransmission is set.
γ_ω	Signal-plus-Interference-to-Noise-Ratio for a single Resource Block ω.
γ_{eff}	Effective Signal-plus-Interference-to-Noise-Ratio.
γ_s	Signal-plus-Interference-to-Noise-Ratio of transport block (re)transmission..
γ_t	Target Signal-plus-Interference-to-Noise-Ratio.
λ_{ON}	arrival rate of IP packet in ON period..
λ	Traffic rate generation from the sender in [Packets/second]..
ν	Control Parameter in Lyapunov Optimization..
$\overline{\zeta}$	Long-term average risk of Transport Block (TB) loss.
$\zeta(a_j)$	Risk of packet loss under action a_j.
ζ_n	Instantaneous risk of Transport Block (TB) loss.
a_j	Action number j-th.
a_{max}	Maximum number of actions.
n_{rb}	Number of allocated resource blocks for user..
n_{refSc}	Number of reference subcarriers carrying reference signal per RB..
n_s	Number of allocated OFDM symbol for user..
r_{a_j}	Proactivity level of action a_j.
r_{max}	Maximum proactivity for each cluster.
r_{min}	Minimum proactivity for each cluster.
t_{off}	OFF period of ON-OFF traffic..
t_{on}	ON period of ON-OFF traffic.
t_s	Time slot duration [second].

CHAPTER 1

Introduction

CHAPTER CONTENTS

1.1 Ultra Reliable and Low Latency Communications: Motivations and Research Orientations

1.1.1 Motivations

Ultra Reliable and Low Latency Communications (URLLC) provides unprecedented supports to guarantee stringent reliability and latency requirements for mission and safety critical applications. The core enablers of URLLC are based on plethora of advanced techniques such as short packet transmission, grant-free mechanisms and diversity-based solutions [1]. However, these studies are carried out in stationary and controlled environments. They lack a deep understanding of wireless channel dynamics, elastic orchestration of heterogeneous service networks, flexibility in topology changes, and effective management of available radio resources [2]. These limitations, with the conflicting trade-offs between low-latency solutions and ultra-high reliability designs, exacerbate the complications of real-world URLLC applications.

In order to address these problems, our work focuses on improving the trade-off between low latency, high reliability while considering the effective use of radio resources and increased network throughput in the dynamic wireless environment at the RAN level. The main contributions of our work can be divided into three phases. The first phase is dedicated to understanding the key technologies that support URLLC. We try to evaluate the performance of combined solutions in the light of System Level Simulation (SLS) based on NS3 [3]. In the second phase, we developed optimisation algorithms for the proactive, adaptive and reliability-aware allocation of resources to deal with the URLLC in the dynamic scenarios. Our objective is to enhance the fundamental trade-off between latency, reliability and efficient usage of resource. Our solutions in this phase are fully exploited on the basis of full-stack simulation (NS3) and hardware experimentation (OpenAirInterface (OAI)). Ultimately, we deal with hybrid centralised/decentralised radio resource management in favour of Multi Agent Reinforcement Learning (MARL) framework in the last phase. By leveraging shared, opportunistic resources which is jointly managed by each agent (UE) in conjunction with controlled, scheduled resources managed by centralised gNb, we show the performance gains in term of RAN latency, transmission reliability and network throughput. Before going to the details of our contribution, we will briefly describe in this chapter the major challenges, current research trends and comprehensive survey on state-of-the-art research.

1.1.2 5G Networks: A technological breakthrough

Future mobile networks are expected to support exponentially growing number of connected devices as well as wide range applications. By 2030, it is forecasted that around 80 billion connected devices will be included in a network [5]. Besides, the novel applications are developed to support various use cases such as automotive and mobility, transportation, healthcare system, energy industry, media and entertainment ecosystem as illustrated in Figure 1.1. Although Fourth Generation (4G) technologies have been massively, commercially used worldwide and enable us to achieve better data capacity and transfer speed when compared to anterior mobile generations, the proliferation of such new use cases, which demand the improvements in terms of latency, reliability, and scalability, are posing unprecedented obstacles for 4G systems. Therefore, Fifth Generation (5G) technologies roll out to enhance network performance such as mobility, energy, speed, range of services, increased reliability, lower latency and higher throughput [6]. The detailed differences between the 4G and 5G networks are shown in Table 1.1.

As in report ITU M.2083-0 [7], the applications which are supported by 5G technologies can be classified into 3 main scenarios:

- Enhanced Mobile Broadband (eMBB): provides high throughput data services (up to 20 Gbps in Downlink [8]) for applications. It provides a foundation for the stable connections in very high

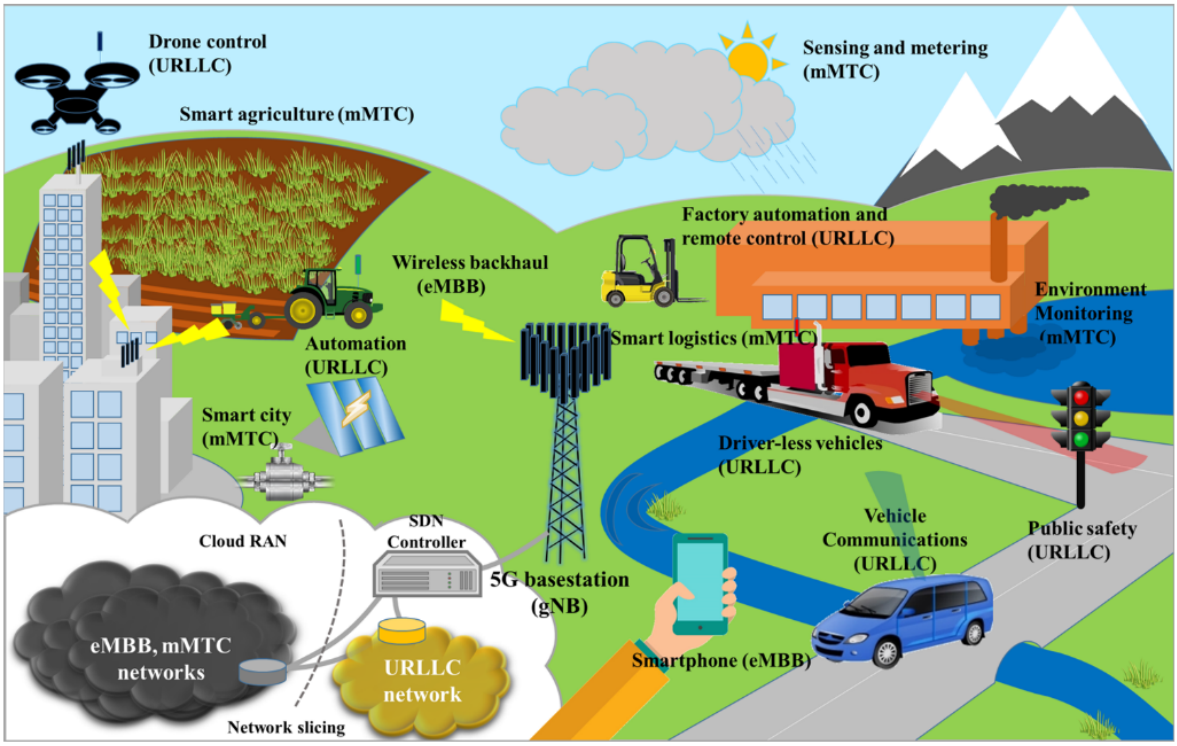

FIGURE 1.1: *Overview of 3 key services supported in 5G networks [4]*

TABLE 1.1: *Comparison between 4G and 5G [2]*

	4G	**5G**
Metadata	Important	Crucial
Packet size	Long (MBB)	Short (URLLC)-Long(eMBB)
Design	Throughput-centric, Average delay good enough	Latency and Reliability centric / tails are matter
Reliability	95 %	$1 - 10^{-x}$, $x = \{3, ..., 9\}$
Rate	Shannon (Long packets)	Rate loss due to short packet
Delay violation	Exponential delay using effective bandwidth	Faster decay than exponential
Latency	15 ms RTT based on 1 ms subframe	1 ms or less, shorter TTI, HARQ RTT
Queue size	unbounded	bounded
Band	sub-6GHz	Above- and sub-6GHz
Scale	Few users/devices	billion devices

peak data rate to multimedia content, services and data (e.g. 3D video, Ultra-High Definition (UHD) screens, augmented reality, etc.)

- Ultra Reliable and Low Latency Communications (URLLC): aims at supporting applications with stringent requirements in throughput, latency, reliability and availability. Such use cases range from industrial automation, mission-critical applications, self-driving cars, etc.
- Massive Machine-Type Communication (mMTC): are applicable for scenarios where a large number of connected devices transmitting sporadic data (unpredictable small data payloads) and relatively low volumes of non-delay sensitive data (e.g. smart grid, smart home/building, smart cities, etc.)

Figure 1.2 illustrates the importance of each capability for the aforementioned usage scenario in 5G [9]. This was obtained by using three -step scaling: "low", "medium" and "high". For eMBB use cases, energy efficiency, area traffic capacity, peak data date, user experience data rate, spectrum efficiency are highly emphasized and are classified as high importance categories. In the case of mMTC, the priorities are set to connection density and network energy efficiency for massive deployment. Finally, the order of latency in User Plane and Control Plane as well as mobility with low interruption time are highest importance for URLLC communications.

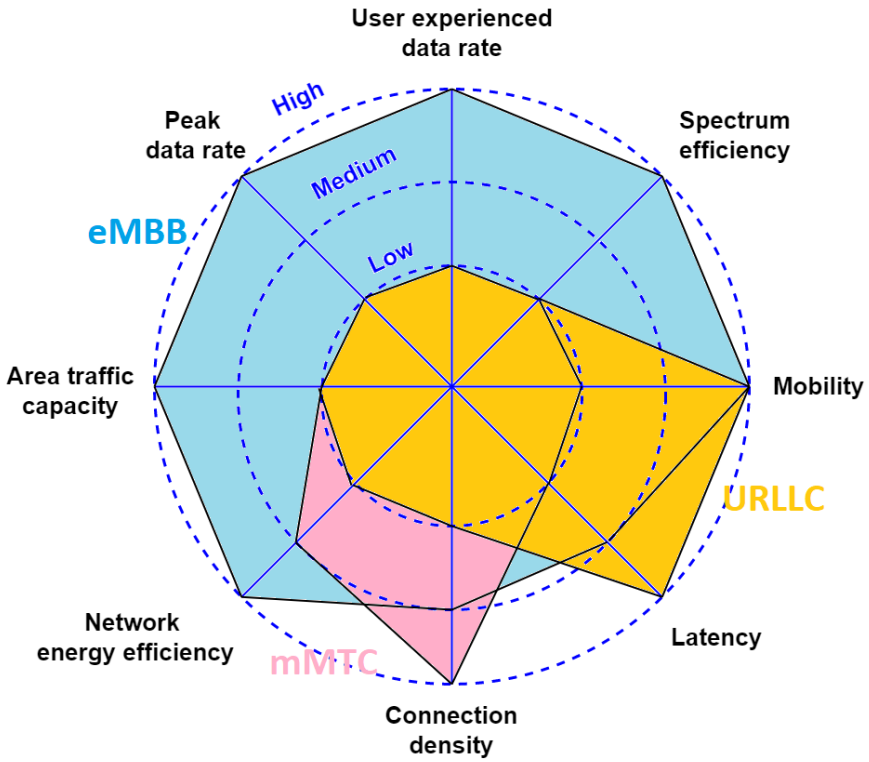

FIGURE 1.2: *Key capabilities in 3 main scenarios in 5G [9].*

1.1.3 Ultra Reliable and Low Latency Communications

As name suggested, Ultra Reliable and Low Latency Communications (URLLC) aims to accommodate emerging services and applications having both stringent latency and reliability requirements such as intelligent transportation and industry automation [10], tactile internet [11], augmented/virtual reality (AR/VR) [12], fault detection [13], frequency and voltage control in smart grids [14]. Each application category generally has different requirements for End-to-End (E2E) latency and reliability, as shown in Table 1.2 [1]. Thus, 3GPP recommends a general requirement for URLLC communications that the transmission of a packet of 32 bytes should be reliably bounded in 1-10^{-5} within a user plane latency of 1ms

[8]. In this section we will analyse the terms "latency" and "reliability" and the relationships between them with regard to the design of a network.

Firstly, latency is considered as the End-to-End (E2E) time spent in the radio network, backhaul connection and transport/core network together, as follows:

- Radio delay is mainly concerned at RAN level where over-the-air transmission delay, queuing delay, processing/computing delay with retransmission are taken into account.
- Back-haul connection delay is the time for establishing connections between gNb and the Core Network (CN).
- Transport and core delay involve the time spent of the packet in the router networks and processing time taken by the core network.

The latency terms in URLLC can also be defined as user plane latency or control plane latency [15]. The former is seen as the one-way time taken to successfully deliver a packet from the radio protocol layer ingress point to the radio interface ingress point, regardless of UL or DL communication of a given service under unloaded conditions. On the other hand, control plane latency is also defined as the transition time from the idle state to the start of continuous data transmission, i.e. the active state. As recommended, the minimum (single)-user plane latency requirements are 4ms and 1ms for eMBB and URLLC, respectively, while the minimum required control plane latency is 20ms [2].

With regards to reliability, it is defined as the probability that a certain amount of data will be successfully transferred within a certain amount of time. This definition of reliability stipulates that packet delivery must happen within latency bounded. Besides that, there are several other definitions of reliability that can be encountered:

- Reliability per node: is defined as the probability of transmission error and/or queuing delay violation.
- Control channel reliability: is defined as the probability of successful decryption of the transmitted metadata.[4].
- Availability: is defined as the probability that a given service is available. For example, 99.99% availability means that one user out of 10000 nodes might not be properly served.

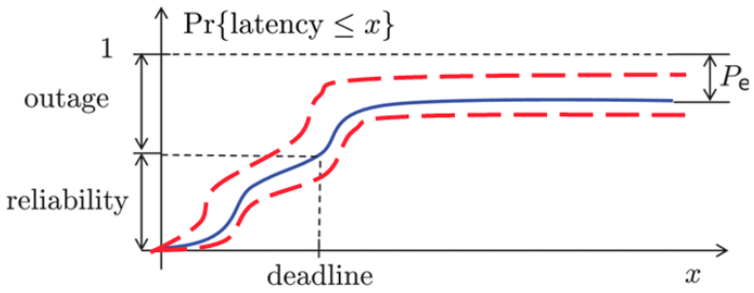

FIGURE 1.3: *Relation between reliability and latency [16]*

The tighten relationship between latency and reliability is shown in Figure 1.3 [16] where a pre-configured deadline is defined as constraint latency of each application. Without the deadline (i.e, there is no delay target), the perfect reliability is easily achieved whenever transmission rate is below channel capacity according to well-known Shannon's theory. On the other hands, in case of a predefined latency constraint is set for a particular application, then reliability is seen as the probability that latency does not exceed the deadline. The definition of outage is complementary to the reliability and it shows the percentage of deadline violation. The blue curve illustrate that probability of delivering information within a deadline x. It is also noted from Figure 1.3 that the stricter deadline is, the lower reliability (i.e. higher outage) is capable of. This phenomenal perfectly demonstrates the fundamental trade-off between low latency and high reliability communications. Furthermore, in wireless environment where statistics of extreme and

TABLE 1.2: *Use case requirements [1]*

Use case	Latency (ms)	Reliability (%)	Data size (bytes)	Communication range (m)
Smart grid	3 ∼ 20	99.999	80 ∼ 1000	10 ∼ 1000
Self-driving car	1	99	144	400
Industrial automation	0.25 ∼ 10	99.9999999	10 ∼ 300	50 ∼ 100
Process automation	50 ∼ 100	99.99	40 ∼ 100	100 ∼ 500
E-health	30	99.999	28 ∼ 400	300 ∼ 500
Augmented reality	0.4 ∼2	99.999	12k ∼ 16k	100 ∼ 400
ITS	10 ∼ 100	99.999	50 ∼ 200	300 ∼ 1000
V2V	5	99.999	1600	300
Tactile internet	1	99.99999	250	100000

rare events are also involved, the variation of blue curve will be bounded between slash red curve and complicate the studies on URLLC trade-off.

1.1.4 Technological constraints and Design Challenges of URLLC

Among the three main categories of 5G communication, the protocol designs of the URLLC service are probably considered to be the most challenging and problematic, because we have to deal with two trade-off requirements: *low latency* and *ultra-high reliability*. In addition to these fundamental trade-off constraints, a number of technical challenges remain to be addressed. These include coexistence with heterogeneous services and the ability to integrate with new architectures. In this section, we will point out some of the difficulties involved in the design of a proper URLLC system.

1.1.4.1 Fundamental trade-off constraints

The primary requirement for the achievement of URLLC is to reliably deliver small payloads of data within a strict and constraining latency target. Obviously, the low latency goal can be achieved by shortening the transmit packets. However, the additional cost of degrading the channel coding gain with respect to the shorter packet length prevents the reliability of the transmission from reaching the predefined goal **[17]**. Furthermore, the control overhead is often reduced for small packet sizes, thus, we may not be able to provide the same level of reliability as the long packet transmission where control overhead is sufficient. It represents the first challenge in addressing the fundamental compromise between the goals of low latency and high reliability designs. As an example to illustrate this obstacle in the design of an URLLC system, while other services in eMBB can simply rely on Hybrid Automatic Repeat reQuest (HARQ) retransmission to achieve high reliability, it is not the case for URLLC when multiple retransmissions easily violates hard latency constraint.

Another dimensional variable that needs to be taken into account, in addition to the fundamental trade-off between reliability and latency, is the efficient allocation of resources **[18]**. Resource efficiency is defined as the ratio of resources required to accomplish packet transmission to those provided. In order to compensate for the loss of reliability in a dynamic environment, feedback-based retransmission systems are traditionally implemented (e.g., HARQ) in which ACK/NACK messages are used to indicate whether additional resources are needed to correct the corrupted packet or not. Principally, this protocol design achieves efficient resource allocation at the cost of high delay due to the regular exchanges of acknowledgements between sender and receiver. By proactively (re)transmitting a large number of packets, equivalent to multiple resource allocations per acknowledgement, we can overcome negative

channel impacts while reducing exchange overhead at the cost of inefficient resource allocation. Thus, efficient resource allocation in meeting URLLC requirements in dynamic scenario remains a challenge to be addressed with.

The fourth variable trade-off that attracts a lot of attention in the URLLC design is the network capacity rate. Generally speaking, the guarantee of high data rate incurs lower reliability and vice versa **[19]**. For example, given the physical link between transmitter and receiver, a higher data rate is achieved by transmitting TB at higher MCS. However, for successful decoding, a higher Signal-to-Interference-plus-Noise-Ratio (SINR) is required for data reception. Therefore, if a higher MCS is set for a transmission, the probability of correct packet reception is lower. As a consequence, reliability and latency constraints in dynamic scenarios (e.g. channel fading, user mobility, etc.) limit the choice of high MCS (e.g. higher data rate) and the trade-off between these requirements needs to be reconsidered.

In addition to the fundamental trade-offs discussed above, there are several key aspects to be aware of, such as the power consumption of the device, control overload and user density, etc **[2]**. However, in the context of our thesis, our focus is on the trade-off between RAN latency, transmission reliability, radio resource efficiency and network throughput.

1.1.4.2 Heterogeneous requirements of services

The deployment of URLLC services together with eMBB and mMTC applications makes a burden on 5G wireless network because different services require particular metrics and transmission policies. Although the traffic for URLLC is utmost prioritised, it is important not to degrade the performance of other services. Moreover, the transitions of various transmission policies require flexible frame structure and multiplexing schemes for all services while current static or semi-static resource allocation algorithms are not flexible enough. Thus, it leads to the requirement of design a dynamic multiplexing scheme where the satisfaction of each on-going service is guaranteed. In order to do that, a promising approach is to apply network slicing, which is a virtualisation approach allowing multiple logical slices to run on the same shared network infrastructure. However, wireless resources are fundamentally different due to their shared nature, and what the literature fails to mention is the efficient allocation of resources to guarantee the desired QoS for specific services. Thus, the optimal co-existence of URLLC applications in parallel with eMBB and mMTC services needs to be further discussed.

In this thesis we open the research direction towards optimal, proactive and hybrid centralised/decentralised 5G networks to guarantee URLLC communications. The heterogeneous support of other types of services is left to future developments that may arise from this thesis.

1.1.4.3 Incorporating new architectures

From the network perspective, URLLC applications may benefit the design of MEC and edge caching which offer a platform to offload intensive computations from end users to the nearby edge network. However, edge computing network functions are incompatible with communication functions of the original URLLC traffic, thus a modification of current protocol stack to accommodate edge computing services are required. Furthermore, edge caching resources are usually distributed and limited when are compared to caching resources in the cloud, while the amount of contents are unlimited and fast-changing. Therefore, it is important to employ advanced prediction features to anticipate the most popular contents and effectively use limited edge cache sizes.

In this thesis, we focus only on the RAN level where wireless protocols for URLLC are carried out. The extension of this work towards MEC and CN to form an architecture solution will be devoted for future developments.

1.1.5 Current Research Orientations

In the literature, researcher propose various techniques/approaches for achieving low latency and/or high reliability in three main categories: *(1)* RAN solutions, *(2)* core network solutions and *(3)* MEC and caching solutions.

1.1.5.1 RAN solutions

In RAN, the objectives of low latency and high reliability are usually achieved separately by different enhancement solutions in Physical Layer (PHY) and Medium Access Control (MAC) as follows:

- Novel frame and packet structure: is considered as a solution in PHY layer to reduce TTI from 1 ms in LTE to 0.125 ms as in NR. It is achieved by using fewer OFDM symbols per TTI **[20]** , shortening OFDM symbols via broaden Subcarrier Spacing (SCS) **[21]** and/or reducing HARQ Round Trip Time (RTT).
- eMBB/URLLC/mMTC multiplexing: It is seen as an effective method of resource allocation in the MAC layer for dealing with heterogeneous QoS requirements. This method typically prioritises resources for mission-critical services by preempting the currently allocated resources of other, lower-priority services (e.g: puncturing the current eMBB transmission **[22]**)
- Hybrid GF/GB access: Traditionally, GB access to the wireless channel ensures that users have collision-free access to the resource. This comes at the cost of high latency and overhead. In GF random access, the handshake-based grant acquisition phase is omitted. Thus, access latency is reduced with the risk of resource selection collisions between users. **[23]**. The trade-off between these access schemes is therefore favourable and requires further study to design a low-latency, high-reliability system.
- Non-orthogonal multiple access (NOMA): shortens latency by supporting more users than conventional orthogonal resource based approaches. By leveraging power or code domain multiplexing and using successive interference cancellation (SIC), multiple users can share the same time/frequency resource block and achieve higher spectral efficiency, energy efficiency and lower latency. On top of that, grant-free NOMA has been introduced as a promising technology for URLLC supporting time-critical applications **[24]**. However, various problems with regards to imperfect CSI, user ordering, processing delay due to resource overlapping and other dynamics are considered as bottlenecks of this solution.

On the other hands, the negative effects that affect the reliability of wireless transmissions are caused by various factors, such as (1) interference and the coexistence of multiple users sharing the same frequency bands. This causes unpredictable signal collisions between users and increases the error rate of wireless transmissions. (2) Reception signal strength fluctuates due to time-varying fading channels. It is caused by the scattering effects of signal propagation in a dynamic environment or by user mobility. In order to address with those nuanced problems, diversity and beamforming play a crucial role to boost the received SINR. The reliability enabler solutions can be listed as follows:

- Multi-connectivity: is of paramount importance to overcome the channel impairments and ensuring high reliable communication. Based on the simultaneous connection of UE to multiple gNb, macro-diversity is exploited to combat large scale fading as well as blockages in the dynamic wireless environment **[25]**.
- massive-MIMO + mmWave: are mutual, complementary solutions to provide better reliable communication, and extreme throughput. Due to the very high frequency of wireless communication, the susceptibility of blockage and signal attenuation are the main sources of signal degradation and lowering the network coverage. However, the short wavelength of mmWaves paves the way for the integration of more antenna elements which make massive MIMO practically feasible **[26]**.

- HARQ + short frame duration: are methods to both ensuring highly reliable and low latency transmissions based on retransmission in shorten duration. For a latency budget of 1 ms, a 0.125 ms TTI is used to allow up to 4 retransmission attempts in which half of them are reserved for DL and UL communications [27]. However, time-critical packets may not benefit from retransmissions, thus 5G public private partnership (5G PPP) project METIS-II suggests turning off the HARQ functions to achieve URLLC in certain scenarios [28].
- Network slicing refers to the process of slicing physical network into logical and dedicated subnetworks for specific applications. By doing that, the interference between services can be reduced and ultra reliable transmission services can be guaranteed [29].

In the scope of our thesis, we mainly focus on the redesign of frame and packet structure, the hybrid GB/GF access and allocation resource management to guarantee URLLC objectives.

1.1.5.2 Mobile Edge Computing (MEC) and Caching solutions

One of the main reasons which cause long-delay of communication is seen at the insufficient capacity of back-haul links in peak-traffic hours. To overcome this impairment, Caching and Mobile Edge Computing (MEC) exploitation are considered as one of the promising candidate technologies. By exploiting caching and computing resources in proximity to the network edge, it has been proved by Bastug et al. [30] that latency can be significantly reduced. This implementation is indispensable for the advent of resource-intensive applications such as augmented/ virtual reality and other mission critical applications such as autonomous driving. The integration of SDN/NFV technologies also facilitates the deployment of MEC technologies which decrease the latency. Ford et al. [31] deploy distributed MEC by the placement of network functions at a decentralized manner. Their demonstrations show that 75 % capacity at the data center could be reduced and meet the 5G latency requirement.

In terms of caching, there are 4 main categories that can be classified as (1) local caching [32], (2) device to device (D2D) caching [33], (3) small cell base station (SBS) caching [34] and (4) macro base station (MBS) caching [33], depending on where the requested contents of the user are temporally stored. The fundamental problem in which caching solutions aim to solve is the trade-off between latency and storage given the fact that the more contents are cached, the lower delay the devices may take to access to the information content at the cost of higher storage capacity. By introducing novel metrics such as normalized delivery time (NDT) [35], fractional delivery time (FDT) [36] and delivery time per bit (DTB) [37], this fundamental trade-off take the upper bound or lower bound of the latency/storage capacity into account for the algorithm designs.

Even though MEC and caching solutions are envisioned to reduce huge load computation at the server as well as the latency and reliability, there are several open issues that encourage researcher to study. For instance, cooperate executing computation tasks in multiple edge nodes may be complexities when dealing with various channel qualities and computational capabilities of different nodes. Furthermore, given the fact that the satisfaction of reliability and latency is usually difficult in multiple edge nodes environment, it is a challenge to design an efficient offloading scheme considering both computations and communications of the task [38].

In our thesis, we focus primarily on the communication aspects of the URLLC rather than the computational and storage capabilities taking advantage of the MEC architecture, so future development can be opened up from this work for feasible integration of the MEC to improve heterogeneous URLLC communications.

1.1.5.3 Core networks solutions

In the core network, several new entities such as SDN and NFV [39] are introduced to flexibly support larger capacity, massive connectivity and low latency with seamless operation [40]. Because of the fact

that control plane and data plane have different network QoS criteria to be met, so it is preferable to decouple them completely to design them effectively. With the exploitation of SDN and NFV, the user plane and control plane separation are facilitated. Particularly, the virtualization in SDN-based system allows handling of QoS by setting specific rules in the switches along with the data path. Along with that, NFV removes the dependency on the hardware platform and make flexible deployments of core networks as well as sharing resources in RAN. Once control plane is separated from data plane, an information centric scheme is introduced by Liang et al. **[41]**. In this architecture, various controller in terms of wireless network architecture, radio spectrum resource, virtual resources including content-level slicing, network-level slicing and flow-level slicing have been presented as key enablers to support low latency services. Another advantages of SDN/NFV integration into core network are applied at mobility management functions to lower its processing delay. By implementing proactive and reactive solutions for mobility management using Mininet and Openflow, Marquezan et al. in **[42]** successfully stretch global processing delay around the mean value with high probability (95 %). Their results suggest a potential solution to guarantee the E2E jitter of communications. Although the huge potentials of SDN/NFV based core network design, there are several issues which need to be addressed with. As Casellas et al. **[43]** suggested, the main challenge is the management and orchestration of the heterogeneous resources. Effective resource allocation and implementation of network functions in this heterogeneous environment, while maintaining low latency and high reliability of E2E communication are still open issues. For the sake of simplicity, we leverage the EPC in LTE network for modelling the functionalities of CN.

1.2 Thesis outline and main contributions

The initial objective of this research is to investigate novel transmission and allocation strategies involving the Physical Layer (PHY), Medium Access Control (MAC) and Radio Resource Management (RRM) layers that provide a flexible trade-off between reliability and latency.

In the first part of my thesis, I have carried out the state-of-the-art studies on potential enabling technologies for URLLC. Then, I have proposed a methodology to classify them in multi-diversity layer across network layer. Following bibliographic works, I have classified potential solutions as a function of network layers (RAN, MEC, CN) and diversity (coding, time, frequency, space and interface), addressing latency reduction, reliability improvement or both. To facilitate the metric measurements and quantify the satisfaction of QoS in URLLC transmission, I have clarified and set up KPIs to see the impact of each network layer on latency with associated jitter, reliability, resource utilisation, energy, coverage and complexity.

Thanks to the preliminary studies on KPIs and mechanisms, I have proposed a complete framework of orchestrator, integrating Artificial Intelligent (AI) and smartly combining the appropriate mechanisms using a reactive or proactive approach in order to guarantee URLLC requirements in a dynamic environment. The integration of the smart management thanks to AI will be useful to manage the entire network and to face to a myriad of (un)predictable events which impact on the global performance. Thanks to the NS3 system simulator, we have developed a sub-6GHz/MIMO network framework capable of combining several mechanisms exploiting modulation and coding diversity (LDPC, MCS, AMC), spatial diversity (MIMO) and time diversity (frame design, HARQ) at the RAN level for URLLC communication. The performance has been evaluated in terms of transmission reliability, E2E latency and jitter for dynamic traffic and a fast fading channel in an indoor factory scenario **[44]**. The results obtained allow us to understand the behaviour of different combinations towards URLLC performance and are a first step for further research using AI.

After promising findings and results from precedent works, we have designed a (i) Decision Maker (DM) (orchestrator) trading off reliability – latency – resource efficiency and (ii) jitter-aware orchestration for URLLC communications. Afterwards, we have seen the benefits of using our orchestrator design to

improve the proactive resource allocation strategy, lower latency communications as described in Chapter 2.

In the last piece of our work, we manage to extend our centralized Decision Maker (DM) framework to (semi)decentralized one in which each user (agent) plays an pivotal role in making an optimal decision in hybrid allocation schemes and optimize the UL latency, network throughput and reliability. Relying on the Lyapunov optimization framework that is exploited in previous works and Multi Agent Reinforcement Learning (MARL), each user can perform optimal resource selection actions to achieve better performance.

The detailed outline of the thesis is as follows:

- **Chapter 1: Introduction** provides a philosophical reasons of 5G designs to deal with heterogeneous requirements of growing network dimensions. Among the three main usage scenarios that 5G network supports, we emphasise that the design requirements for URLLC communication are much more challenging. Then, we focus on the recent obstacles and design challenge in which current studies are limited to support URLLC requirements. Afterwards, we carry out a comprehensive survey of state-of-the-art research on URLLC, which lays the foundation for our thesis.

- **Chapter 2: Enabling technologies for Ultra Reliable and Low Latency Communications** aims at providing a methodological design of an orchestrator to flexibly apply the joint mechanisms acrossing network layers and guaranteeing E2E low latency, ultra-reliable communications at RAN level. Based on the studies of cutting-edge solutions for either reducing latency and/or improving reliableness of wireless communications, a comprehensive classification of solutions into 5 dimensional diversity (time, frequency, signal processing, space and hardware) across network architecture (RAN, MEC, Core Network (CN)) is carried out. This classification serves as a look-up table to support orchestrator decisions of functional (de)activations to improve latency or reliability as the dynamic of environment. The evaluations of enabling mechanisms for URLLC at RAN level are also conducted in the light of 5-dimensional diversity mechanisms. Afterwards, we are going to describe our *(i)* early decision maker and *(ii)* jitter-aware scheduling at RAN level. The material of this chapter is recognized in the journal paper [J1], the patents [P1] and [P2]:

  [J1] M. Maman, E. Calvanese Strinati, **L. N. Dinh**, et al., *"Beyond Private 5G Networks: Applications, Architectures, Operator Models and Technological Enablers,"* EURASIP Journal on Wireless Communications and Networking, doi: 10.21203/rs.3.rs-430193/v1.

  [P1] M. Maman, **L. N. Dinh**, E. Calvanese Strinati, *"Method to exploit latency distribution for early decision making,"* filled patent, FR2103542.

  [P2] **L. N. Dinh**, M. Maman, E. Calvanese Strinati, *"Methods and apparatus for jitter-aware scheduling in wireless Time Sensitive Network communications,"* filled patent, .

- **Chapter 3: Ensuring Latency, Reliability and Effective Resource Allocation for URLLC**. The whole of our research in this chapter can be divided into four distinct phases, as follows: *(i)* First, using a system-level simulation based on NS3, we demonstrate how proactive resource allocation can be integrated into the regular HARQ procedure at the MAC layer to reduce the overall latency of the RAN level. In this work, we have clarified the correlation between traffic source throughput and PHY/MAC layer configurations to serve data in a highly dynamic channel. We then highlighted the trade-off between RAN latency, transmission reliability and radio resource efficiency by analysing various proactive resource allocation schemes to re-transmit interrupted packets. Although RAN latency is reduced with higher levels of proactive resource scheduling, the lack of adaptation is the main cause of low radio resource utilisation efficiency. *(ii)* In the next step, an

adaptive control algorithm is proposed to improve the proactive resource management. Based on the Lyapunov stochastic optimisation tool, a mathematical framework is proposed to understand the performance-delay trade-off by minimising the objective function of the total resource allocation and the total queue length, parameterised by a hyper-parameter (i.e.,ν). *(iii)* In the third step, we propose an adaptive and reliability-aware solution to deal with the system where the critical disturbance occurs frequently and more radio resources are needed to improve the reliability of the communication. *(iv)* Finally, by means of OpenAirInterface (OAI) experimentation, we successfully validate the design of algorithms on the real-time and full-stack hardware where the constraints of experimentation are taken into account. The contributions in this chapter are adapted from conferences papers [C1], [C2] and [C3].

[C1] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Proactive Resource Scheduling for 5G and Beyond Ultra-Reliable Low Latency Communications,"* IEEE 95th Vehicular Technology Conference: (VTC2022-Spring, doi: 10.1109/VTC2022-Spring54318.2022.9860872.

[C2] **L. N. Dinh**, I. Labriji, M. Maman, and E. Calvanese Strinati, *"Toward URLLC with Proactive HARQ Adaptation,"* in 2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), pp. 220–225. doi: 10.1109/EuCNC/6GSummit54941.2022.9815615.

[C3] **L. N. Dinh**, R. Bertolini, M. Maman, *"Dynamic Resource Scheduling Optimization for Ultra-Reliable Low Latency Communications: From Simulation to Experimentation,"* in 2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), sept. 2022, p. 1026-1031. doi: 10.1109/PIMRC54779.2022.9977893.

- **Chapter 4: Semi-decentralized resource scheduling for QoS improvement** extends the vision of dynamic decision maker into (semi)distributed resource scheduling frameworks where each end user (agent) together with gNb participate to maximize the global objective of low latency, high reliability and high network throughput. Relying on hybrid allocation schemes of Grant-Based (GB) and Grant-Free (GF), our solution takes the advantages of each access regime to deal with massive URLLC traffic in a dynamic environment. The content of this chapter is adapted from our accepted conference paper.

[C4] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Hybrid Radio Resource Management based on Multi-Agent Reinforcement Learning,"* accepted in 2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Gothenburg, Jun. 2023.

- Finally, **chapter 5: Conclusions and Future works** concludes this manuscript. We also present different research topics that are identified as open issues and future research directions in the improvement of URLLC communications.

CHAPTER 2

Enabling technologies for Ultra Reliable and Low Latency Communications

CHAPTER CONTENTS

2.1 Introduction

2.1.1 Motivations

Many independent mechanisms efficiently increase reliability and/or reduce communication latency in the radio access and core network. These mechanisms can exploit different diversity to improve system performance. However, it is not always preferable to activate all diversity-based mechanisms at once to achieve Key Performance Indicators (KPIs) satisfaction due to the huge resource consumption in terms of power, computation, communication and caching memory. They highlight the need for a controller (e.g. orchestration) designed to intelligently combine the appropriate mechanisms and flexibly manage the complex system with heterogeneous service requirements and efficient use of resources. [45]. In this chapter, we propose a classification of the mechanisms and a methodology to design an orchestrator at RAN to manage classified mechanisms dynamically. Then, we evaluate the performance of current diversity-based mechanisms in achieving URLLC goals with regards to RAN latency and transmission reliability.

Afterwards, we propose two algorithms to enhance the trade-off between reliability and latency taking into account the effective resource management and goal-oriented communication with jitter-aware scheduling. **The former algorithm** deals with the tactical operations of network orchestrator at Radio Access Network (RAN) level. Conventionally, a methodology is proposed for a systematic combination of several mechanisms to stretch the network latency below the deadline. This approach has disadvantages when it is not rapidly adapted to dynamic environment. On the other hand, proactive adaptation strategies select mechanisms for the worst case with margin to compensate the unpredictable decay of unwanted defects. This implies a high cost of ultra-reliable communications because defects/impairments (worst cases) can be very rare. Then, a novel decision maker model, which considers performance statistics and makes a better trade-off between network latency, communication reliability and radio resource efficiency, need to be considered. **The second algorithm** addresses jitter-aware control problem in a complex wireless environment. In particular, the attainment of highly deterministic communication (low jitter) is generally problematic due to the presence of a variety of uncontrollable impairments (e.g. uncertainty of channel dynamics, sporadic and unpredictable user traffic behaviour, mobility and varying number of users, etc.). It demands a jitter-aware orchestration method that mitigates the behaviour of uncontrollable wireless dynamics and improves communication latency/jitter. By doing this, this algorithm is beneficial in application controls when resources (power, radio, computation, caching, storage, etc.) are more concentrated to guarantee the occurrence of data transmission in a controllable time window.

2.1.2 Contributions

The contributions of this chapter can be summarized as follows:

- Our contribution is firstly to classify enabling solutions according to multi-level diversity. The introduction of additional diversity (i.e. hardware and signal processing) to the existing diversity (i.e. time, space, frequency) brings interoperability and allows the combination of several technologies [46]. This classification will act as a look-up table for the combination of diversity that will have an impact on the performance of the network in terms of reliability, latency/jitter.
- Our second contribution is to propose a new methodology to design an orchestrator that takes into account the heterogeneity and coexistence of services, the dynamic evolution of requirements (e.g. traffic, number of users, QoS) and the dynamics of the wireless propagation medium. We then focus on the RAN performance, taking into account the enabling mechanisms at this level. Before considering a AI-based orchestrator, we modified a NS3 network simulator based on [3] to multiplex low-latency mechanisms (e.g., frame design) and reliability enhancing mechanisms

(e.g., multiple antennas, redundancy, and adaptive modulation and coding using code/time/space diversity). This simulator is used to evaluate the impact of the combined mechanisms using a diversity subset at the RAN level working alongside the EPC/LTE core network [J1].

- We propose online decision maker (orchestrator) which defines dynamically one or several thresholds based on latency distribution in order to make a tradeoff between network latency, communication reliability and spectrum efficiency. The objectives are to define when to (de)activate more resources/ mechanisms, to make an efficient tradeoff between reactive and proactive approaches and to exploit multi-modal latency distribution [P1].
- We introduce a jitter-aware orchestration method that forces latency to fall within predetermined windows. As the results, latency/jitter is independent of the wireless environment and now adapted to the control system. Thus, we transform the dependence of communication on the environment into a dependence on control/application. [P2]

The technical content of this chapter is based on our journal paper [J1], the patents [P1] and [P2]:

[J1] M. Maman, E. C. Strinati, **L. N. Dinh**, et al., *"Beyond Private 5G Networks: Applications, Architectures, Operator Models and Technological Enablers,"* EURASIP Journal on Wireless Communications and Networking, doi: 10.21203/rs.3.rs-430193/v1.

[P1] M. Maman, **L. N. Dinh**, E. Calvanese Strinati, *"Method to exploit latency distribution for early decision making,"* filed patent, FR2103542.

[P2] **L. N. Dinh**, M. Maman, E. Calvanese Strinati, *"Methods and apparatus for jitter-aware scheduling in wireless Time Sensitive Network communications,"* filed patent, .

2.2 Classification of enabling mechanisms

Several advanced mechanisms can be used to reduce latency/jitter, improve transmission reliability, increase network throughput and strengthen communication availability. These mechanisms can exploit multi-level diversity based on *(1)* Time, *(2)* Frequency, *(3)* Space, *(4)* Signal Processing and *(5)* Hardware. While time, frequency and space are classic, two additional diversities have been added for more flexibility and elasticity. First, hardware diversity refers to the ability to switch from one technology to another (e.g. antenna selection, Multi-Radio Access Technology (Multi-RAT), hardware selection for virtualisation, etc.). In addition, signal processing diversity refers to software such as data processing redundancy, channel coding, different modulation schemes to achieve more robust transmission on the noisy channel. It could also include Mobile Edge Computing (MEC) **[47]**, softwarization **[48]** and various virtual network functions in the new network architecture to bring computing, communication, processing and storage closer to the end-user. Fig. 2.1 classifies the mechanisms proposed in the literature according to the multi-level diversity and the network layer to which it belongs.

2.2.1 Radio Access Networks

Ultra-reliable communication at the RAN has been widely studied for many years and it is enabled according to the activation of various diversity-based techniques: channel coding in the signal processing domain, redundancy in the time and frequency domain, spatial diversity in the space domain and multi-RAT related to the hardware domain. For example, Liva et al. **[49]** study short block coding schemes (signal processing diversity) for ultra-low error rate transmission of short packets. Afterwards, Chen et al. **[50]** attempt to combine coding schemes and spatial diversity to further improve reliability. In

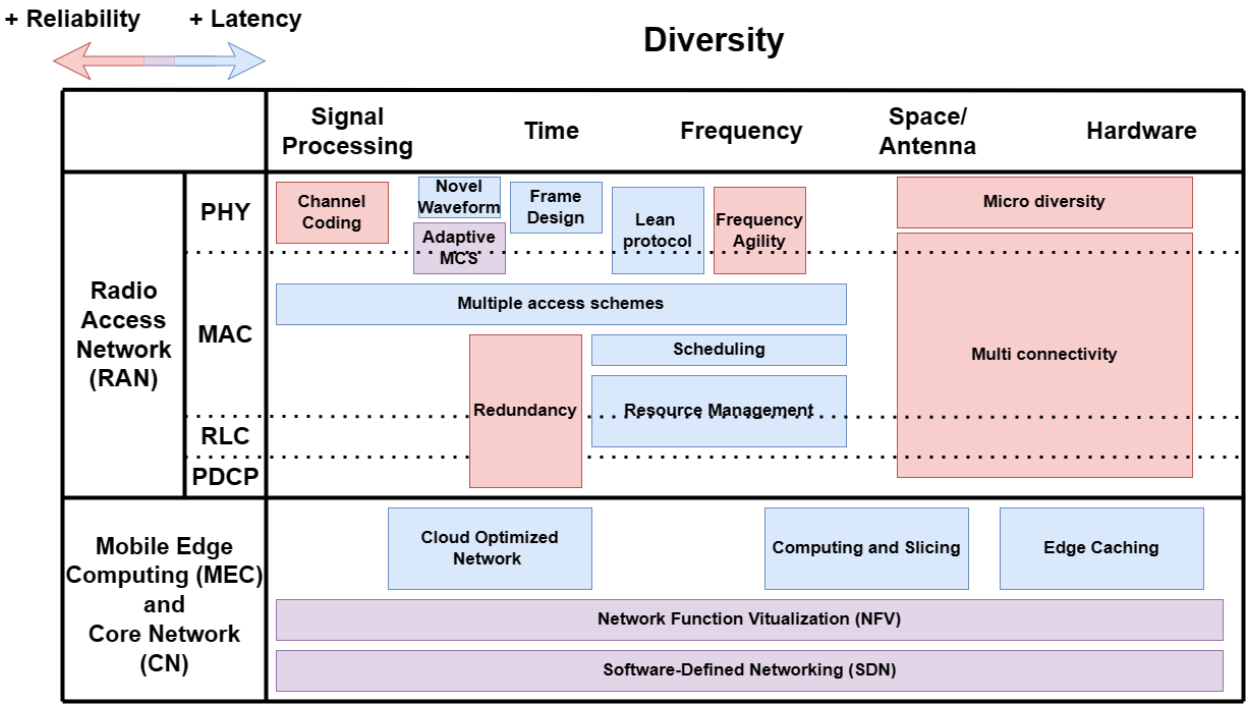

FIGURE 2.1: *Classification of the mechanisms as a function of the diversity and the network layer.*

particular, they combine an analog fountain code with 10 antennas at the receiver to obtain a BLER of 10^{-6}. In addition, antenna diversity, also known as spatial diversity, is seen as a enabler for URLLC by providing more spatial degrees of freedom to improve system reliability. Taking spatial diversity into account, Pocovi et al. describe in [51] that microscopic MIMO schemes and macroscopic diversity are sufficient to guarantee the SINR outage performance. The joint spatial diversity is able to handle fast fading and increases the robustness of the communication by reaching 10^{-5} SINR outage.

Latency reduction techniques can be classified into several categories such as frame design, lean protocol, multiple access scheme and scheduling protocol that relate to the PHY and MAC layers of the RAN. In particular, frame structure design is an important aspect of the 5G PHY and concerns a shortened version of the frame that reduces communication latency without affecting reliability. Also, Li et al. show in [52] that low latency design with guaranteed reliability is achieved by using HARQ with a shorter CSI turnaround time and TTI. Another method to provide high levels of reliability with reduced latency is to use multiple communication technologies ([53], [54]). This hardware-based diversity method can complement signal processing/time/frequency/space-based methods for improved reliability by taking advantage of multi-path signal propagation effects. For instance, dual connectivity, where users connect to primary and secondary cells simultaneously, adds another layer of diversity to improve network robustness.

Besides PHY techniques, some MAC mechanisms are considered to reduce latency through time and frequency diversity. Pedersen et al. have proposed a punctured scheduling [55] reducing control latency at the MAC layer by prioritising URLLC traffic over eMBB to immediately schedule more time-critical traffic. While the use of dynamic scheduling is advantageous when intermittent traffic is generated to ensure some level of resource efficiency, the associated cost of high latency connection establishment makes this approach inappropriate for time-critical services in URLLC. On the other hand, Semi-Persistence Scheduling (SPS) [56] is proposed to shorten latency by periodically allocating resources to users. The periodicity of resource allocation can be updated according to channel conditions. Thus, resources can be

adequately allocated to each user in a short period of time, reducing spectrum inefficiency. However, the aperiodic traffic behaviour of users reduces the efficiency of resource allocation and can degrade network performance when multi-users are considered. It is therefore necessary to discuss new approaches to the scheduling of resources for aperiodic traffic behaviours, in multi-user scenario and guaranteeing URLLC communications.

2.2.2 Mobile Edge Computing and Core Networks

In Core Network (CN), packet forwarding takes place over a wired medium such as fibre or copper. Thus, signalling, routing and packet processing are mainly considered to design a core network supporting URLLC communications. Due to the huge amount of data and high congestion of packets in the core network, its redesign needs to be discussed. To support low latency communication at Core Network, SDN and NFV are considered as the main candidates when they allow the separation of the control plane and the data plane, increasing the flexibility and scalability of the network and opening a door for latency reduction. Furthermore, with the combination of SDN and NFV, it is viable to bring the core network closer to the end-user to form MEC and thus further reduce E2E latency.

Mobile Edge Computing (MEC) is a well-known concept based on the distribution of computing resources, storage and control services at the edge of the network and close to end users. This enables end users to run time-sensitive and/or mission-critical applications without having to wait for their data to reach the remote server via core network, thereby reducing back-haul and core network delay. Based on a distributed computing system integrating multiple technologies: network virtualisation, cloud computing, SDN, small cells and network slicing, the use of advanced MEC and CN will be more beneficial for 5G networks due to its scalability, resilience, fault tolerance and reduced network congestion.

The MEC main features can be summarised as follows **[57]**:

- On-premises: MEC can run in a standalone mode and provide a mobile user with appropriate services.
- Proximity: MEC servers are located in close proximity to a mobile user.
- Low latency: The computing power of the MEC is sufficient to process applications in real time. This allows us to support latency-critical 5G applications and also reduce the burden of backhaul traffic.
- Location awareness: Since an MEC server is close to a mobile user, it allows us to provide location-based services.
- Network contextual information: MEC is able to use information about radio network conditions and local context information for optimisation of network and application operation. and application operations

2.3 Evaluation of combined mechanisms

In this section, we study how to combine some of mechanisms defined in Figure 2.1, enabling URLLC and which can be a first step for later research using data-driven approach such as Machine Learning (ML) algorithms. In particular, we will focus on numerology (frame design), micro-diversity (MIMO), HARQ (redundancy) and adaptive MCS. We consider an 5G network framework based on new radio module implemented by **[3]** in NS3 network simulator. The core network is based on LTE/EPC model while the RAN takes into account sub-6 GHz communications in the NR architecture. Performance is evaluated in terms of E2E latency, network jitter and RAN reliability.

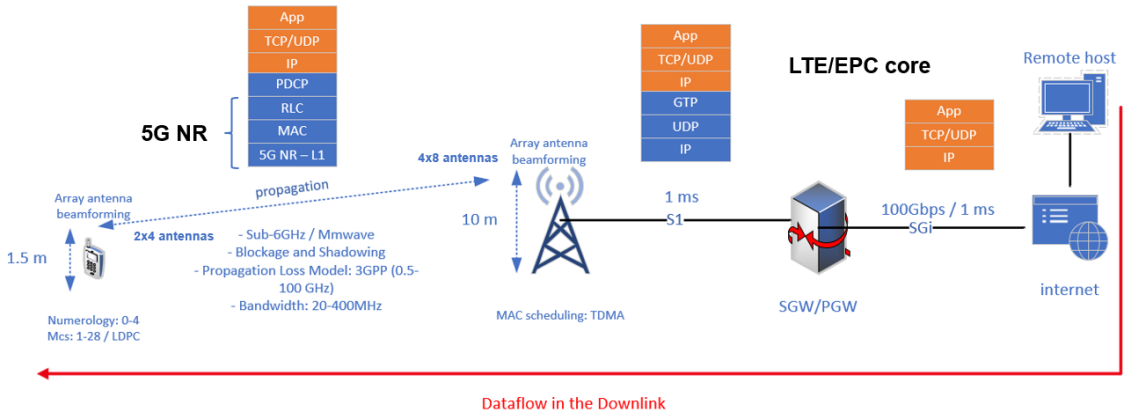

FIGURE 2.2: *Simulation scenario.*

2.3.1 SYSTEM DESCRIPTION

The considered scenario is that of a UE equipped with a (2x4) Planar Linear Array (PLA) antenna communicating with its gNb with a (4x8) PLA antenna via a DL direction and requesting service from a remote server. Spatial diversity is thus initially achieved by a MIMO antenna on the UE and gNb side. The sub-6GHz NR communication characteristics are as follows: 20 MHz bandwidth in the 3.5 GHz band, three different numerology (0, 1 or 2) and LDPC channel coding associated with various MCSs (MCS5, MCS12, MCS17 and MCS25). These MCSs are respectively the four different modulation orders (4-QAM, 16-QAM, 64-QAM, 128-QAM) and refer to four coding rates (0.37, 0.42, 0.43 and 0.8). Numerology and MCS (time, frequency and signal processing diversity) are applied to reduce the delay at the RAN level. To improve reliability, HARQ-IR is applied. The selected propagation model is the indoor factory channel model recommended in [44]. The fast-fading channel, which has a coherence time of 10 ms, is a source of transmission impairment. Another impairment is the dynamic generation of traffic at the application layer (exponential distribution of message size and inter-arrival time with an average of 60 Bytes and 1 ms respectively). Our simulation scenario and the associated parameters are then shown in more details in Figure 2.2 and Table 2.1, respectively.

2.3.2 NUMERICAL RESULTS

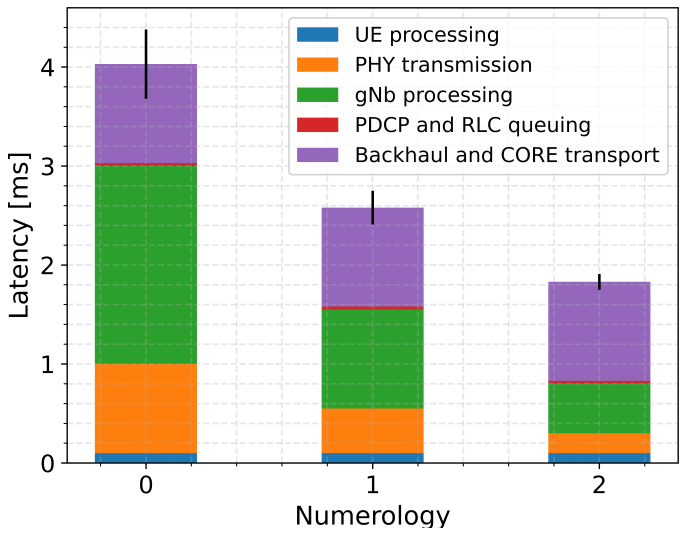

FIGURE 2.3: *Distribution of E2E latency across the network layers*

TABLE 2.1: *Table of the parameters for the simulation*

Simulation parameters	**Values**
Device	1 UE vs 1 BS
Freq	3.5 GHz
Bandwidth	20 MHz
Distance Ue-Bs	120 m
Pathloss Model	Indoor Factory
Numerology	Num0 - Num1 - Num2
MCS scheme	5 - 12 - 17 - 25
Coding Rate	0.37 - 0.42 - 0.43 - 0.8
HARQ	Incremental Redundancy (IR)
Service type	UDP
Message size	60 Bytes
Message inter-arrival time	1ms

Figure 2.3 shows the distribution of the E2E latency through the different layers according to numerology. The impact of the LTE/EPC core network was quantified by the fixed delay of 1 ms for the back-haul and core transport link, respectively. It refers to the time taken for packets to travel through the transport network from a remote host before reaching the RAN network. At RAN level, the packets are first proceeded with PDCP layer before being queued at RLC layer (The details on Appendix B) which the delays are represented by PDCP and RLC queuing, respectively. At MAC layer, every packet is scheduled with a delay of L_{12} = 2 slots which corresponds to the preparation time of gNb before pushing them over the air **[58]**. At PHY layer, a time slot transmission is considered and its delay is measured by PHY transmission. Once UE receives the transmitted message, it takes a constant UE processing delay of 100μs .The E2E latency is mainly due to the PHY and MAC layers where TTI is successfully shortened compared to the frame structure and the scheduler delay. Figure 2.3 also illustrates the jitters caused by the mismatch between message generation at the application layer and resource allocation at the MAC layer.

Figure 2.4 extends the study performed in Figure 2.3 by considering E2E latency and BLock Error Rate (BLER) as a function of numerology and MCS. As expected, lower MCSs (e.g. MCS5) have more robust communications and higher latency while higher MCSs offer higher throughput, lower latency at the cost of reliability. Higher MCS at PHY layer means higher modulation order is set to carry more bit-wise information per OFDM symbol and higher coding rate (Appendix B, Table B.2). Nevertheless, the required SINR at the reception is also more strict to be able to decode the message. At lower MCS (e.g. MCS5) the obtained error rate is small because only low SINR is enough to decode the transmitted messages. When highest MCS (e.g. MCS25) is applied, error rate is exponentially increased and it shows that almost all transmitted messages are failed. In exchange, latency under low MCS is higher because less information is carried in each transmission when compared to higher MCS. This shows the trade-off between latency and reliability at RAN when various MCS are selected.

In order to give a better insight into the distribution of E2E latency over time when retransmissions are involved, Figures 2.5 and 2.6 illustrate the distribution profile and Cumulative Distribution Function (CDF) of E2E latency for MCS 17 with different numerology, respectively. It is evident that the integration

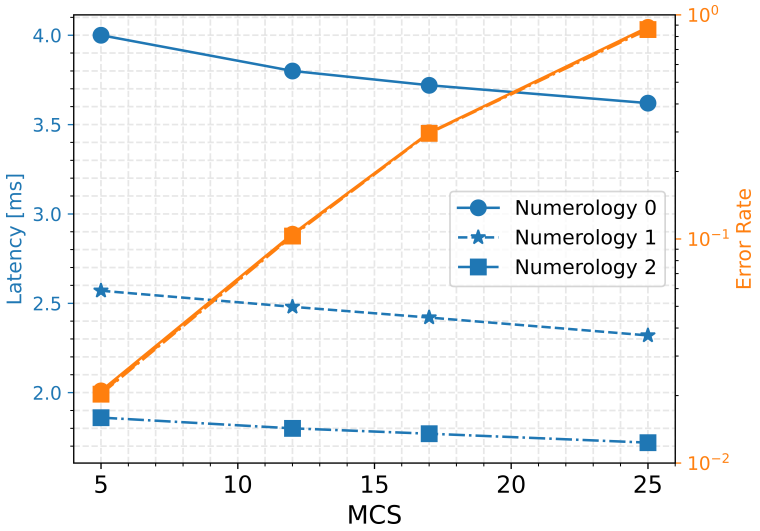

FIGURE 2.4: *E2E Latency and Packet Error Rate*

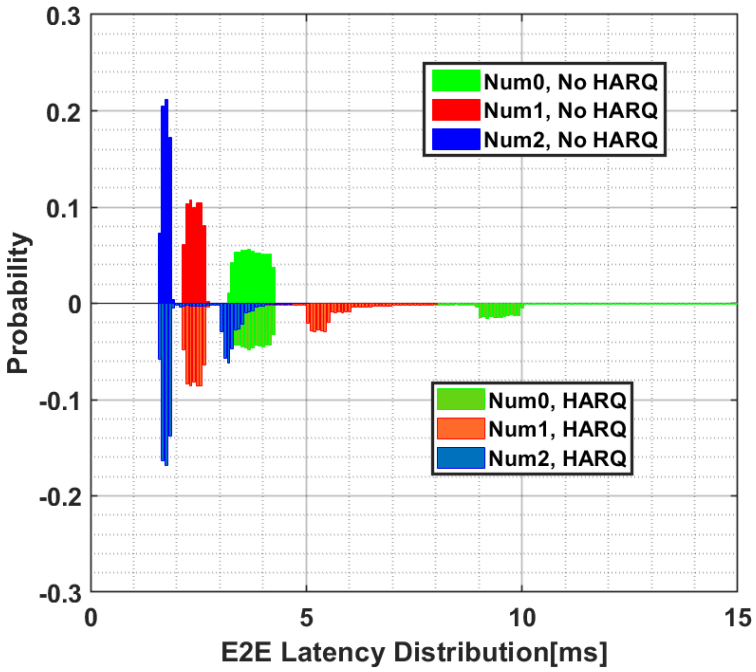

FIGURE 2.5: *E2E latency distribution of MCS17 with or without HARQ*

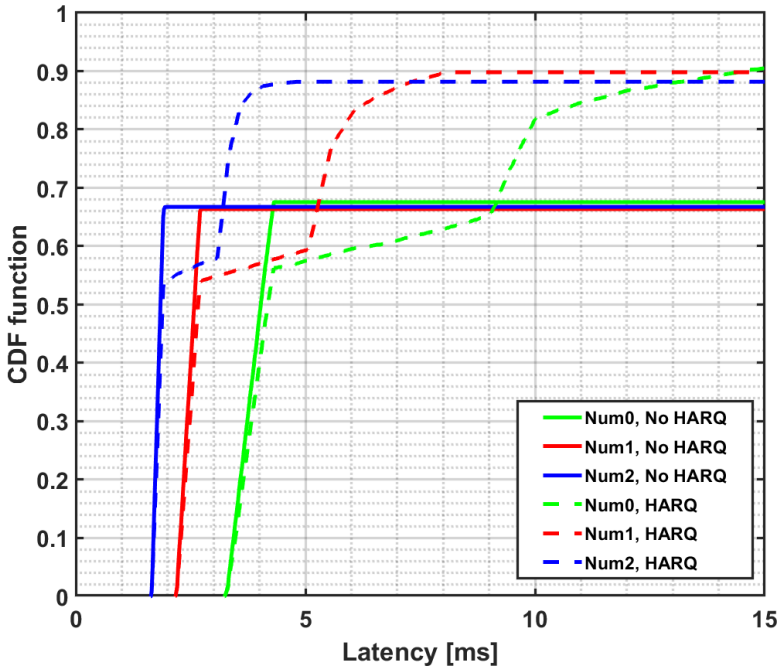

FIGURE 2.6: *CDF of E2E Latency of MCS17 with or without HARQ*

of HARQ and higher numerology allows us to both improve reliability and reduce PHY transmission latency. However, the inevitable generation of groups of adjacent delays (retransmission time diversity) causes the measured delays to diverge into different time windows depending on the maximum number of retransmissions allowed and the numerology applied. As a consequence, the latency is spread out in time and its average latency is shifted towards higher values.

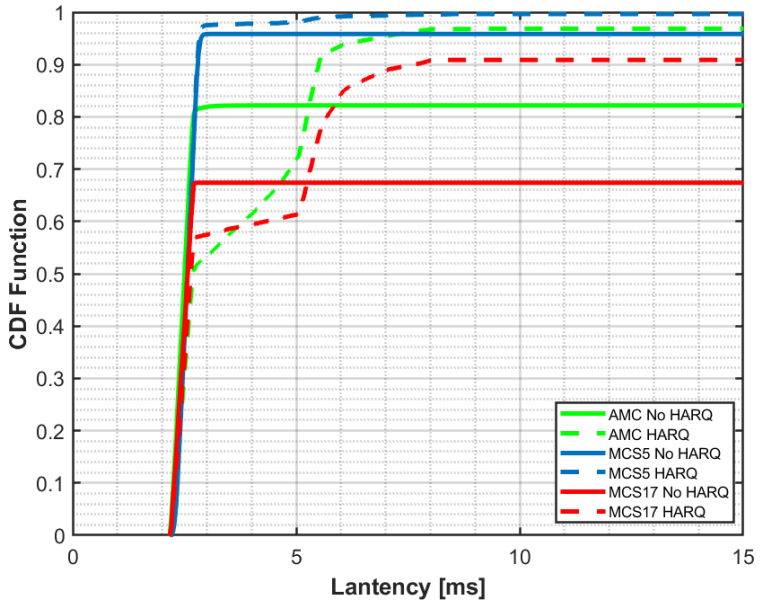

FIGURE 2.7: *CDF of E2E Latency for MCS5, MCS17 and AMC*

Figure 2.7 compares the performance of the AMC mechanism exploiting signal processing diversity with the robust MCS 5 and the high throughput MCS 17 in a fast fading channel. The CDF curves show that the MCS 5 with HARQ outperforms the AMC and achieves a reliability of more than 99% for a latency of less than 5 ms, compared to 81% and 61% respectively for the AMC and MCS 17. This shows

that strict adaptation of the MCS may not always be the best solution to tackle signal degradation in a fast-fading channel and that certain margins are needed to avoid unpredictable events and to bound the performance.

2.4 Network Orchestration

Based on the 5G architecture in [59] and the integration of AI in an unified framework as illustrated in [60], we propose an intelligent framework to control the entire network and guarantee not only URLLC but other heterogeneous services. Our proposed framework serves as a controller (orchestrator) for each network entity (RAN, MEC and CN) to intelligently coordinate optimal solutions in an efficient manner to ensure the target performance. In the scope of our thesis work, our concerns are mainly upon on RAN level orchestration for URLLC communications.

The main objective of AI integration is to make the prediction of the impairments that could occur for a better provisioning of resources and an optimal selection of mechanisms adapting the environment. The proposed framework is able to deal with multiple QoS by efficiently combining several mechanisms simultaneously. Besides, performance monitoring/measurement paves the way for the orchestrator to dynamically adapt network parameters and mechanisms to respond to these changes. Dynamic adaptation can be either reactive or proactive with respect to the current system analysis or long-term data collection for later prediction, respectively. Finally, depending on the network infrastructure's capability, the degree of elasticity can be ensured by making resources (storage, communication, caching, computing) available to different parts of the network.

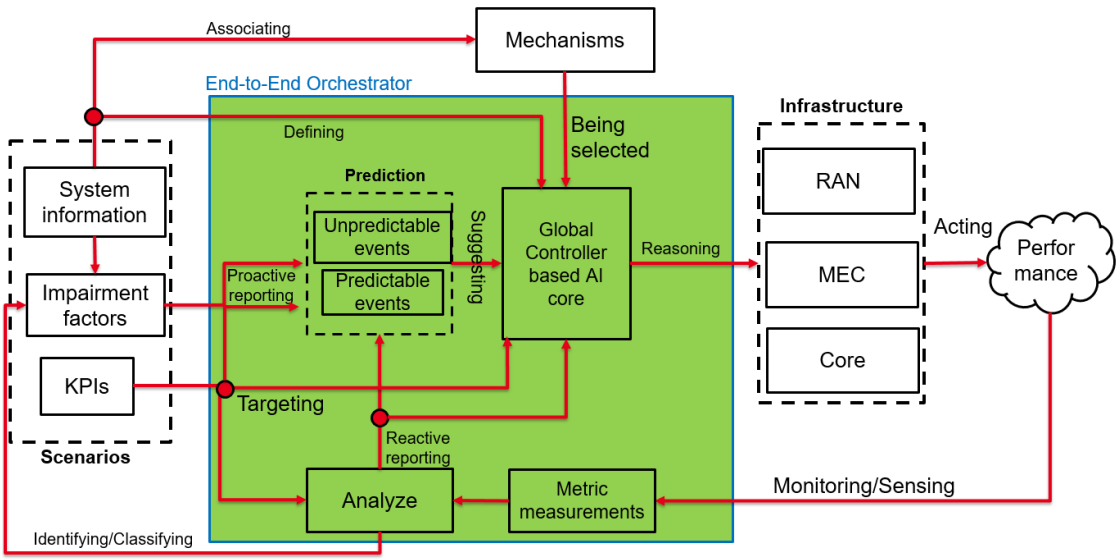

Figure 2.8: *Framework to design an orchestration.*

The orchestrator design proposal is to smartly coordinate UEs, gNbs and network entities. It is designed to find the optimal subset of preferred mechanisms for configuring the RAN, MEC and CN to fulfill the targeted KPIs without wasting resources. The complexity depends on the scenario, the subset of mechanisms to be chosen, the level of abstraction at RAN and core and the selected AI. Figure 2.8 illustrates the orchestrator framework. The orchestrator inputs are system information, KPIs and impairment factors which can be described as follows:

2.4.1 System Information

This block provides user and infrastructure information and allows the appropriate mechanisms to be defined to support the QoS. User-level information defines the capabilities of each user, such as the user's antenna configuration, the supported RATs, battery limitations. On this basis, the orchestrator will coordinate appropriate strategies to adapt to the services required by each user by (de)activating the available mechanisms. Infrastructure information such as computing and storage capacities and network standards, will be considered as a performance constraint that the orchestrator will have to adapt to the heterogeneity of user needs.

2.4.2 Key Performance Indicators (KPIs)

This block presents the network states at different network layers. For example, the E2E latency is composed of many elements in the RAN, the edge and the core network. In particular, the RAN latency corresponds to the transmission time (i.e. to transmit the physical block data in the PHY layer) or the latency for establishing the grant which depends on scheduling scheme at the MAC, the processing time at the UE/gNb and the notification messages at upper layers (RLC, PDCP). The edge latency corresponds to the computation time, queue delay, feedback and forwarding delay, and reconfiguration time in the MEC. Transport latency is the time that transport network needs to transfer the packet to the target direction. Similarly, the E2E reliability, usage, energy efficiency, coverage and complexity could be considered.

2.4.3 Impairment Factors

Several factors can degrade the performance of the different layers of the network. With appropriate information and prediction leveraged by AI, the orchestrator can prevent predictable events from degrading system performance. In case of unpredictable events, the orchestrator can avoid performance degradation and bound the performance. From a communication point of view, reliability impairments are due to the power degradation of the useful signal, uncontrollable interference, resource depletion, protocol, reliability mismatch and equipment failure. The source of perturbation may be due to time-varying channel and channel uncertainty due to mobility or changing environment, interference (e.g., collisions with other users in uncoordinated channel access, coexistence with other systems in the same frequency bands) or imperfect knowledge of the environment (e.g., measurement inaccuracy or obsolete information). Besides, intermittent connectivity, time-varying traffics or a change in the number of mobile users can also cause an increase in RAN latency. From a computing and caching point of view, impairments are related to the unavailability of computing resources due to the congestion or overloading of tasks. Besides, poor coordination between multiple edge servers when users are multi-connected can lead to wasted resources and increased latency. Concerning the use of AI in CN, deviation in accuracy between the training phase and the test phase could also lead to errors.

2.4.4 Orchestrator

Several closed-loop management processes are designed. First, the system information will inform the orchestrator about the behaviours and capabilities of the network in a specific region. Then, it will measure the state of the network and apply the appropriate mechanisms at the different levels of the network to achieve the targeted performance. In the feedback direction, the orchestrator will analyze the performance by comparing measurements and KPIs requirements. In case of a difference between the expected and the actual performance, the orchestrator will adjust the decision and re-orchestrate the network in a reactive way. Metric measurement and analyzer are the key elements that enable close-loop management when it monitors the performance and the network state during operation. The analyzer will also identify and classify impairment factors or network faults into predictable and unpredictable events. With appropriate information and prediction leveraged by ML, the orchestrator can prevent

predictable events degrading system performance and can avoid performance degradation due to unpredictable events by bounding performance.

In this case, the orchestrator will proactively reconfigure the network. The introduction of a new network architecture (Open-RAN (O-RAN) and MEC) brings flexibility, elasticity, efficiency to the network since storage, processing, control, communication are centralized. During communication operation, (un)predictable impairments can occur. It will then trigger the orchestration of mechanisms to adapt to those degradation. Depending on the short or long-term characteristics of the impairments, the orchestrator can decide on various actions at different locations in the network. For example, blockage or fast-fading channel can be overcome by allowing dual connectivity or band switching. However, long-term impairments, such as communication/computing resource depletions, may require not only additional resources at the RAN level but also at the MEC and CN level. Therefore, the global controller needs to distinguish the characteristics of the impairments for better orchestration.

2.5 Decision maker models

In this section, we show two models which can be exploited with our proposed orchestrator at RAN level: *(i)* Exploit latency distribution for decision making and *(ii)* Jitter-aware scheduling in wireless time sensitive network communications. The first model addresses the trade-off between latency, reliability and efficient use of radio resources by taking into account the flexible transformation between reactive and proactive approaches. The second model allows the orchestrator to force latency to a predetermined "window" regardless of the dynamics of the environment and focuses on jitter awareness in addition to latency through " mean" and "standard deviation" control.

2.5.1 Exploit latency distribution for decision making

More advanced technologies/ mechanisms are needed to jointly reduce latency and improve reliability while maintaining an appropriate efficiency. The classical approach is to propose a systematic combination of several mechanisms to stretch the latency below the deadline. This approach is not adapted to dynamic environment. On the other hand, proactive adaptation strategies select mechanisms for the worst case with margin. This implies a high cost of ultra-reliable communications because impairments (worst cases) can be very rare. Reactive adaptation strategies proposes to activate additional resources but these strategies increase considerably the latency and are based on average single-modal latency with jitter. In this section, we propose a decision maker which trades off between Reliability –Latency –Efficiency for URLLC in 5G networks.

Decision maker (orchestrator) defines dynamically one or several thresholds based on latency distribution in order to make a tradeoff between latency, reliability and efficiency. The objectives are to define when to (de)activate more resources/ mechanisms, to make an efficient tradeoff between reactive and proactive approaches and to exploit multi-modal latency distribution as the illustration in Figure 2.9. If the decision is taken too early, the decision maker will be close to a proactive approach: simple cases do not happen yet and the cost is a possible inefficiency usage of resources If the decision is taken too late, it means that the decision maker waited for many cases to happen. It uses a solution close to reactive approach with high efficiency usage at the cost of an increase of latency. The decision maker helps to solve the tail problem of the distribution (rare case and/or hard to serve)

An example of decision making applying for HARQ protocol can be shown as following: Classically, in the literature, they proposed reactive approach for HARQ. The first transmission happens and we wait for an ACK or a NACK in order to know the success or not of the transmission and the necessity to retransmit the packet. The main drawback is the latency because we have to wait the packet transmission and the reception of ACK/NACK before to the retransmission. The advantages are the efficiency because we re-transmit only after the failure notification. The proactive approach is like the multi-layer HARQ

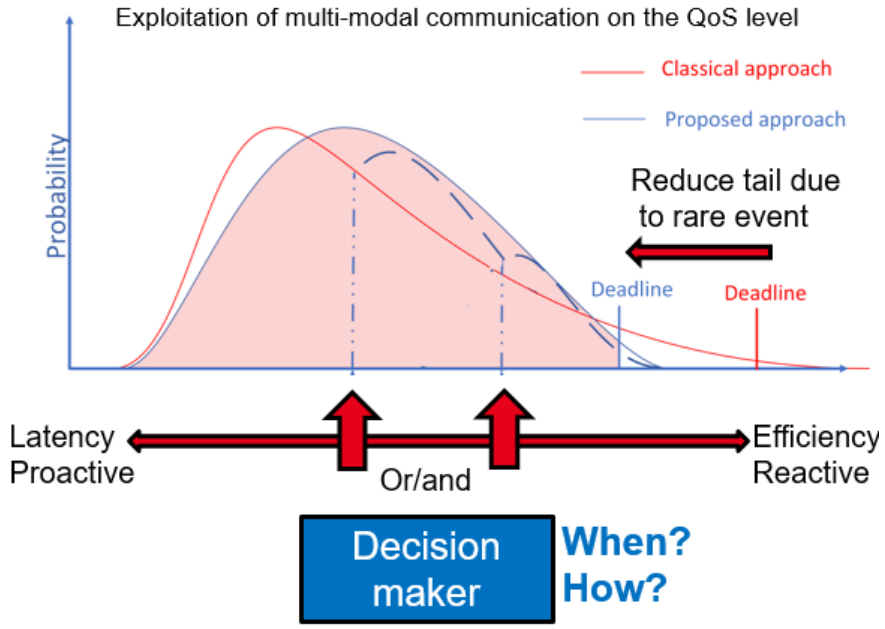

FIGURE 2.9: *Multi-modal communication on the QoS level.*

paper proposed in the literature. It transmits in parallel several packets to reduce the latency but if the first transmission were enough, resources are wasted.

In our approach, we propose to activate the parallel transmission on demand depending on the condition or gradually depending on the latency distribution. In normal condition, we use reactive HARQ and when rare events occur, we activate proactive HARQ. The classification of normal and other conditions is based on the multi-modal distribution of the latency and the level of confidence (integration of the distribution). Our approach makes a trade off between latency and efficiency. We activate additional mechanism after regular event occurs (then with higher latency than proactive but lower than reactive), but we avoid waste of resources (more than reactive but less than proactive) as in Figure 2.10.

Moreover, in reactive or proactive approaches, the input of the orchestrator is the average E2E latency and jitter profile. Then less information is extracted and it is hard to orchestrate several mechanisms. With our proposition, the input of the orchestrator is the statistic distribution of E2E delay profile, then more information is extracted (e.g., multi-cluster distribution) and progressive decision spread over time can be made.

The decision maker will take into account the following information:

- Multi modal Latency distribution instead of statistical distribution
- Diversity scheme (time, frequency, space, signal processing, hardware) Time, frequency and space are classical. Hardware diversity refers to the capability to switch from one technology to another (e.g., antenna selection, multi-RAT, hardware for virtualization). Signal Processing diversity refers to software such as data processing redundancy, channel coding, various modulation orders to obtain a more robust transmission on the noisy channel. It could also include MEC, softwarization and separation of the virtual network function in the new network architecture to bring computing, communication, processing and storage closer to the end-user. The decision maker will apply different and complementary diversity schemes according to the current state of the communications.
- The time remaining and the appropriate mechanisms to react “on the fly”. It will schedule in time the combination of mechanisms according to the time to apply a mechanism, the remaining time,

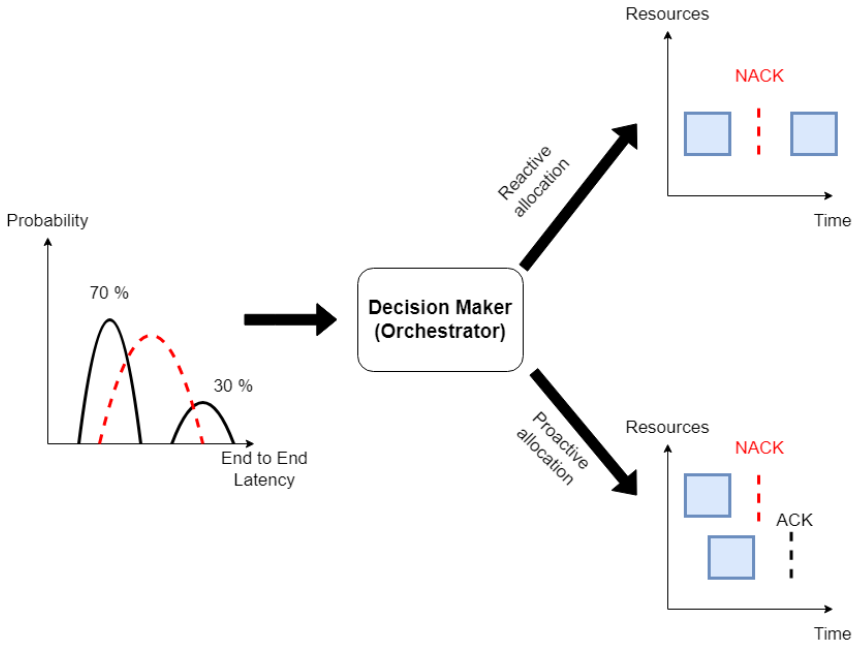

FIGURE 2.10: *Exploitation of HARQ under decision making process.*

the current status of already applied mechanisms in order to make a better latency-reliability-efficiency trade off.

- Classification of the multi-modal latency as a function of the source of impairment with a certain level of confidence.

The advantages of our decision maker is a better latency-reliability-efficiency trade off (it is not necessary the most latency-reliability performance one), which is not limited inside a mechanism but takes into account high level orchestration and it does not need to know the source of impairments because it will learn online how to react and is adapted to dynamic environments. Concerning the multi-step decision, one or several thresholds are selected based on the confidence level to make the trade off between latency, reliability and efficiency.

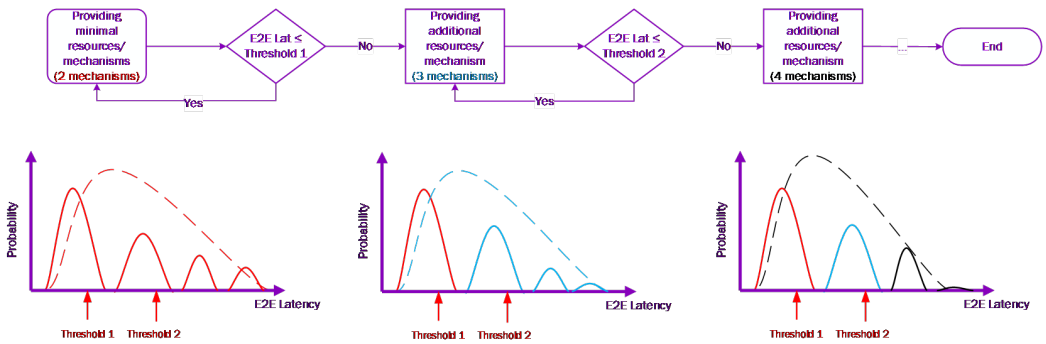

FIGURE 2.11: *Multi step decision making process.*

When the time passed, the activation of additional resources/mechanism is essential to give the robustness to the communication. After several steps, the dynamic allocation of mechanisms/resources reduces the latency. The decision maker will exploit the latency distribution by classifying the multi-cluster distribution according to the level of confidence and intermediate deadlines (threshold).

The classification process with differentiate easy to serve and risky to serve cases. While in the literature, systematic approach is used. In our case, the decision maker with learn and differentiate on

the fly the different cases. It is important to recall that easy to serve cases can be solved with a limited number of resource. These cases happens in general, for the majority of communication and thus it is not necessary to waste more resources. The risky to serve cases are rare and necessitate many resources. By degrading the latency performance with respect to proactive approach by waiting that easy to serve cases happen, then increasing the level of confidence that this risky to serve case happen, the decision maker becomes more efficient and limits the waste of resources. This classification can be done with multi-steps. From case 1 simple case (more efficient) which provides minimal resources and mechanisms to guarantee the communication, to case k risky case (but only when necessary) with the activation of additional resources/mechanisms on demand based on the confidence in latency distribution. (QoS level)

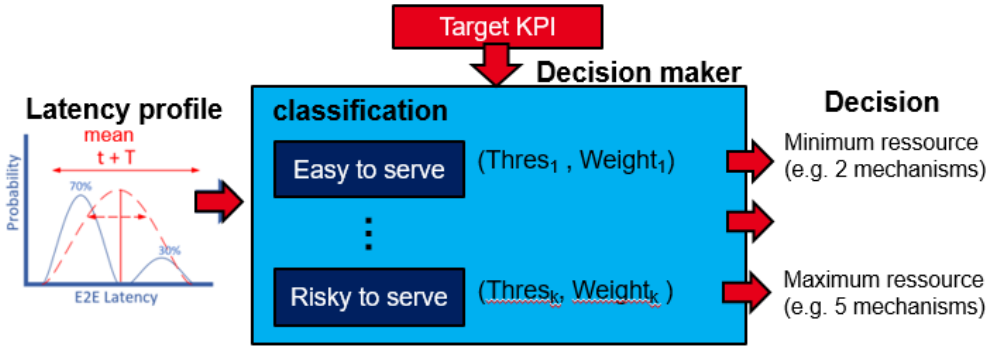

FIGURE 2.12: *Decision making based on latency statistic acquisition.*

Our solution steps develop as follows:

- Step 1: UE defines its target KPIs and inform the Decision maker
- Step 2: Decision maker centralised at gNb defines a initial strategy in perfect condition (without impairments). For example, a relaxed strategy can be initialized at the beginning to effectively consume resources (communication, computation, caching storage, etc.).
- Step 3: Decision maker applies this strategy to the network and UE will experience various performance along the process.
- Step 4: The feedback from UE is essential for decision maker measures the network behaviours and start building statistical information to demonstrate how good/bad of the applied decision (e.g. latency distribution function)
- Step 5: The useful measurements of the UE feedback support decision maker to classify how "difficult" to guarantee the QoS it is (from easy-to-serve to risky-to-serve). Then, the corresponding **threshold** and **weight** are defined to demonstrate the optimal level of proactivity for each use case.
- Step 6: Decision maker will define multi-step exploitation-exploration strategies. Particularly, various diversity-based mechanisms (from 5-dimension diversity) will be pre-determined beforehand and sequentially activated within a constraint delay budget.
- Step 7: Scheduled decisions are applied to users sequentially. In case of success, UE will return feedback to decision maker and any ongoing scheduling decision will be terminated.
- Step 8: UE dynamically adapts its strategy according to status of communication and feedback the message to the decision maker for the future evaluation.
- Step 9: Return to Step 4 (evaluate latency distribution).

2.5.2 Jitter-aware scheduling in wireless time sensitive network communications

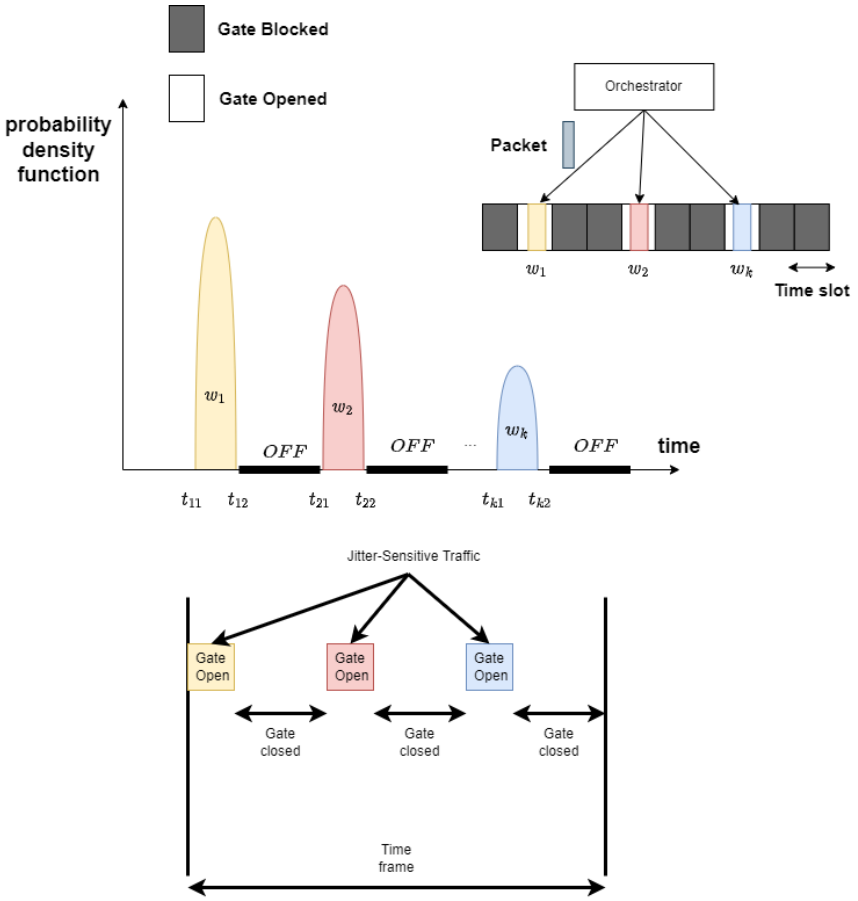

Figure 2.13: *Jitter-aware orchestrator.*

In this work, we propose a jitter-aware orchestration method that forces latency to fall within predetermined windows. The objective is to be independent of the wireless environment (channel condition) and to be able to concentrate the collection of the communication (thus the moment of decision) in a short moment. The system is optimized so that decisions are made within small predefined windows i.e. optimize latency and variation of communication latency (jitter) based on action/control but not on communication conditions. To reduce the latency and the jitter of the communication, whatever the conditions, in a single short window, is very consuming. Thus, we propose a scheduling of the communication resulting in a multi-windows latency with two levels of control:

- "mean" control $\sim$ inter windows orchestration: The “mean” control schedules packet between windows. By defining several windows makes a trade-off between the cost of the action and the decision/goal oriented timeline (joint communication control optimization). The illustration of inter windows orchestration is shown in Figure 2.14.
- "standard deviation (std)" control $\sim$ intra window orchestration. The “std” control activates mechanisms to guarantee the distribution of delay around the mean. (communication optimization for determinism). The objective is to reduce the width of each window in order to provide deterministic condition. Outside these windows, packets are considered as failed.

The objective of the mean control is to define the number of windows K and to classify the windows. When K=1, the optimisation is similar to the one of Time sensitive networks (TSN) protocols. Our work allows to extend to K>1, classify the K windows and schedule each packet in the best fitting windows.

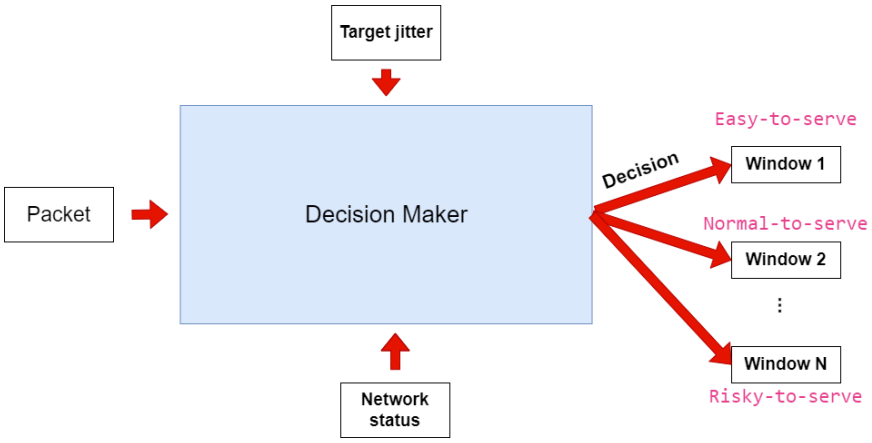

FIGURE 2.14: *Inter-window decision maker.*

An example of windows classification is "Easy-to serve", "Average-to-serve" and "Risky-to-serve". This inter windows orchestration will distribute action intensity for each designed windows. For example, he will define a relaxed action (gate open yellow) for "easy-to-serve" window (i.e. resulting in a yellow latency between t11 and t12 for w1 of packets), average action (gate open red) for "Average-to-serve" (i.e. resulting in a red latency between t21 and t22 for w2 of packets) and aggressive action (gate open blue) for "Risky-to-serve" (i.e. resulting in a blue latency between t31 and t32 for w3 of packets). The reliability of the communication is w1+w2+w3. This example can easily be generalised to N windows.

The objective of the "std" control is to reduce the width of each window in order to provide deterministic condition. This has several advantages: It reduce the uncertainty of when the result will be ready, it eases the coordination or the scheduling of the window of several processes (multi-users or multi-control) at the same time and it facilitates the optimization of the energy consumption of the system. During inter window periods, the system can be turned off or can focus on another task.

The architecture design of jitter-aware orchestration is illustrated in Figure 2.15. Each packet is timestamped and is added to the TX Buffer. The estimator block estimates the status of the communication impairments (e.g., how good of channel state is and how many packets are waiting in the TX buffer). The Controller decides appropriates windows according to the application and drives packets to the transmitter with specified configurations according to the decision maker. The estimator receives a feedback from the receiver concerning the channel condition whereas the controller receives information about the usage of data.

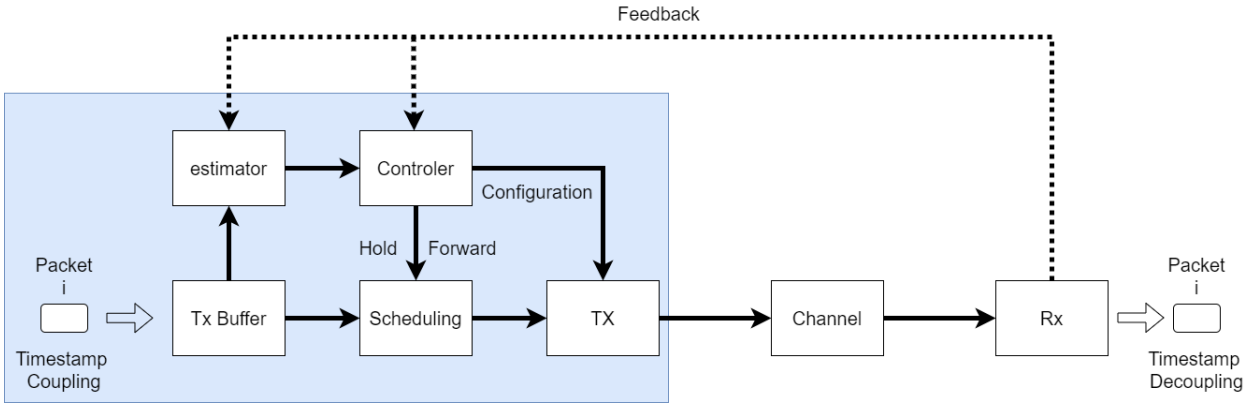

FIGURE 2.15: *Jitter-aware scheduling architecture.*

According to the status of TX buffer (number of packets), the target jitter, the network status and how

the communication will be used by the control system or the application, the decision maker classifies the use-cases and defines a set of actions to fit the communications inside several windows. The difficulty of the realisation is how to choose the best latency windows and how to delay/advance a packet transmission. This include a tradeoff between costs of action, decision intensity and required latency/jitter from the application/control system point of view. The innovation of our solution is to be independent of the communication environment as much as possible. The other solutions are subject to changes in the environment and their optimizations are only from the communication point of view. In our case, we are optimising both communication and control/application system and we take into account when the information communicated will be processed.

The decision maker provides two levels of optimization. The first level concerns the management of several windows (inter-window) or when the information should be provided ("mean" control). The decision maker makes a joint optimization of the communication and the control systems. It defines and classifies several uses cases (from easy to serve to risky to serve) according to the channel conditions (i.e. the latency distribution of the communication). The number of use-cases K can be fixed or defined directly by the decision maker. Once the number of cases/windows defined, the decision maker calculates the optimal positions (in time) of these windows t1...tk. The decision maker can delay or advance a packet transmission in order to constrain the latency inside a window. This is not necessarily the best latency but it is the one that meets the application/control system requirements. Our approach proposes to manage the timing of the communication (latency and jitter) whatever the environment with a deterministic scheduling (multiple opportunities/windows) based on the control system availability/requirements. Each window represents a part of the communication with a certain weight. The earliest window corresponds to the best conditions of communication and the latest window to the worse condition. For each window, a number of mechanisms will be activated as in our previous patent (FR2103542 **[61]**) but now we will also consider the usage of the communication by the control system (goal oriented). A trade off between the cost of the action and the decision/goal oriented timeline (joint communication control optimization) will be made in a multi-step manner. The second level of optimization concerns the management of each window individually (intra-window). The objective is to reduce the size of each window (std control) and thus the jitter of each use case (determinism). It is easier and more efficient to reduce the jitter of each window (meaning use cases with similar conditions) than the global jitter. The decision maker schedules each packet in the best fitting windows. The objective is to reduce the width of each window in order to provide deterministic condition but outside these windows, packets are considered as failed. Our approach has several advantages. It can coordinate or schedule multiple processes or users by aligning their windows as a **rendez-vous**. It can also spread the windows over the time in order to be compliant with the number of computing resources available. Finally, by reducing the sum of the jitters, it can reduce the energy consumption of the system.

The implementation steps are derived as follows:

- Step 1: UE defines its target (communication oriented) KPIs and inform the decision maker
- Step 2: Decision maker defines a strategy to guarantee the communication oriented KPIs (number of mechanism to reach the target KPIs and the defined deadline)
- Step 3: Decision maker applies this strategy to the network and UE will experience various performance along the process.
- Step 4: Decision maker collects measurement and evaluate the performance of the network (latency distribution)
- Step 5: The network and the application/control system define goal oriented KPIs according to the task scheduler (usage of the transmitted data)
- Step 6: Decision maker classifies the cases (from easy to serve to risky to serve) depending on the latency distribution of the communication and the control system jointly (mean control)

- Step 7: Decision maker defined multi-step strategies: progressive decision spread over time can be made in order to reduce the jitter of each window k individually (Std control)
- Step 8: Decision maker applies this strategy to the network and the UE launches it
- Step 9: The decision maker evaluates the performance of the optimization (e.g. cost of the decision, communication loss)
- Step 10: GOTO step 4 (evaluate latency distribution)

The purpose of the decision maker is on the one hand to provide deterministic communications (low jitter) and on the other hand to adapt them to the application (for the one using the transmitted information whatever the conditions). Several variants or alternatives can be made according to:

- The network : when a group decision is needed, each decision maker can synchronise their windows and find a consensus on the same window. For example, when a system must confirm an information from several sensors (e.g. temperature for fire) or simply collect a set of information at the same time, the decision maker can avoid additional latency by limiting the collection to be spread over the time.
- The control system: The decision maker can define the windows according to the availability of the control system or computational resources when multi-processes are involved. It can avoid the queuing of the processing at the receiver.
- The application: The decision maker can schedule the windows of each packet to maximise the probability to receive packets in the order they will be processed or according to their priority. It can take into account sequential processes and define a decision tree with the success probability of each windows of each process.
- The energy: The decision make can schedule the windows that are most appropriate to reduce the sum of the fluctuations, thus the time the decision maker should be on, thus the energy consumption of the system. It can also schedule windows and the probability of success when energy is available (energy harvesting)

2.6 Conclusion

In this chapter, we investigate an orchestrator exploiting multi-level diversity's classification to deal with complex impairments of the environment. The most significant features of our orchestrator are the efficient management of heterogeneous services characterised by competing KPIs, the elastic orchestration of the hardware resources obtained by the concept of O-RAN and SDN/NFV, and the dynamic (de)activation of mechanisms depending on the state of the network with the support of ML algorithms. In order to evaluate the performance of the combined mechanisms on the E2E network, using NS3 network simulator, we have exploited several mechanisms combining signal processing diversity (LDPC, MCS, AMC), spatial diversity (MIMO) and time diversity (frame design, HARQ) at the RAN level, ensuring URLLC communications. Performance are evaluated in terms of reliability, E2E latency and jitter for dynamic traffic and fast-fading channel in indoor office scenario. The results provide insights into the behaviour of different combinations with respect to URLLC performance and are a first step for further research using ML. For example, the mismatch between message generation at the application layer and resource allocation at the MAC layer or HARQ re-transmission can cause jitter and delay clustering and should be avoided for deterministic communications. Moreover, MCS adaptation offers a good trade-off between throughput, latency and reliability but may not be suitable for unpredictable events such as fast fading channel. In this case, some margins on the budget link may be more relevant.

Besides, we propose two decision maker models: *(i)* mechanism scheduling and *(ii)* jitter-aware orchestration. The former helps us exploit a better compromise between network latency, communication

reliability and spectrum efficiency. The latter provides a framework in which latency/jitter of communication is much dependent on our control system rather than uncontrollable environment.

The technical content of this chapter is based on our journal paper [J1], the patents [P1] and [P2]:

[J1] M. Maman, E. Calvanese Strinati, **L. N. Dinh**, et al., *"Beyond Private 5G Networks: Applications, Architectures, Operator Models and Technological Enablers,"* EURASIP Journal on Wireless Communications and Networking, doi: 10.21203/rs.3.rs-430193/v1.

[P1] M. Maman, **L. N. Dinh**, E. Calvanese Strinati, *"Method to exploit latency distribution for early decision making,"* filed patent, FR2103542.

[P2] **L. N. Dinh**, M. Maman, E. Calvanese Strinati, *"Methods and apparatus for jitter-aware scheduling in wireless Time Sensitive Network communications,"* filed patent, .

CHAPTER 3

Ensuring Latency, Reliability and Effective Resource Allocation for URLLC

CHAPTER CONTENTS

3.1 Introduction

3.1.1 Motivations

Supporting Ultra Reliable and Low Latency Communications (URLLC) for 5G wireless networks and beyond is of capital importance for the deployments of novel applications such as self-driving cars and smart factories. Previous chapter deals with cross-layer orchestration of multi-diversity mechanisms to enable End-to-End (E2E) URLLC communications. In this chapter, we exploit the dynamic decision maker architecture for resource allocation problems at Radio Access Network (RAN) in guaranteeing low latency, high reliability and high resource efficiency (spectral efficiency). Besides latency and reliability, which are QoS required by each UE, resource efficiency is also considered as another key performance metric for URLLC **[18]**. Given the requirements for delay and reliability, two approaches can activate available resources for the communication. On the one hand, reactive strategies activate additional resources on demand, which allows for efficient resource utilization, but significantly increases latency, as the demand for additional resources is not instantaneous. On the other hand, the proactive approach is proposed to systematically apply additional resources to stretch the latency below the deadline and it usually activates for the worst case with a margin. As a result, this approach implies a high cost in terms of resource utilization, especially when worst-case impairments are very rare. Our goal is then to define the level of proactivity to apply, to cope with various scenarios, at the scheduling level, taking into account the trade-off between reliability, latency and resource efficiency. This study will help in the design of early decision maker, as patented in **[61]**, dynamically adapting reactive-proactive modes required in dynamic scenario. In order to highlight the importance of the above-mentioned trade-off for improving URLLC, we investigate the direct application of reactive and proactive strategies to the well-known HARQ protocol.

Among the promising enablers for URLLC communication at RAN level, we believe that the Hybrid Automatic Repeat reQuest (HARQ) protocol is the bottleneck that needs to be optimised to meet stringent latency requirements. In the HARQ protocol, we can clearly see that latency is traded off against reliability targets and resource efficiency. In particular, any retransmission following the HARQ protocol will be terminated until the sender receives a Acknowledgement (ACK) response or until it reaches a predefined maximum number of retransmissions. If more retransmissions are required in a highly dynamic channel, the reactive allocation of radio resources for retransmissions will result in a higher communication delay. On the other hand, if the channel is in good condition, then the proactive allocation of abundant resources will be a waste of resources. This requires the adaptation of Hybrid Automatic Repeat reQuest (HARQ) strategies at the scheduling level in a dynamic scenario (i.e. traffic behaviours and rapidly changing channel).

In this chapter, we attempt to fill this gap by exploring the parallelisation of HARQ procedure with a resource scheduling optimization algorithm without sacrificing the efficiency of resource allocation. Relying on Lyapunov's optimizations for two-queue state system management at Radio Link Control (RLC) layer and Medium Access Control (MAC) layer, we design an optimization framework in which RAN latency, reliability and resource efficiency are taken into account. Afterwards, we implement in our 5G OpenAirInterface (OAI) testbed the algorithm introduced in and its evolution, in order to bring a proof of their feasibility under real time restrictions, and evaluate their performance in experimentation.

3.1.2 Related works

Under the strict requirements of URLLC applications, the retransmission regime in classic HARQ is inadequate. Particularly, in the usual reactive HARQ scheme, retransmission (RTX) is triggered as soon as the sender receives the negative acknowledgement (NACK) from the receiver, thus the latency requirement is no longer met if many RTXs are required. In order to tackle with the problem of long RTTs for mission critical applications, K-repetition scheme **[62]** and a proactive scheme with early termination

[63] have been proposed, allowing for a number of redundant retransmissions upon receipt of the acknowledgement by the sender. By doing so, one can opportunistically decode the packet at the receiver in a shorter time at the expense of inefficient resource usage [64].

In order to cope with the over-estimation of packet repetition which causes inefficient resource usage, Le et al. [65] proposed a reserved resource scheme to guarantee that each HARQ process can perform an adequate number of repetitions under a configured period. However, in a time-varying radio channel, the lack of an adaptive packet RTX strategy evidently degrades the system performance since the number of RTX is applied imprecisely and both resource utilization and latency are negatively impacted. With regards to the adaptation strategies in HARQ, in the current literature, it can be achieved in several ways such as adapting the modulation and coding scheme [66], the transmission power [67] and the maximum number of RTXs [68]. By optimally tuning these parameters under dynamic channel conditions, the performance of adaptive HARQ can be substantially improved. However, the adaptation of HARQ strategies at the scheduling level in dynamic scenarios is limited in these research.

Considering the benefits of proactive HARQ and the adaptation strategy to dynamic channel conditions, in this work, we proposed an adaptation control algorithm to improve proactive HARQ. In addition, we evaluate our design in an intermittent traffic scenario that is close to the use cases of URLLC in 5G. To the best of our knowledge, there is still limited amount of research that considers such dynamics in the results. In [69], they proposed a Closed-Loop ARQ protocol that dynamically re-allocates the remaining resource between UL and DL slots upon the result of last uplink transmission. In contrast to the current state-of-the-art, our proposition will robustly select the number of RTXs sufficient to maintain a good level of resource efficiency and ensure a shorter packet delivery time. To design such an optimal control action, knowledge of stochastic processes such as traffic behaviour and instantaneous channel quality is required. However the acquisition of such information is usually difficult and their behaviours are unpredictable, which is an obstacle to design a feasible optimization algorithm. Fortunately, the framework of so-called Lyapunov stochastic optimization does not require the prior probabilities associated with these processes [70]. Furthermore, given the compatibility of the time-slot system in the 5G New Radio (NR), the Lyapunov optimization framework becomes a promising candidate to deal with our problem.

3.1.3 Contributions

In this chapter, we deal with resource allocation problem in guaranteeing RAN latency, reliability and resource efficiency. The developments of this work include 3 phases as follows:

- In the first phase, we exploit the proactive resource allocation in system level simulation to see how better latency is traded-off by reduced resource efficiency at RAN level.
  - ☐ We consider RAN performance by developing a system-level simulator based on NS3 [3] applying to the 5G New Radio (NR). This simulator handles several HARQ processes and measures the latency between the transmitter Radio Link Control (RLC) layer and the receiver RLC layer assuming that the transmission buffer size is infinite. We therefore consider both the queuing latency at the scheduler (due to reactive/proactive approaches) and (re)transmission latency (i.e. PHY/MAC).
  - ☐ We highlight the crucial importance of proactive HARQ adaptation for enhancing the reliability, latency and resource efficiency tradeoff in dynamic scenarios. We consider several levels of proactivity, different traffic rates and channel conditions.
- In the second phase, we proposed two different algorithms for efficient and proactive radio resource allocation to optimise the joint objective between low latency and high resource efficiency under the constraints of the long-term and short-term reliability objective, respectively.
  - ☐ We propose an adaptation control algorithm to improve proactive resource allocation without

scarifying resource efficiency in dynamic scenario. An optimal level of proactivity should be dynamically chosen instead of a static one.

- ☐ We present 2 versions of algorithm branches, in which long term reliability is associated with the evolution of the *risk* in *virtual queue* (adaptive allocation) and short-term reliability is constraint within the limited decisions and is suitable for time-critical messages (adaptive, reliability-aware).
- ☐ By means of System Level Simulation (SLS), we evaluate our design in an intermittent traffic scenario that is close to the use cases of URLLC in 5G.

- In the last phase, we evaluate our algorithms in a real-time hardware experimentation based on OpenAirInterface (OAI).
  - ☐ We show how higher proactivity level is an enabler for achieving lower latency communication in OAI experimentation.
  - ☐ We examine Lyapunov optimization for adaptive allocation and adaptive-reliability aware allocation schemes to see their potential gains in real-time hardware constraints.
  - ☐ We validate the deployment of proposed algorithms with the 5G-compliant hardware.

The technical content of this chapter is based on our conferences papers [C1], [C2] and [C3]:

[C1] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Proactive Resource Scheduling for 5G and Beyond Ultra-Reliable Low Latency Communications,"* IEEE 95th Vehicular Technology Conference: (VTC2022-Spring, doi: 10.1109/VTC2022-Spring54318.2022.9860872.

[C2] **L. N. Dinh**, I. Labriji, M. Maman, and E. Calvanese Strinati, *"Toward URLLC with Proactive HARQ Adaptation,"* in 2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), pp. 220–225. doi: 10.1109/EuCNC/6GSummit54941.2022.9815615.

[C3] **L. N. Dinh**, R. Bertolini, M. Maman, *"Dynamic Resource Scheduling Optimization for Ultra-Reliable Low Latency Communications: From Simulation to Experimentation,"* in 2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), sept. 2022, p. 1026-1031. doi: 10.1109/PIMRC54779.2022.9977893.

3.2 Proactive resource allocation for URLLC

Effective resource use in Ultra Reliable and Low Latency Communications (URLLC) is one of the main challenges for 5G and beyond systems. In this work, we propose a novel scheduling methodology (combining reactive and proactive resource allocation strategies) specifically devised for URLLC services. Our ultimate objective is to characterise the level of proactivity required to cope with various scenarios. Specifically, we propose to operate at the scheduling level, addressing the trade-off between reliability, latency and resource efficiency. We then offer an evaluation of the proposed methodology in the case of the well-known Hybrid Automatic Repeat reQuest (HARQ) protocol in which the proactive strategy allows a number of parallel retransmissions instead of the "send-wait-react" mode. To this end, we propose some deviations from the HARQ procedure and benchmark the performance in terms of latency, reliability outage and resource efficiency as a function of the level of proactivity. Afterwards, we highlight the critical importance of proactive adaptation in dynamic scenarios (i.e. with changing traffic rates and

channel conditions).

The results of this study pave the way for the design of an online decision maker framework enabling to dynamically choose the appropriate level of proactivity based on the network state which is derived in section 3.3.

3.2.1 System model

Our system consists of 1 gNb and 1 UE. It is assumed that a DL traffic is generated by a remote host located near the gNb with an arrival rate of λ and a packet size of L Bytes. Given the physical slot duration t_s, the amount of traffic that fills the buffer at each slot n is $A(n) = \lambda.L.t_s$.

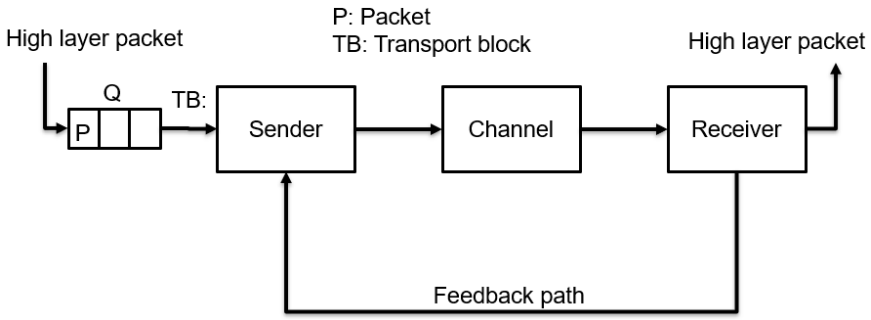

Figure 3.1: *System model for proactive resource allocation*

In order to improve the diversity order of the communication, $U_{tx} = M_{BS} \times N_{BS}$ antennas and $S_{rx} = M_{UE} \times N_{UE}$ antennas, which are modelled as Planar Linear Array (PLA), are installed at the gNb and UE, respectively. In this study, the position between the gNb and the UE is fixed, so the shadowing effect is negligible. Concerning the fast fading model, the sub-channel between the gNb antenna u and the UE antenna s is modelled by channel impulse response H_n. Then, the MIMO channel matrix at transmission slot n e.g., H_n is formed by the superposition of single sub-channels $H_{u,s}(t,\tau)$ which were described in [44]. In this work, we also consider the scenario where the channel fading is too fast to apply instantaneous channel quality feedback to the transmitter, resulting in the transmitter has little knowledge of the channel statistics or only having the outdated delay versions of the feedback. The details on the comprehensive spatial channel model as well as the configuration of antennas in both gNb and UE will be given in Appendix C.

In our study, HARQ Incremental Redundancy (HARQ-IR) is assumed in which each RTX contains different coded bits than the previous RTX. The quality of data transmission over different RB is modelled using Exponential Effective SNR Mapping (EESM) method where a single effective signal-to-noise ratio γ_{eff} is combined from individual SINR received from individual RB, i.e. γ_ω. In HARQ-IR, the effective SINR, i.e. ${\gamma_{eff}}^r$ after r RTXs is derived as follows [71]:

$$ {\gamma_{eff}}^r = -\beta \times \log\left(\frac{1}{|\Omega|} \times \sum_{\omega \in \Omega} e^{-\frac{\gamma_\omega + {\gamma_{eff}}^{r-1}}{\beta}}\right) \tag{3.1} $$

where ${\gamma_{eff}}^{r-1}$ is the effective SINR after the previous (i.e., $r-1$) retransmissions, γ_ω is the SINR experienced by the ω-th RB in the r-th RTX, and $\omega \in \Omega$ is the set of RBs. The value of β depends on the MCS selection and it represents the different effects of MCS on modelling γ_{eff}. Our error model with various values of β are detailed on Appendix B.5.2.

In order to model the error of the signal transmission (i.e., BLER) Link To System (L2S) method is exploited for the system level simulation. In doing so, a SINR-BLER lookup table can be used for the calculation of the TBLER with respect to the MCS selection m associated with Low Density Parity Check Code (LDPC) coding that corresponds to multiple lifting sizes following the 3GPP specifications in [72].

Then, the communication between the gNb and the UE is done on a slot basis, where the slot duration t_s is pre-configured by selecting the numerology (Num). The TDMA-based scheduler is capable of reserving a maximum of 12 OFDM data symbols in a slot, depending on the state of the transmission buffer queue. Based on the number of OFDM symbols allocated to users in a single slot ($1 \leq n_s \leq 12$), as well as the choice of MCS m, the numerology Num, the bandwidth BW, the TBS is defined according to Equation B.1

In terms of MAC procedure, a preparation delay of L_{12} slots is configured at the scheduler to model the time needed for gNb to put the data over-the-air. At the receiver, the packet is considered corrupted if the TBLER is greater than a reliability constraint given by a target error ϵ_t i.e, the actual ${\gamma_{eff}}^r$ received after RTX r is less than a target γ_t. In this case, the NACK will be returned to the sender after a delay of K_1 slots. Basically, depending on the channel statistics and the selected MCS m, the probability of a packet being successfully received after r RTXs is:

$$P_s(m,r) = \mathbb{P}({\gamma_{eff}}^r \geq \gamma_t) \tag{3.2}$$

where γ_t, which is translated from ϵ_t according to EESM method, is the target SINR to ensure that the threshold ϵ_t is not be violated. Concerning the length of transmission buffer queue, the accumulated data in the transmission buffer queue until time slot $n+1$ is determined as follows:

$$Q(n+1) = Q(n) + A(n) - \sum_{i=0}^{n+1} \mathbb{1} \times TBS(m) \tag{3.3}$$

where $Q(n)$ and $A(n)$ are the queue state and the arrival event at time slot n, respectively. For the departure event, a quantity of transport block size $TBS(m)$ will be served to current slot when it received an ACK feedback.

3.2.2 Simulation strategy

In this section, we use system-level simulation to demonstrate the close relationship between E2E latency, reliability and efficiency of both reactive and proactive resource allocation. Here, We defined the resource efficiency $\eta_{R,eff}$ as the ratio of the resources required to accomplish packet transmission to the resources provided. The E2E delay L_{e2e} of the received packets consists of two main parts, namely the queuing delay at the transmission buffer L_q and the over-the-air delay L_a which includes the retransmission process (i.e., L_{e2e}=L_q + L_a).

Obviously, the trade-off between E2E latency and resource usage can be studied by applying the R-parallel allocation scheme. In this regards, R-parallel allocation scheme refers to the fact that R resources are proactively allocated in R consecutive slots for the retransmission of corrupted TB. By opportunistically triggering retransmissions before their feedback, the corrupted packet can be recovered faster than the classical approach i.e, the over-the-air latency L_a of the transmitted packet is shortened. On the other hand, the protocol has to reserve unnecessary slots to re-transmit the already decoded message, which results in low resource utilization.

With respect to the queue stability condition, proactive allocation is feasible under certain conditions related to the arrival/departure rate of packets. Particularly, departure rate (D) must be greater than or equal to the arrival rate (A) to avoid buffer overflow. Assuming that out of N time slots, there are N_1 slot(s) ($N_1 \leq N$) that do not have an ongoing NACK feedback, the queue stability condition can be expressed as follows:

$$\frac{\mathbb{E}(A)}{\mathbb{E}(D)} = \lim_{N\to\infty} \frac{N \times L \times \lambda \times t_s}{N_1 \times TBS} \leq 1 \tag{3.4}$$

Equation 3.4 shows that either N_1 or TBS must be large enough to prevent an adverse queue delay that is greater than the target delay. This equation also shows the trade-off between packet latency and

reliability when running RTX. By triggering HARQ protocol in the current slot, we can improve the communication reliability at the cost of increasing the queue delay caused by the departure process to zero (small N_1).

3.2.3 Reactive and proactive allocation schemes

Figure 3.2 depicts the DL HARQ scheme in which reactive single retransmission ($R=1$) and proactive parallel retransmission ($R=3$) are compared. The details of how the corrupted packet is resent in these 2 different cases are revealed below.

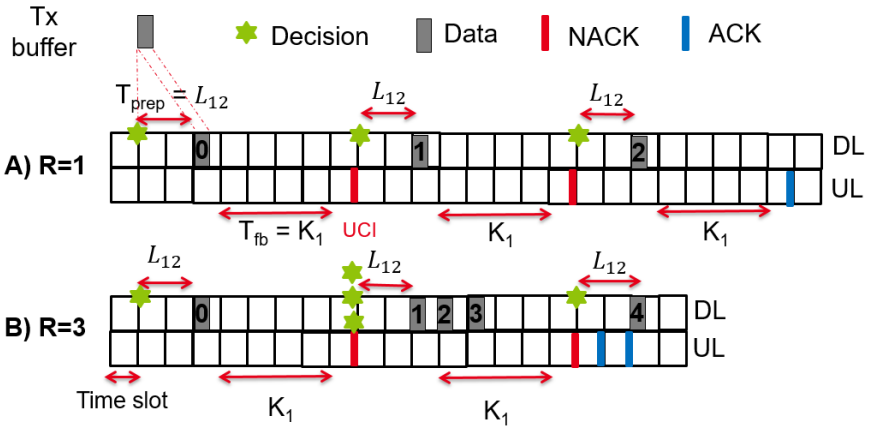

Figure 3.2: *Schema of (A) classical DL HARQ scheme and (B) R=3-parallel RTXs DL HARQ scheme*

The classical allocation, which is based on HARQ procedure, corresponds to a reactive retransmission procedure (i.e. $R=1$). Figure 3.2A depicts DL data flow from an application server, which is close to the gNb, and the UL feedback from the UE side. The procedure of reactive HARQ can be divided into several steps:

- **Step 1:** Data packets are generated periodically at the application layer with a size of L bytes and are queued in the gNb transmission buffer.
- **Step 2:** Once the target UE is determined, the TDMA-based scheduler at gNb takes $T_{prep} = L_{12}$ slot(s) to prepare the TB with a RV equal to 0 in the corresponding buffer and send it over the air. The propagation time between the sender and the receiver is negligible.
- **Step 3:** On the receiver side, the UE processes the received TB to see if it is corrupted or not. If an error occurs, a NACK feedback is encoded together with UCI message and sent back to the gNb after $T_{fb} = K_1$ slot(s) which represents the processing time of the UE. Otherwise, an ACK is sent.
- **Step 4:** The UE feedback is acquired by gNb. A NACK feedback triggers a retransmission of the corrupted TB (RV=1) that was previously recorded with the sender.

From there, steps 2 and 3 can be repeated until the maximum number of allowed RTXs is reached or the packet is successfully received by the receiver (e.g., RV=2 as shown in Figure 3.2A). In contrast, an ACK gives the scheduler the opportunity to serve new TBs from the queue and repeat step 1. Based on the statistic of the probability that the packet is successfully decoded after being re-transmitted r times with a given MCS m e.g $P_s(m,r)$, the average latency $L_{a,r}$ of reactive HARQ is given as follows:

$$L_{a,r} = P_s(m,0) \times L_{12} \times t_s + \sum_{r=1}^{K_{max}} P_s(m,r) \times r \times L_{RTT} \times t_s \tag{3.5}$$

where $L_{RTT} = L_{12} + K_1 + 1$ refers to the RTT of the retransmission, K_{max} is denoted as the maximum number of allowed retransmissions.

Since the receiver requires retransmission of the error packet after receiving the NACK, resource efficiency is always guaranteed, $\eta_{R,eff} = 1$ (i.e., the number of resources requested to accomplish the retransmission is equal to those allocated).

On the other hand, R-Parallel HARQ deviates from the last step of regular reactive HARQ as follows:

- **Step 4':** Once NACK is acquired at the sender, instead of systematically re-transmitting the single corrupted TB, multiple TBs are scheduled consecutively in adjacent transmission slots. The level of redundancy R is determined by the decision maker taking into account that a higher value of R improves latency in exchange for resource efficiency and vice versa.

This approach is illustrated in Figure 3.2B where $R = 3$ refers to the amount of resources the scheduler gives to the sender. In this case, instead of processing a single transmission, the receiver must process a stream of RTXs (RV=1,2,3) to generate their corresponding feedback. Again, these feedback must undergo a constant delay of K_1 slots before reaching the gNb's scheduler. Since the first feedback is a NACK and the others are ACKs, a wasted RTX (RV=4) will be triggered before the scheduler realized that the packet has already been decoded at RV=2. Therefore, RV=3 and RV=4 are considered wasted resources. The resource efficiency can be derived as follows:

$$\eta_{R.eff} = \frac{\sum_{r=1}^{K_{max}} r \times P_s(m,r)}{\min\{\sum_{r=1}^{K_{max}} r \times P_s(m,r) + R, K_{max}\}} \tag{3.6}$$

The underlying over-the-air latency in this HARQ scheme is based on many factors that take into account the redundancy level R chosen by the gNb scheduler and the over-the-air queue caused by the scheduler operation. In general, the sender opportunistically retransmits the corrupted packet R times in R consecutive slots and thus postpones the scheduling of others packets. In this case, the over-the-air latency is given by:

$$\begin{aligned} L_{a,p} &= P_s(m,0) \times L_{12} \times t_s + \\ &\mathbb{E}\left\{\sum_{r=1}^{K_{max}} \left[\left\lfloor \frac{r+2}{R} \right\rfloor \times L_{RTT} + (r+2) \bmod R\right] \times P_s(m,r) \times t_s\right\} \end{aligned} \tag{3.7}$$

where $\lfloor . \rfloor$ is the nearest integer less than or equal operator and mod is the modulo operator. This formulation consists of two parts: the average over-the-air latency of the initial transmission and the average latency of the RTXs. In the example shown in Figure 3.2B where $L_{12} = 2$, $K_1 = 4$ (then $L_{RTT} = 7$) and $R = 3$, the distance in TTIs between the initial transmission (RV=0) and second retransmission (RV=2) is 8 slots.

3.2.4 Numerical evaluation

3.2.4.1 Simulation scenario

Our system consists of 1 gNb and 1 UE at a fixed distance. Then, each packet generated by a remote server has a fixed length of $L = 60$ Bytes. The arrival rates λ are 100 and 800 for low and high rate data generation, respectively. In terms of PHY/MAC configurations, the bandwidth of DL communications is 50 MHz in the sub-6GHz band (i.e. $f_c = 3.5GHz$). The PLA antenna designs are operated on the gNb and UE side, in which $U_{tx} = 8 \times 4$ and $S_{rx} = 4 \times 4$ antenna elements are equipped respectively. Since the gNb scheduler does not know the instantaneous link quality of the communication, a fixed transmission power $P_{tx} = 10$ dBm is distributed equally among the transmitted antenna elements. Besides, a fixed MCS $m = 5$ with a code rate $R_e = 0.3701$ are chosen according to [44]. The numerology Num is set to 1 for the slot configuration. The TDMA-based scheduler takes $L_{12} = 2$ slots to prepare the data stored in the transmission buffer and push it to the air. Once the received payload is processed by the receiver, a delay $K_1 = 4$ slots is required to send the feedback (ACK/NACK) to the transmitter. The TBLER target

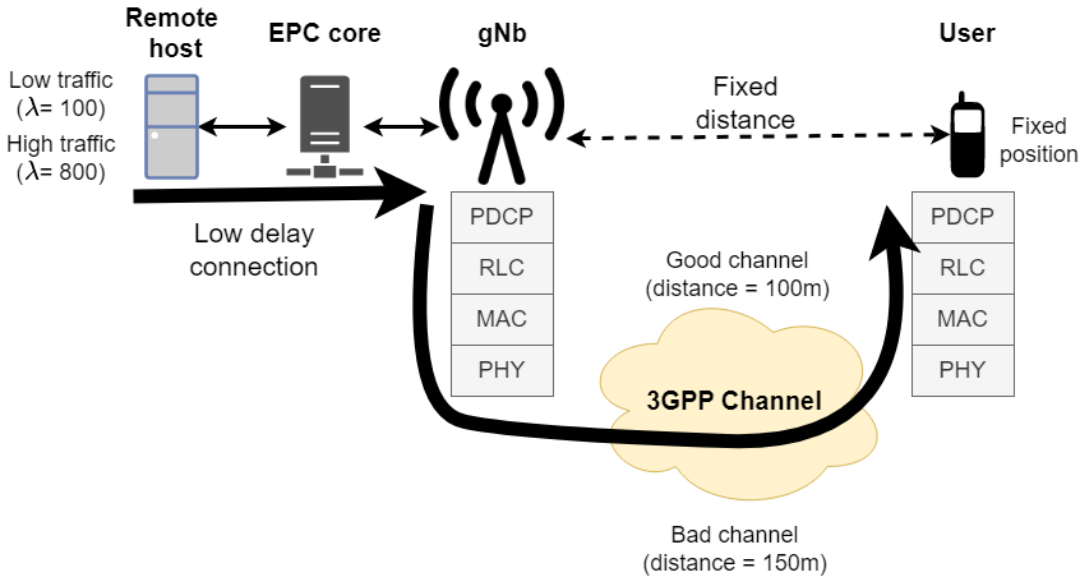

FIGURE 3.3: *Simulation scenario*

TABLE 3.1: *Simulation Parameters*

Parameters	**Values**
(f_c, BW)	(3.5 GHz, 50 MHz)
P_{tx}	10 dBm
(U_{tx}, S_{rx})	$(8 \times 4, 4 \times 4)$
Good vs Bad Channel	d = 100 vs 150 m
$BLER\ \epsilon_t$	10^{-4}
Num	1
(m, R_e)	(5 , 0.7402 , 0.3701)
(L_{12}, K_1)	(2,4) slots
K_{max}	10
L	60 Bytes
Low vs High Traffic	λ = 100 vs 800 packets/s

threshold is set to $\epsilon_t = 10^{-4}$. If this reliability requirement is not met before the maximum number of RTXs (i.e. $K_{max} = 10$) is reached, the packet is considered lost. Finally, the performance is evaluated in a bad/good channel where the UE is placed far/near the gNb at the distance $d = 150m$ and $d = 100m$, respectively. Table 3.1 summarises the parameters that are applied in our simulation.

In this section, we evaluate the performance in terms of latency, reliability outage and resource efficiency, for different traffic rates and channel conditions. The simulation results are averaged over 10 random seeds. Latency is measured from the time the packet arrives in the RLC buffer of the gNb to the time it leaves the RLC buffer at the UE side, assuming that core network latency, signal processing latency and propagation delay are negligible.

3.2.4.2 Performance evaluation

Figure 3.4 shows the Cumulative Distribution Function (CDF) of the latency for different HARQ schemes: classical (i.e. $R = 1$) and proactive (i.e. $R > 1$). In this case, a low application traffic rate is associated with a bad channel condition to perform DL communications between the gNb and the UE. Compared to the reactive approach ($R = 1$), a significant latency gain is achieved when a higher redundancy level R is set. By increasing R, the retransmission process will be accelerated by consecutively scheduling the radio resources for the retransmission and prior to the ACK/NACK reception. As a result, the delivery of the corrupted packets from the source to the destination is both reliably guaranteed and considerably faster than the classical method.

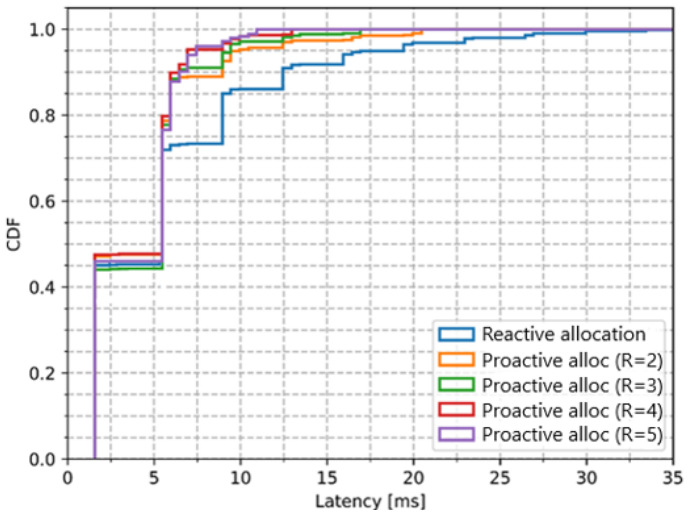

Figure 3.4: *CDF of latency for different redundancy levels R*

To evaluate the impact of channel conditions and arrival traffic rates, Figure 3.5 plots the latency CDF for a redundancy level $R = 3$. Under very good channel condition, a low traffic regime performs best compared to the other scenarios, as shown in particular by the probability of successful transmission on the first communication. When packets are generated by the application at very high rates, and a packet is error at the receiver, the aggregation of queuing effects occurs at the transmit buffer and the radio at the same time, resulting in degraded system performance. In this regard, a poor channel condition or a high traffic rate scenario will severely prolong the overall packet transmission delay.

Figure 3.6 shows the gain in latency and jitter as a function of the redundancy level R for different channel conditions and traffic rates. Latency and jitter are improved compared to the classical HARQ ($R = 1$) regardless of the channel condition and traffic generation behaviour. In low traffic mode, the increased redundancy further improves performance. In contrast, in high traffic mode, performance is optimized at a particular redundancy level before it becomes saturated and degrades. This is because queue aggregation causes buffer latency to increase and moderates the latency gain.

Figure 3.7 shows the outage probability over latency (i.e. the latency required to reach an outage from 0.9 to 0.99999 with different R). It demonstrates how the latency gap is optimized to reach a more critical outage from a more relaxed outage by increasing R. In general, the cost to reach to critical outages in the R-parallel RTX scenarios is lower than the classical method ($R = 1$). The benefit of significantly reduced outage is clearly demonstrated in the low traffic regime where latency outages continue to decrease as R increases. For high traffic, the latency required is less expensive than reactive HARQ ($R = 1$) and starts increasing after an optimal value of R.

To highlight the losses in terms of resource efficiency in exchange for faster response time for different channel conditions and traffic types, the resource efficiency is illustrated in Figure 3.8 as a function of the redundancy level R. We can see that the efficiency of resource usage is lower when a higher redundancy

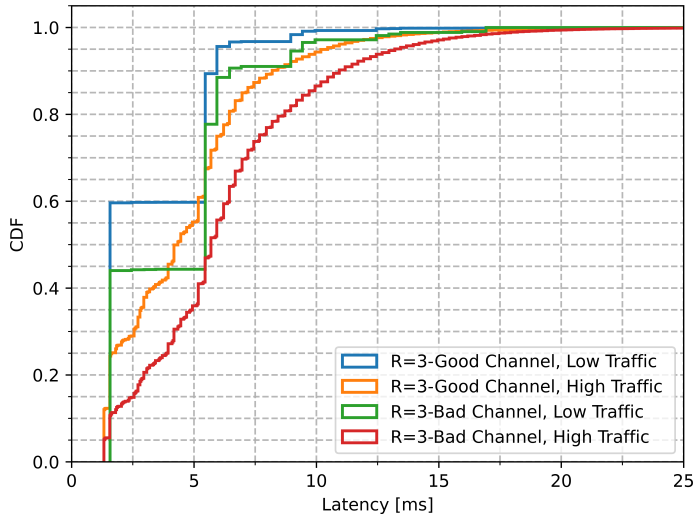

FIGURE 3.5: *CDF of latency for different channel conditions and traffic rates.*

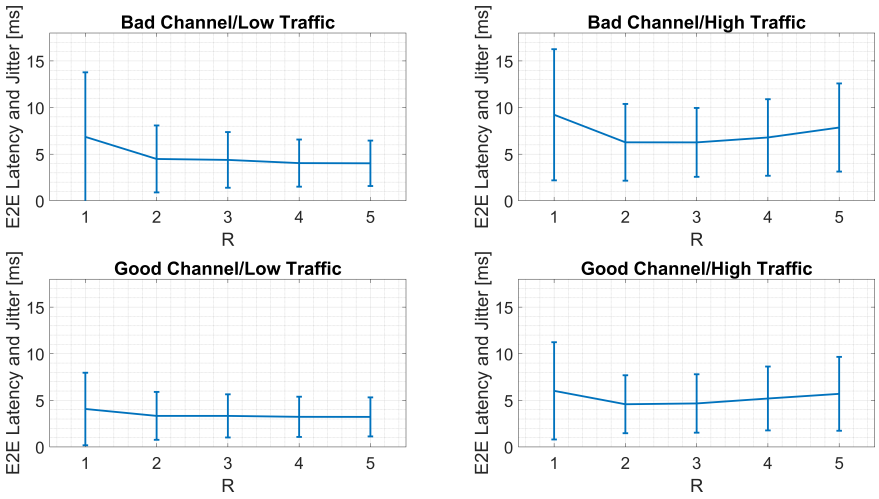

FIGURE 3.6: *Latency as a function of the redundancy levels, channel conditions and traffic source rates*

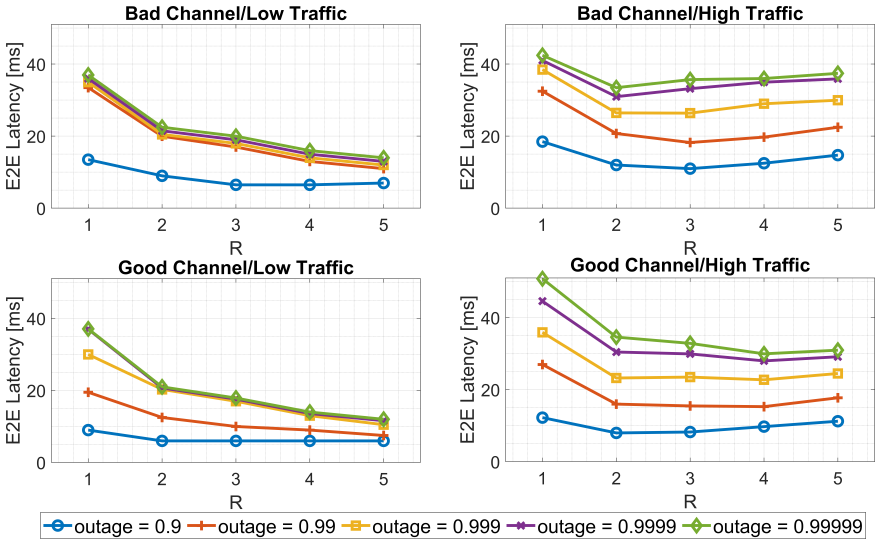

FIGURE 3.7: *Outage probability on latency as a function of the redundancy levels, channel conditions and traffic source rates*

level is configured by the scheduler. At $R = 1$, the scheduler always understands the status of packet transmission before triggering retransmissions, so the resource pools are managed to the maximum regardless the channel conditions and traffic rates. On the other hand, there are significant differences between bad and good channels in terms of resource efficiency at $R > 1$. In particular, when channel condition is significantly good, proactively activating additional RTXs often results in redundancy of allocated resources. Thus, we are more likely to cause the unnecessary retransmission of successfully decoded message, which reduces resource efficiency.

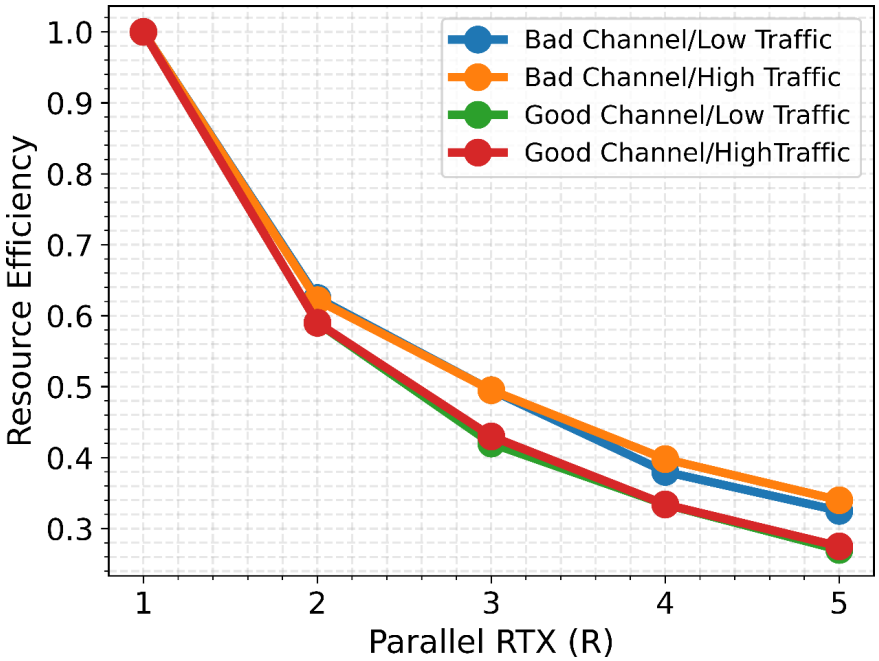

FIGURE 3.8: *Resource Efficiency*

Remarks 3.2.1

In this study, we focused on different resource allocation strategies based on HARQ retransmissions for URLLC in a system level simulator. We proposed some deviations of HARQ procedure and evaluated the tradeoff between reliability, latency and resource efficiency by comparing the performance of reactive HARQ and proactive HARQ as a function of traffic source rates and channel conditions. This studies clarified the correlation between traffic source rate and PHY/MAC configurations to serve the data over the air in a highly dynamic channel and highlighted the aforementioned tradeoff by analyzing the application of different redundancy levels to re-transmit corrupted packets in a classical or R-parallel manner.
Although the latency gain is observed, the exponential decrease resource efficiency in highly dynamic scenario is a burden when higher proactivity is set. In the next section, we are going to propose a solution based on Lyapunov optimization theorem. The objective of this proposition is to improve resource efficiency level in guaranteeing low latency and high reliability level.

3.3 ADAPTIVE, RELIABILITY-AWARE RESOURCE ALLOCATION GUARANTEEING LATENCY, RESOURCE EFFICIENCY AND RELIABILITY

In this work, we propose an adaptive and reliability-aware decision making algorithm to improve proactive resource allocation based on HARQ without sacrificing resource efficiency for better latency

and reliability. Based on Lyapunov stochastic optimization, our adaptation control framework optimally selects the number of proactive retransmissions for intermittent URLLC traffic scenarios under time-varying channel conditions without requiring any prior knowledge associated with this stochastic process. It then better exploits the trade-off between Radio Access Network (RAN) latency, reliability and resource efficiency, which is still limited in its realization on current HARQ designs. We then evaluate the performance of several HARQ strategies and show that our proposal further improves latency over the reactive regime without affecting the resource efficiency such as fixed proactive retransmission while maintaining target reliability.

In terms of reliability, there are two main classes of reliability requirements: *(i)* Ultra-Reliability over long-term and *(ii)* Ultra-Reliability in a short term **[73]**, where the former deals with problems requiring minimal probability rate over long period of above 10 ms (e.g: reliability for a connection to a public cloud in a densely populated area, etc.) and the latter addresses problems with stringent latency requirement of below 10 ms such as vehicles communicating at a crossroad, smart grid, etc. Taking into account their formal definitions, in this section we transform short and long term reliability into cumulative and instantaneous reliability requirements. In particular, the cumulative reliability requirement is related to the mean or average of the reliability over the entire communication being below a certain threshold. On the other hand, the instantaneous reliability requirement is directly associated with the current transmission and we need to guarantee that the current packet transmission will be successfully received with a high probability. Then, we propose two different algorithms which address adaptive allocation and adaptive, reliability aware allocation, respectively.

Our goal is to design an early decision maker, as patented in **[61]**, defining one or more decision moments to dynamically optimize the resource scheduling by adapting reactive-proactive modes to cope with various dynamic scenarios. The efficiency-latency-reliability trade-off is achieved by the timing and intensity of the decisions. The earlier (resp. stronger) the decision is made, the greater the latency gain (resp. reliability gain) at the cost of resource efficiency, and vice versa. To highlight the benefits of early decision making in resource scheduling, Figure 3.9 illustrates the probability density function when the system reacts by setting a series of actions to achieve a latency gain. The clusters represent the latency when a transmission or several RTXs, are required for the receiver to decode the packet. At the end of each cluster, the system knows whether the packet was successfully delivered or not. Figure 3.9 shows how decisions made at different times (e.g., parallel RTXs at actions a_1 and a_2) can reduce latency at the cost of resource efficiency (i.e. after action a_1, packets that only needed one RTX were allocated two RTXs).

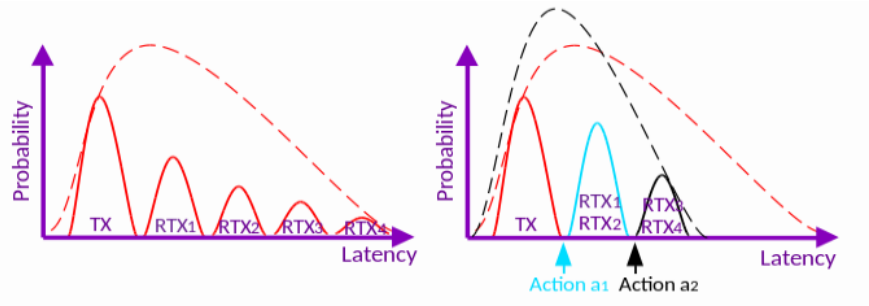

FIGURE 3.9: *Early decision making scheme*

3.3.1 ADAPTIVE RESOURCE ALLOCATION WITH LONG-TERM RELIABILITY REQUIREMENT

3.3.1.1 SYSTEM MODEL

The network contains 1 gNb and 1 UE. During the simulation, the UE node moves around the gNb and varies its distance to it. Then, DL traffic is assumed to be generated by a remote host located near the gNb, in which the variable arrival rate and packet length follow an exponential distribution. Therefore,

the dynamics introduced into the system are an illustration of the unpredictable traffic behaviour and the time-varying channel.

In order to cope with the fluctuating dynamics of the channel and the traffic behaviour, an algorithm is centralised at the MAC layer of gNb, since we are able to observe the evolution of the transmission buffer at the RLC layer and the retransmission queue, which implies the channel characteristics at the PHY layer. Particularly, the application packets are queued in the transmission buffer $Q_1(t)$ of the RLC layer. After completing the scheduler operation at MAC layer, the gNb prepares a TB whose data is extracted from $Q_1(t)$ and sends it over the air. At the same time, the scheduler keeps a copy of this TB and associates it with an identifier to construct an identical HARQ process [74], which will give the receiver information about the processed data. In order to show the state of ongoing HARQ processes that are not yet decoded at the UE side, we defined a $Q_2(t)$ that contains them. Due to the dynamic nature of not only the traffic model but also the channel behaviour, the lengths of $Q_1(t)$ and $Q_2(t)$ are influenced and can be considered as random variables. The state of $Q_1(t)$ and $Q_2(t)$ expresses a two-stage queuing system whose length should be minimised. The details of system model is shown in Figure 3.10

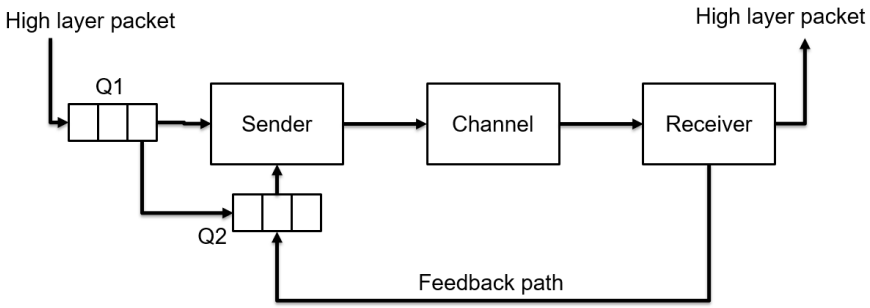

FIGURE 3.10: *System model*

In terms of the channel block, the indoor factory (IF) channel model is used and the details of how the fading channel is modelled are given in section Appendix C. Basically, the Spatial Channel Model (SCM) between sender and receiver depends on large number of parameters, such as the number of antennas at sender/receiver sides, 3D distance between them, statistical characterisations of the embedded scenario (LOS/NLOS probability, shadowing effects, delay spread, etc.), mobility of associated users, etc. In this study, we consider the UE mobility with respect to the gNb, so the shadowing effect plays an important role in the channel status and it has been modelled as in [75].

To better understand the benefits of good decision making for proactive HARQ in improving RAN latency and resource efficiency, Figure 3.11 shows the schemes of *(A)* classical reactive HARQ procedure, *(B)* fixed repetitions of 3 RTXs and *(C)* adaptive redundancy applied to the number of RTXs.

In Figure 3.11A, a delay L_{12} is introduced to demonstrate the TB preparation time from the gNb scheduler to the antenna. Then, a feedback will be encoded together with an UCI message and sent back to the gNb after $T_{fb} = K_1$ slot(s), thus illustrating the processing time at the UE. In NR standard, this processing time reflects a delay between the reception of the UL grant in the DL (PDCCH) and the transmission of the corresponding UL data (PUSCH) [3]. Afterwards, the gNb has the information about the corrupted HARQ process on the UE side and decides to re-transmit the erroneous TB after L_{12} slots. This process continues 5 times until the corrupted TB is successfully decoded by the UE. By doing this, the resource are perfectly utilised, but the latency could be unacceptable for URLLC communications.

Figure 3.11B shows the fixed repetition of corrupted TB with a fixed redundancy level $R = 3$. This means that the first allocation action (i.e., a_1) which will allocation 3 consecutive resources(i.e., $r_{a_1} = 3$), will be reserved for retransmissions. If it fails, the second action (i.e., a_2) with the same redundancy level (i.e., $r_{a_2} = 3$) will be allocated. As shown in Figure 3.11B, latency is significantly improved at the cost of resource efficiency (RTX6 is useless when the TB has been decoded after 5 retransmissions). In this

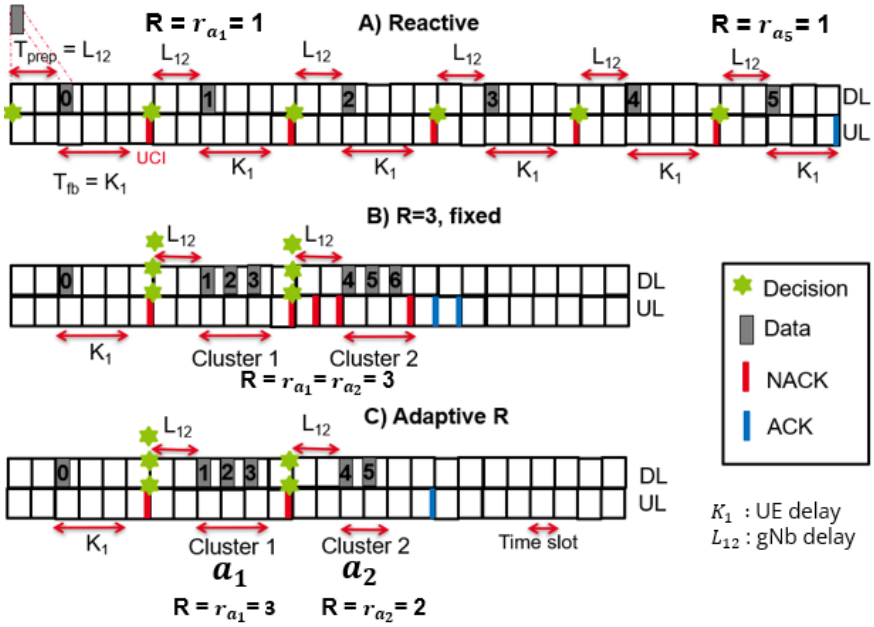

FIGURE 3.11: *Different allocation procedures: (A) classical, (B) fixed R=3 parallel RTXs and (C) adaptive and proactive RTXs.*

allocation scheme, each cluster is defined according to the number of consecutive resource allocations, and cluster sizes are equal.

By adaptively selecting the proactive redundancy level for each action, Figure 3.11C shows better performance in terms of reduced latency and improved resource efficiency. In this design, to reduce the control overhead due to multiple feedbacks to the transmitter, we grouped their feedbacks into a single feedback that represents the current proactive retransmission status. In addition, the cluster size in this scheme is designed to be flexible to improve resource efficiency.

3.3.1.2 PROBLEM FORMULATION

In this work, we limit the number of decisions or actions into a_{max} sequential actions of proactive RTXs. Each action $a_j \in \{a_0 ... a_{max}\}$ can perform r_{a_j} RTXs where a_0 corresponds to the initial action corresponding to a new transmission. The decision maker will take the action which dynamically choose the proactivity level of each action a_j (i.e., r_{a_j}) to reduce both RAN latency and resource waste. With respect to the resources allocated for proactive RTXs for TB (i.e., TB_n with n is the index of TB), the decision maker selects an element-wise positive resource allocation vector $(r_{n,a_0}, r_{n,a_1}, ..., r_{n,a_{max}})$ that satisfies the following condition:

$$1 \leq \sum_j r_{n,a_j} \leq K_{max} \tag{3.8}$$

where K_{max} is the maximum number of RTXs for every transport block (re)transmission at PHY layer. In the case where a TB is not decoded at receiver after K_{max} RTXs, the TB that contains many application packets is considered a loss. Then, $r_{min} \leq r_{n,a_j} \leq r_{max}$ constrains the number of proactive RTXs at action a_j not to exceed a value r_{max}. We can define an objective function f_{obj} that is akin to the average resource allocation to be provided for each TB as follows:

$$f_{obj} = \lim_{N \to \infty} \frac{1}{N} \sum_{n=0}^{N-1} (\sum_j r_{n,a_j}) \tag{3.9}$$

where N is the number of times that decisions are made. The decision of how to optimally select r_{a_j} is based on various factors, such as the current status of the $Q_1(t), Q_2(t)$, the current action a_j and the current aggregated, effective SINR (i.e., γ_{eff}).

The first decision at time t is to select the queue $Q_i(t)\ \forall i \in \{1,2\}$ to proactively (re)transmit the TB over r_{a_j} consecutive time slots. Then, a series of actions $a_j \in \{a_0 ... a_{max}\}$ is made at the corresponding action slot t_a in which the proactive RTXs span r_a time slots. The time interval (t_a, t_{a+1}) forms the $a-th$ time frame denoted by F_a. In order to control at frame F_a, which queue among $Q_i(F)$ will be served, we introduced the control variable α_a where $\alpha_a = 1$ means serving $Q_1(F_a)$ and $\alpha_a = 0$ means serving $Q_2(F_a)$. Knowing that HARQ processes have a higher priority, $\alpha_a = 0$ when $Q_2(t) > 0$. Then, the queuing dynamic will be given as follows:

$$Q_1(F_{a+1}) = \max\{Q_1(F_a) - \alpha_a . TB_a^{r_{a_0}}, 0\} + A_1(F_a) \tag{3.10}$$

$$Q_2(F_{a+1}) = \max\{Q_2(F_a) - (1-\alpha_a).\mathbb{1}_{TB_a^{r_{a_j}}} . TB_a^{r_{a_j}}, 0\} + A_2(F_a) \tag{3.11}$$

where $Q_i(F_{a+1})$ are the backlogs of the queue i at time frame F_{a+1}. The value $A_1(F_a) = \sum_{t=1}^{F_a} A_1(t)$ represents the total amount of high layer packets that accumulated in Q_1 during frame F_a. During this frame, an amount of $TB_a^{r_{a_j}}$ that corresponds to the proactive RTX of in r_{a_j} slots, will be served. The indicator function $\mathbb{1}_{TB_a^{r_{a_j}}}$, in Equation 3.11, is equal to 1 if after the r_{a_j} proactive RTXs, is successful and is 0, otherwise. If the first transmission of $TB_a^{r_{a_0}}$ is a failure, $A_2(F_a) = TB_a$ will be added as a HARQ process to Q_2, otherwise $A_2(F_a) = 0$ as the HARQ process of $TB_a^{r_{a_0}}$ will be removed from Q_2. According to Little's law, the average delay is related to the queue length. Therefore, the first constraint of our problem is to make the average long-term queue length $\overline{Q_1(F_a) + Q_2(F_a)}$ stable (mean rate stable). By definition, a stochastic process Q(t), which corresponds to the queue behaviour, is defined as mean rate stable if **[70]**:

$$\lim_{t\to\infty} \frac{\mathbb{E}\{Q(t)\}}{t} = 0 \tag{3.12}$$

This mean rate stability is necessary to ensure that the time average of the queuing departure process is at least equal to or greater than the time average of the queuing arrival process. The system is therefore stable. In the following, we present an additional constraint that is associated with the average long term risk. The visible risk in our problem is when the decision maker at a_{max} chooses to serve the HARQ process in Q_2 by insufficient allocation of $r_{a_{max}}$ to recover the corrupted TB:

$$\zeta_n = \mathbb{P}[(\gamma_s^{(\sum_{j=0}^{max} r_{a_j})} \leq \gamma_t) \mid \gamma_s^{(\sum_{j=0}^{max-1} r_{a_j})}] \tag{3.13}$$

where $\gamma_s^{(\sum_{j=0}^{max} r_{n,a_j})}$ is the SINR of TB after a_{max} actions of RTX and γ_t is the target SINR. The long-term average risk across all TBs is:

$$\overline{\zeta} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \zeta_n \tag{3.14}$$

The constraint is then to guarantee the expectation of the long-term risk $\overline{\zeta}$ below a predefined value ζ_o. For that purpose, we introduce a *virtual queue* $Z(t)$ which is incremented each time slot t by $\overline{\zeta} - \zeta_o$ each time $\overline{\zeta}$ exceeds ζ_o. The prove of this transformation can be found in Section D.

$$Z(t+1) = \max\{Z(t) + \overline{\zeta} - \zeta_o, 0\} \tag{3.15}$$

Hence, the constraint of satisfying the long-term risk becomes a stability constraint on the average rate of the queue $Z(t)$. The prove of this transformation can be found in Section D.

$$\lim_{t\to\infty} \frac{\mathbb{E}\{Z(t)\}}{t} = 0 \tag{3.16}$$

To summarize, our optimization problem $\mathcal{P}_1$ is to minimize the objective function f_{obj} subject to several constraints:

$$\begin{aligned}
&\text{minimize } f_{obj} && && (\mathcal{P}_1)\\
&\text{s.t. } \lim_{t\to\infty}\frac{\mathbb{E}\{Q_i(t)\}}{t}=0, && \forall i\in\{1,2\} && (\mathcal{C}_{1,2})\\
&\lim_{t\to\infty}\frac{\mathbb{E}\{Z(t)\}}{t}=0, && && (\mathcal{C}_3)\\
&1\le\sum_{j}^{a_{max}} r_{n,a_j}\le K_{max}, && && (\mathcal{C}_4)\\
&r_{min}\le r_{n,a_j}\le K_{max} && && (\mathcal{C}_5)
\end{aligned}$$

3.3.1.3 Proposed solution: adaptive resource allocation

In this section, we propose a decision maker algorithm based on Lyapunov's optimisation tools to solve the optimization problem $\mathcal{P}_1$. To simplify the design of the sequential decision maker, we assume that *(i)* the first decision is the same as the reactive decision (i.e. only one slot is reserved for the TB transmission) and *(ii)* proactive RTXs in the same action share the same HARQ feedback. First, we define the current state in the slot t as $\boldsymbol{\Theta(t)}=(Q_1(t),Q_2(t),Z(t))$ and the Lyapunov function as follows:

$$L(\boldsymbol{\Theta(t)})\triangleq\frac{1}{2}\left[Q_1^2(t)+Q_2^2(t)+Z^2(t)\right] \tag{3.17}$$

Next, we define the one-slot conditional Lyapunov drift $\boldsymbol{\Delta\Theta(t)}$ representing the expected change of the Lyapunov function over a slot as follows:

$$\boldsymbol{\Delta\Theta(t)}=\mathbb{E}\{L(\boldsymbol{\Theta(t+1)})-L(\boldsymbol{\Theta(t)})\mid\boldsymbol{\Theta(t)}\} \tag{3.18}$$

By minimizing both $\boldsymbol{\Delta\Theta(t)}$ and f_{obj}, we can solve the problem $\mathcal{P}_1$ because the queues are stable in terms of average rate and the objective function is minimized . However, according to Neely in [70], there is a *performance-delay* trade-off between these dual objective optimizations that can be parameterized by a constant v. By setting a large positive value to v, the control algorithm will favor minimizing the objective function f_{obj} over the stability of the average rate queues. Our ultimate objective now is to minimize the following *Lyapunov-drift-plus-penalty* function:

$$g(t)=\boldsymbol{\Delta\Theta(t)}+v.\mathbb{E}\{f_{obj}\mid\boldsymbol{\Theta(t)}\} \tag{3.19}$$

Its upper bound, $h(t)$, can be derived as follows for any action, any possible value of $\boldsymbol{\Theta(t)}$ and any parameter $v>0$.:

$$\begin{aligned}
h(t)=&v.\mathbb{E}\{f_{obj}\mid\boldsymbol{\Theta(t)}\}+\mathbb{E}\{Z(t).(\overline{\zeta}-\zeta_o)\mid\boldsymbol{\Theta(t)}\}\\
&+B+\sum_{i=1}^{2}Q_i(t).\mathbb{E}\{A_i(t)-b_i(t)\mid\boldsymbol{\Theta(t)}\}
\end{aligned} \tag{3.20}$$

where B is a constant that satisfies:

$$\begin{aligned}
B\ge&\frac{1}{2}\sum_{i=1}^{2}\mathbb{E}\{A_i^2(t)-b_i^2(t)\mid\boldsymbol{\Theta(t)}\}\\
&+\frac{1}{2}\mathbb{E}\{(\overline{\zeta}-\zeta_o)\mid\boldsymbol{\Theta(t)}\}\\
&-\sum_{i=1}^{2}\mathbb{E}\{A_i(t).\min\{Q_i(t),b_i(t)\}\mid\boldsymbol{\Theta(t)}\}
\end{aligned} \tag{3.21}$$

and $b_i(t)$ is the quantity of TB_n at time slot t that the queue i can process.

$$b_i(t) = \begin{cases} TB_n^{r_{n,1}} & \text{if } i = 1 \\ TB_n^{r_{n,a_j}}.1_{TB_n^{r_{n,a_j}}} & \text{if } i = 2 \\ 0 & \text{otherwise} \end{cases} \tag{3.22}$$

Proof.
By squaring Equation D.1 stressing the fact that $max[p,0]^2 \leq p^2$, we will get the following expression:

$$Q(t+1)^2 \leq (Q_k(t) - b_k(t))^2 + a_k(t)^2 + 2\max[Q_k(t) - b_k(t), 0]a_k(t) \tag{3.23}$$
$$= (Q_k(t) - b_k(t))^2 + a_k(t)^2 + 2\left(Q_k(t) - \tilde{b}_k(t)\right)a_k(t) \tag{3.24}$$

Therefore:

$$\frac{Q_k(t+1)^2 - Q_k(t)^2}{2} \leq \frac{a_k(t)^2 - b_k(t)^2}{2} - \tilde{b}_k(t)a_k(t) + Q_k(t)[a_k(t) - b_k(t)] \tag{3.25}$$

Similarly, we will apply the same proofs for $Z_l(t)$ and we get the following expression:

$$\frac{Z_l(t+1)^2 - Z_l(t)^2}{2} \leq \frac{i_l(t)^2}{2} + Z_l(t)i_l(t) \tag{3.26}$$

Then, we will take the conditional expectations of the above three equations and summing over $k \in \{1, ..., K\}$ and $l \in \{1, ..., L\}$, we will get the following results:

$$\begin{aligned} &\frac{1}{2}\sum_{k=1}^{K}\mathbb{E}\left\{\left(Q_k^2(t+1) - Q_k^2(t)\right)|\boldsymbol{\Theta(t)}\right\} + \frac{1}{2}\sum_{l=1}^{L}\mathbb{E}\left\{\left(Z_l^2(t+1) - Z_l^2(t)\right)|\boldsymbol{\Theta(t)}\right\} \\ &\leq \sum_{k=1}^{K}\mathbb{E}\left\{\left(a_k^2(t) - b_k^2(t)\right)|\boldsymbol{\Theta(t)}\right\} - \sum_{k=1}^{K}\mathbb{E}\{\tilde{b}_k(t)a_k(t)|\boldsymbol{\Theta(t)}\} + \sum_{k=1}^{K}\mathbb{E}\{Q_k(t)[a_k(t) - b_k(t)]|\boldsymbol{\Theta(t)}\} \\ &+ \frac{1}{2}\sum_{l=1}^{L}\mathbb{E}\{i_l(t)^2|\boldsymbol{\Theta(t)}\} + \sum_{l=1}^{L}\mathbb{E}\{Z_l(t)i_l(t)|\boldsymbol{\Theta(t)}\} \end{aligned} \tag{3.27}$$

We see that the right hand side of Equation 3.27 follows the definition of *one-slot conditional Lyapunov drift function* $\Delta(\boldsymbol{\Theta(t)})$. By choosing a positive constant B value that satisfies Equation D.7, we will transform Equation 3.27 as follows:

$$\Delta(\boldsymbol{\Theta(t)}) \leq B + \sum_{k=1}^{K}Q_k(t)\mathbb{E}\{a_k(t) - b_k(t)|\boldsymbol{\Theta(t)}\} + \sum_{l=1}^{L}Z_l(t)\mathbb{E}\{i_l(t)|\boldsymbol{\Theta(t)}\} \tag{3.28}$$

Afterwards, by adding $v.\mathbb{E}\{f_0(t)|\boldsymbol{\Theta(t)}\}$ in both sides of equation above, we will get the upper bound of the Lyapunov drift-plus-penalty-function as Equation 3.20 □

Through the opportunistic minimization framework of a conditional expectation **[70]**, by minimising $h(t)$, the upper bound of the dual objective optimization, we can guarantee that the optimization problem $\mathcal{P}_1$ will be satisfied. Therefore, the design of our algorithm will be based on the control action a at the decisive time t_a and will choose the control action that exhaustively minimises function $h(t)$ as follows:

Since the optimal action is chosen based on the exhaustive search minimizing the designed function $h(t)$ and the action space is bounded from r_{min} to r_{max}, the computation complexity is relatively low and does not lead to an increase in the transmission processing time.

Algorithm 1 Control algorithm

1: Observe time slot t
2: **if** $t \neq t_a$ **then**
3: $t = t + 1$ **else**
end
Observe the concatenated queue $\boldsymbol{\Theta(t)}$
5: Choose optimal action : $a = \arg\min_{a}\{h(t)\}$
6: **end if**
7: Observe the outcomes of taken action
8: Update the queue $\boldsymbol{\Theta(t+1)}$

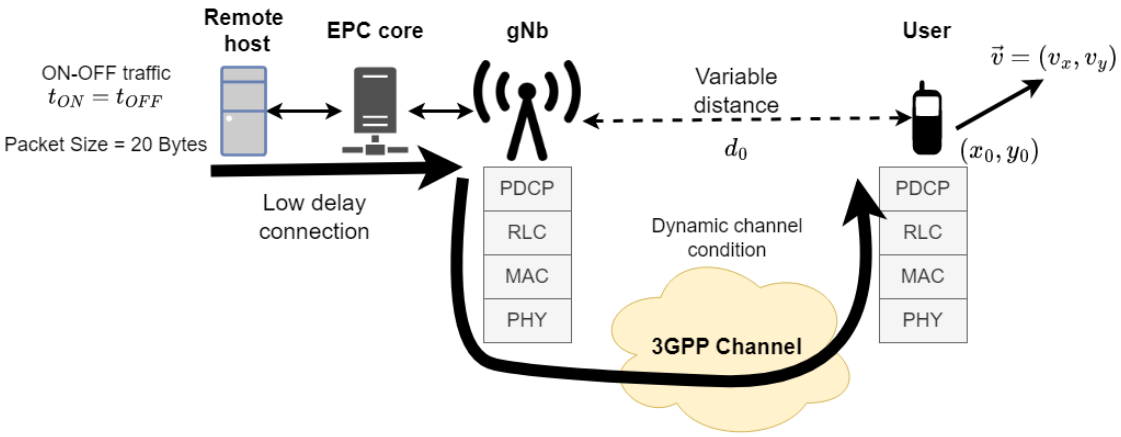

FIGURE 3.12: *Simulation scenario*

3.3.1.4 NUMERICAL EVALUATION

Simulation scenario

Our simulation scenario is illustrated in Figure 3.12. Initially, the UE is placed at a distance d_0 from the gNb and moves away from it with constant speed and direction (v_x, v_y). In this work, packets are generated in distinct ON and OFF periods that follow the Internet Protocol (IP) traffic model **[76]**. The average duration of the ON and OFF periods are t_{on} and t_{off}, respectively. In the ON state, packets of variable size are generated at the application layer with an arrival rate of λ_{ON} [Packets/s] and fill $Q_1(t)$. The performance is evaluated in terms of RAN latency, reliability outage, packet loss and resource efficiency. We define the resource efficiency as the ratio of the number of radio resources required for a TB to be received by the receiver to the number of radio resources allocated by the scheduler. We also define the RAN latency as the times between the arrival of IP packets in the RLC layer of the gNb and their arrival in the IP layer at the UE side. In the scheduling process, K_1 and L_{12} **[74]** are modelled to illustrate the feedback processing time and data preparation time at the UE and gNb, respectively. For simplicity, we assumed that the core network latency and propagation delay are negligible. Table 3.2 summarizes all parameters including communication band, transmit power P_{tx}, target BLER ϵ_t, maximum number of proactive action a_{max} and proactive retransmission per action K_{max} and risk threshold ζ_o.

Performance evaluation

In this section, we compare the performance of several HARQ-based allocation strategies (i.e. our adaptation algorithm, a fixed proactive and a reactive) between a gNb and single mobile user.

Figure 3.13 illustrates how the objective function and resource efficiency behave as a function of the parameter ν. For small value of ν, our algorithm tends to minimize the expected changes in the Lyapunov function, i.e $L(\boldsymbol{\Theta(t)})$ rather than the number of allocated resources (i.e. f_{obj}). By doing so, a

TABLE 3.2: *Simulation Parameters*

Parameters	Values
(f_c, BW)	(3.5 GHz, 50 MHz)
P_{tx}	8 dBm
(U_{tx}, S_{rx})	$(4 \times 4, 2 \times 2)$
d_0	110m
Velocity (v_x, v_y)	(4,4) m/s
$BLER\ \epsilon_t$	10^{-4}
ζ_o	0.05
$Numerology$	1
Processing delay (K_1, L_{12})	(2,2) time slots
K_{max}	10
MCS (m, CR)	(5, 0.3701)
a_{max}	5
(r_{min}, r_{max})	(2, 5)
Packet Size L (Application level)	Exponential(20) Bytes
ON-OFF Traffic	t_{on} = t_{off} = 2.5 ms, λ_{ON}=50000 Packets/s

large number of radio resources are generously provided to each TB for proactive RTXs, so the average resource allocation is high and the resource efficiency is low. When the value of v increases, the focus is on minimising the objective function and fewer resources are allocated. At a certain value of v (i.e. around 50), we found that a minimum value of the objective function is reached (i.e. around 3.2). This is because at a high value of v, the algorithm favours allocating smaller number of parallel RTXs on each action and thus more proactive RTX action are required for each TB. Therefore, in our future evaluations, we will set v to 60.

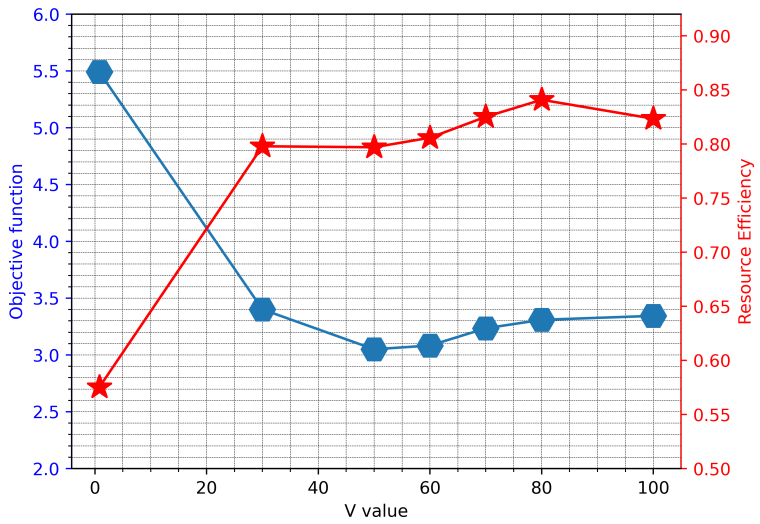

FIGURE 3.13: *Objective function and resource efficiency as a function of* v

Figure 3.14 compares the CDF of latency for different HARQ-based allocation schemes: *(i)* reactive allocation scheme (i.e. $R = 1$) with a maximum number of 10 RTXs, *(ii)* 5 RTX actions where each action contains 2 proactive RTXs (2-2-2-2-2), *(iii)* 4 RTX actions where the first three actions contain 3 proactive RTXs and the last action contains a single RTX (3-3-3-1) and *(iv)* our proposition with a maximum of 5 actions. Figure 3.14 shows that the reactive allocation scheme performs the worst compared to the other solutions due to the high RTT cost associated with triggering many RTXs. By enabling two parallel RTXs per received NACK, the latency can be improved at the cost of decreasing resource efficiency to around 0.82 compared to 1 in the reactive case. The latency can be further improved with 3 proactive RTXs per action, but the resource efficiency drops significantly to 0.69 when we redundantly provide radio resources for a TB to be successfully decoded. Concerning the performance of our proactive allocation, the performance of latency is slightly better than the others cases. As the channel condition and traffic behaviour varies dynamically, our decision maker will wisely select different redundancy level according to the current queues status and instantaneous channel condition. As the result, we can enhance the latency while keeping a good level of resource efficiency at around 0.8.

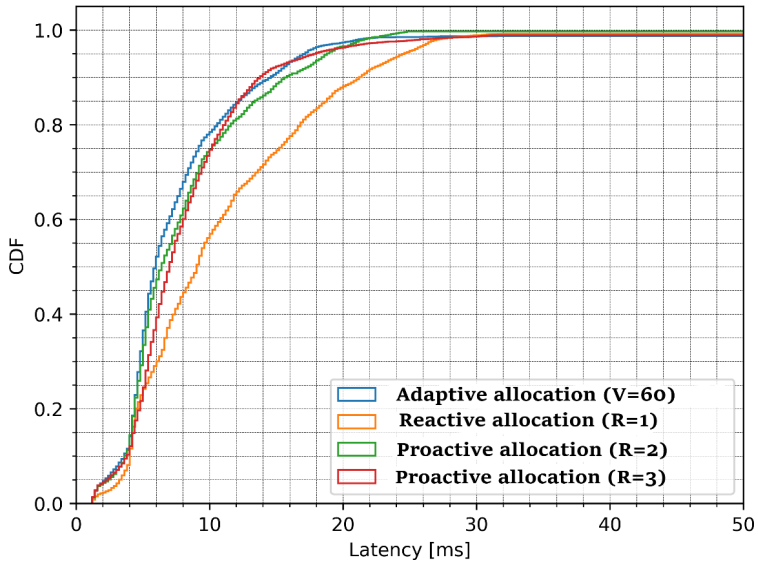

FIGURE 3.14: *CDF of latency for reactive allocation, fixed 2-parallel, 3-parallel allocation and our adaptation algorithm with* $v = 60$

To show the temporal variation in MAC layer delay of the different allocation schemes for intermittent URLLC traffic scenarios under time-varying channel conditions, Figure 3.15 compares the latency between the departure of a TB at the sender's MAC layer and its successful arrival at the receiver's MAC layer. The reactive allocation scheme experiences more peaks as it needs more time to successfully decode a packet when critical errors occur. When fixed proactive allocation schemes are applied, 2-parallel strategy reduces the peak delay and 3-parallel strategy further improves the delay when the radio resources allocated in parallel help decode the packet faster. Our proposed algorithm also reduces the number and amplitudes of peaks when it dynamically selects an optimal level of proactive redundancy for each action, thus achieving a better trade-off between reserving more radio resources to recover the corrupted packet faster and less to maintain a minimum objective function.

In addition, various performance metrics such as application packet loss (APP loss), resource efficiency as well as average delay and its standard deviation for different allocation schemes are discussed. First, the APP loss is guaranteed to be between 0.8 % and 1 % because the maximum number of retransmissions (i.e. $K_{max} = 10$) is applied in all cases. Second, we find the largest average delay and its standard deviation in reactive allocation whose values are around 10.77 ms and 6.57 ms, respectively. These coupled values are enhanced to roughly 8.14 ms, 4.91 ms and 8.1 ms, 4.5 ms in fixed 2-parallel and 3-parallel

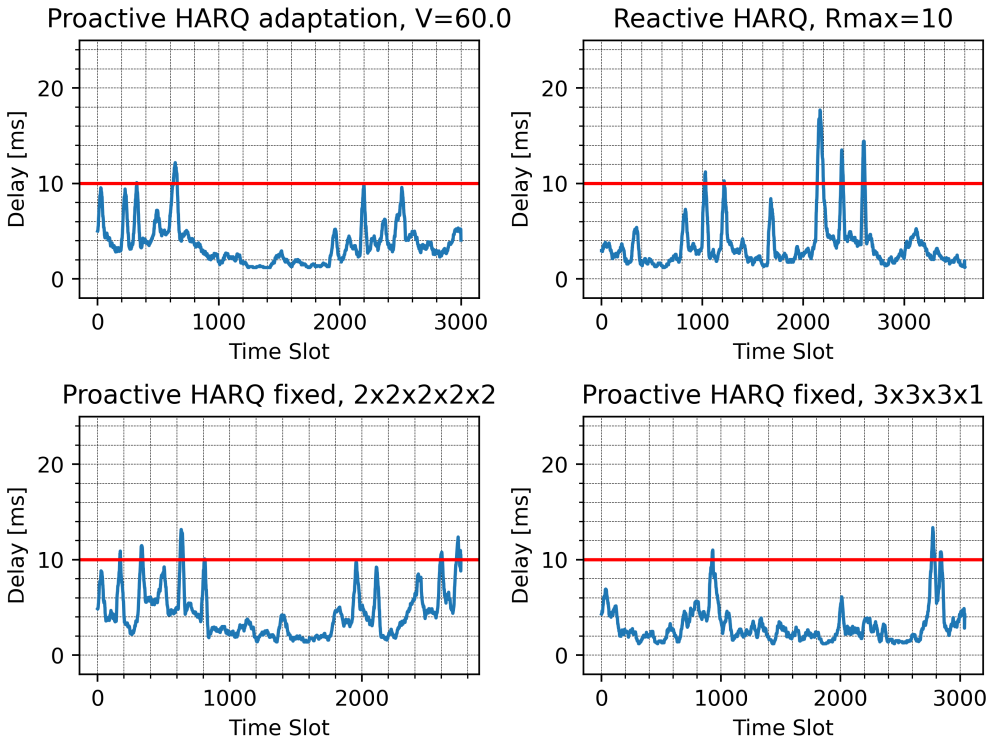

FIGURE 3.15: *Evolution of MAC delay between HARQ schemes*

allocation schemes at the expense of resource efficiency which decreases to 0.82 and 0.69, respectively. By dynamically deciding the number of proactive retransmissions at $\nu = 60$, the average delay and jitter are slightly better than other HARQs which are about 7.2 ms and 4 ms while maintaining a good resource utilization level at about 0.8.

TABLE 3.3: *End-to-end evaluations between HARQ schemes*

Performance metrics					
Schemes	APP loss (%)	PHY loss (%)	Resource Eff	Mean delay	std delay
Reactive regime	0.8 %	0.8 %	1	10.77 ms	6.57 ms
Fixed proactive 2-2-2-2-2	0.8 %	0.4 %	0.82	8.14 ms	4.91 ms
Fixed proactive 3-3-3-1	1 %	0.8 %	0.69	8.1 ms	4.5 ms
Proactive adaptation with $\nu = 60$	1 %	0.6 %	0.80	7.2 ms	4 ms

Remarks 3.3.1

In this study, we demonstrate the application of decision maker framework that enables a novel proactive HARQ design to cope with a time-varying channel and intermittent traffic source rate. Based on Lyapunov stochastic optimization tool, a mathematical framework is proposed to understand the performance-delay trade-off by minimizing the objective function of the total resource allocation and the total queue length that is parameterized by a ν value. Our results reveal that an appropriate selection of ν enables the dynamic selection of proactive retransmission to overcome the defect of a long RTT in the reactive scheme while maintaining a good level of resource efficiency that is considered a drawback of the fixed proactive schemes. In the upcoming work, this study will be extended towards the guarantee of short-term reliability constraints. The resource efficiency-latency-reliability trade-off will be fully achieved by the number, timing and intensity of the decisions.

3.3.2 Adaptive, reliability-aware resource allocation with short-term reliability requirement

In this work, we propose an adaptive and reliability-aware allocation based on Lyapunov optimization for Ultra Reliable and Low Latency Communications (URLLC), taking into account the arrival of traffic at the network layer, the behaviour of queues at the data link layer, and the risk that the applied decision results in packet losses. Particularly the trade-off between the resource efficiency, latency and reliability is achieved by the timing and intensity of decisions and it is adapted to dynamic scenarios (e.g., random bursty traffic, time-varying channel). More importantly, the short-term reliability is also considered to complement the previous study on long-term reliability constraint based on Lyapunov's optimization theorem.

The contributions of this work are as follows: *(1)* We formulate the adaptive, reliability-aware resource scheduling problem by considering the traffic arrival in the network layer, the queue behaviours in the data link layer and the instantaneous risk of applying vulnerable decision which causes packet loss. *(2)* The proposed solution includes resource efficiency considerations for URLLC applications whereas most solutions in the literature only consider latency reliability tradeoff. *(3)* We consider E2E performance by developing a system-level simulator based on NS3 **[3]** applying to the NR. This simulator handles several HARQ processes and measures the latency between the transmitter RLC layer and the receiver RLC layer assuming that transmission buffer size is infinity. We therefore consider both the queuing latency at the scheduler (due to reactive/proactive approaches) and (re)transmission latency (i.e. PHY/MAC).

3.3.2.1 System model

Our system model is based on the previous description in Section 3.3.1.1 where a two-stage queuing system $Q_1(t)$ (dynamic traffic queue) and $Q_2(t)$ (dynamic transmission queue) is introduced. The queuing dynamic is defined as follows:

$$Q_1(t+1) = \max\{Q_1(t) - \alpha_a.TB_{a_0}, 0\} + A_1(t) \tag{3.29}$$

$$Q_2(t+1) = \max\{Q_2(t) - (1-\alpha_a).\mathbb{1}_{TB}.TB_{a_j}, 0\} + A_2(t) \tag{3.30}$$

where $Q_i(t+1)$ are the backlogs of the queue i at the action slot $t+1$. $A_1(t)$ represents the total amount of high layer packets that arrive Q_1 at time t. During this action slot, an amount of TB_{a_j} will be served. The indicator function $\mathbb{1}_{TB}$, in Equation (3.30), is equal to 1 if the scheduling process of TB is successful and is 0, otherwise. If the first transmission of TB_{a_0} is a failure, $A_2(t) = TB_a$ will be added to Q_2, otherwise $A_2(t) = 0$ as the scheduling process of TB_{a_0} is ending. In order to control which queue will be served, we introduced the control variable α_a (1 and 0 mean serving $Q_1(t)$ and $Q_2(t)$ respectively). Knowing that ongoing processes have a higher priority, $\alpha_a = 0$ when $Q_2(t) > 0$.

Instead of limiting the maximum number of allowed RTXs K_{max} for the scheduling process, our dynamic resource scheduling is restricted in terms of maximal possible actions a_{max}. Each action $a_j \in \{a_0, ..., a_{max}\}$ can allocate r_{a_j} proactive RTXs, between r_{min} and r_{max}, and a_0 corresponds to the first transmission. The decision maker we designed, dynamically chooses the number of actions a_j and their intensities (i.e, r_{a_j}) to reduce latency and improve resource efficiency and reliability. With respect to the short-term reliability requirement, we define the instantaneous risk (i.e $\zeta(a_j)$) as follows:

$$\zeta(a_j) = \mathbb{P}[(\gamma_s^{\sum_{k=0}^{j} r_{n,a_k}} \leq \gamma_t) \,|\, \gamma_s^{\sum_{k=0}^{j-1} r_{n,a_k}}] \tag{3.31}$$

where $\gamma_s^{\sum_{k=0}^{j-1}(r_{n,a_k})}$, $\gamma_s^{\sum_{k=0}^{j}(r_{n,a_k})}$ are respectively the SINR of TB_n at previous (a_{j-1}) and current (a_j) action. γ_t is the target SINR to decode TB_n.

Our proposed procedure dynamically adapts the resource scheduling to the traffic arrival in the network layer, the queue behaviours in the data link layer and the risk that the applied decision causes loss. It also automatically adapts the maximum number of RTX to the channel conditions. Finally, to reduce the control overhead due to multiple feedbacks to the transmitter, we grouped their feedbacks into a single feedback that represents the current proactive retransmission status.

3.3.2.2 Problem formulation

The objective is to optimally select r_{a_j} based on various factors, such as the current status of the $Q_1(t), Q_2(t)$, the current action index a_j and the risk that the applied decision causes loss. The main reliability constraint is to reduce the risk of the last action $\zeta(a_{max})$ below a predefined value ζ_o. However, the constraint associated with poor decision making must be defined for each upcoming action, not just for the last action. We define the risk for the current action $\zeta(a_j)$.

In this case, the procedure has to re-trigger other actions later, which consumes not only time and resources but also the reliability of the communication, when we are close to the maximum number of actions allowed. The index of the current action (i.e a_j) is thus very important. Clearly, the higher j is, the greater the sensitivity of TB loss will be if a wrong decision is applied, and the earlier the action (i.e low j) is, the higher the total number of RTXs can be.

We define an objective function f_{obj} as the weighed sum of average number of resources allocated to each TB and the current risk, as follows:

$$f_{obj} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \sum_{a_0}^{a_{max}} r_{n,a_j} \times 1_{a_j} + \alpha \times f(a_j) \times \zeta(a_j) \tag{3.32}$$

where the indicator function 1_{a_j} is equal to 0 if the action a_j is successful (i.e. $\zeta(a_j) < \zeta_o$) and is 1, otherwise. $\alpha \geq 0$ is a constant value trading off risk and resource allocation. A higher value of α implies greater importance of risk minimization over the number of resources allocated (i.e. reliability over resource efficiency). The function $f(a_j)$ increases with the action index a_j. In our study, we consider $f(a_j) = j$.

Thus, our optimization problem $\mathcal{P}_1$ is to minimize the objective function f_{obj} subject to several constraints:

$$\min_{\{r_{n,a_j}\}_{n,a_j}} \quad f_{obj} \tag{$\mathcal{P}_1$}$$

$$\text{s.t.} \quad \lim_{t\to\infty} \frac{\mathbb{E}\{Q_i(t)\}}{t} = 0, \quad \forall i \in \{1,2\}; \tag{$\mathcal{C}_{1,2}$}$$

$$r_{min} \times 1_{a_j} \leq \sum_{a_j} r_{n,a_j} \leq K_{max} \times 1_{a_j}, \ \forall a_j \leq a_{max} \tag{$\mathcal{C}_3$}$$

$\mathcal{C}_{1,2}$ concerns the stability constraint of the queue $Q_{1,2}(t)$. $\mathcal{C}_3$ limits the number of decisions into a_{max} actions and constrains the maximal number of proactive RTXs at action a_j to K_{max}.

3.3.2.3 Proposed solution: Adaptive, reliability-aware allocation

Our decision maker algorithm is based on Lyapunov's optimization tools, which do not require a-priori knowledge of stochastic processes in the ongoing system such as channel dynamics or traffic behaviours, to solve the optimization problem $\mathcal{P}_1$. Given a time-slotted system, we define the current state in the slot t as $\boldsymbol{\Theta(t)} = (Q_1(t), Q_2(t)))$. Next, we define the one-slot conditional Lyapunov drift $\boldsymbol{\Delta\Theta(t)}$ representing the expected change of the Lyapunov function over a slot as follows:

$$\boldsymbol{\Delta\Theta(t)} = \mathbb{E}\{L(\boldsymbol{\Theta(t+1)}) - L(\boldsymbol{\Delta\Theta(t)}) \mid \boldsymbol{\Delta\Theta(t)}\} \tag{3.33}$$

Where $L(\boldsymbol{\Delta\Theta(t)}) = \frac{1}{2}\left[Q_1^2(t) + Q_2^2(t)\right]$ is the Lyapunov function. By minimising both $\boldsymbol{\Delta\Theta(t)}$ and f_{obj}, we can solve the problem $\mathcal{P}_1$ because the queues are stable in terms of average rate and the objective function is minimized. However, according to [70], a *performance-delay* trade-off between these dual objective optimizations can be parameterized by a constant ν. By setting a large positive value to ν, the control algorithm will favor minimizing the objective function f_{obj} over the stability of the average rate queues. Our objective is then to minimize the following *Lyapunov-drift-plus-penalty* function:

$$g(t) = \boldsymbol{\Delta\Theta(t)} + \nu.\mathbb{E}\{f_{obj} \mid \boldsymbol{\Theta(t)}\} \tag{3.34}$$

As defined in [70], the upper bound, $h(t)$, can be derived for any action, any possible value of $\boldsymbol{\Theta(t)}$ and any parameter $\nu > 0$ as follows:

$$\begin{aligned} h(t) = B + \nu.\mathbb{E}\{f_{obj} \mid \Theta(t)\} \\ + \sum_{i=1}^{2} Q_i(t).\mathbb{E}\{A_i(t) - b_i(t) \mid \Theta(t)\} \end{aligned} \tag{3.35}$$

where B is a constant that satisfies:

$$\begin{aligned} B \geq \frac{1}{2}\sum_{i=1}^{2} \mathbb{E}\{A_i^2(t) - b_i^2(t) \mid \Theta(t)\} \\ - \sum_{i=1}^{2} \mathbb{E}\{A_i(t).\min\{Q_i(t), b_i(t)\} \mid \Theta(t)\} \end{aligned} \tag{3.36}$$

The proof for Equation 3.35 can be found similarly as in Section 3.3.1.3 or in Appendix D.

Through the opportunistic minimization framework of a conditional expectation [70], by minimising $h(t)$, the upper bound of the dual objective optimization, we can guarantee that the optimization problem $\mathcal{P}_1$ will be satisfied.

3.3.2.4 Numerical evaluation

Simulation scenario

This simulation scenario is similar to Section 3.3.1.4 except the fact that we do not consider the mobility between the user and the gNb. Because this work serves as a preliminary study for the validation with the OAI which will be detailed in the next section. Thus, our network contains 1 gNb and 1 UE with fixed distance. In this work, packets are generated in exponentially distributed ON and OFF periods that follow the Internet Protocol (IP) traffic model [76]. The average duration of the ON and OFF periods are t_{on} and t_{off}, respectively. In the ON state, packets of variable size are generated with an arrival rate λ_{ON} [Packets/s] and fill $Q_1(t)$. In the scheduling process, K_1 and L_{12} are modelled to illustrate the feedback processing time and data preparation time at the UE and gNb, respectively. For simplicity, we assumed that the core network latency and propagation delay are negligible. Table 3.4 summarizes application, optimization and communication parameters.

TABLE 3.4: *Simulation parameters*

Parameters	Value
(f_c, BW)	(3.61 GHz, 50 MHz)
Transmission power P_{tx}	8 dBm
Antennas (U_{tx}, S_{rx})	$(4 \times 4, 2 \times 2)$
d_0	100 m
Numerology	1
Packet Decoding	Risk $\zeta_0 = 10^{-4}$
MCS (m, CR)	(5, 0.3701)
a_{max}	5
(r_{min}, r_{max})	(1, 5)
Processing delay (K_1,L_{12})	(2,2) time slots
Traffic Type	ON-OFF
Traffic Parameters	t_{on}/t_{off}= 1/3 λ_{ON}= 75000 Packets/s

Performance evaluation

Performance is evaluated in terms of RAN latency, packet loss and resource efficiency. We define resource efficiency as the ratio of the number of radio resources required for a TB to be successfully received to the number of radio resources allocated by the scheduler. We also define RAN latency as the time between the arrival of IP packets in the RLC layer of the gNb and their arrival in the IP layer at the UE side. Figure 3.16 shows the evolution of resource efficiency (solid line) as well as the average total

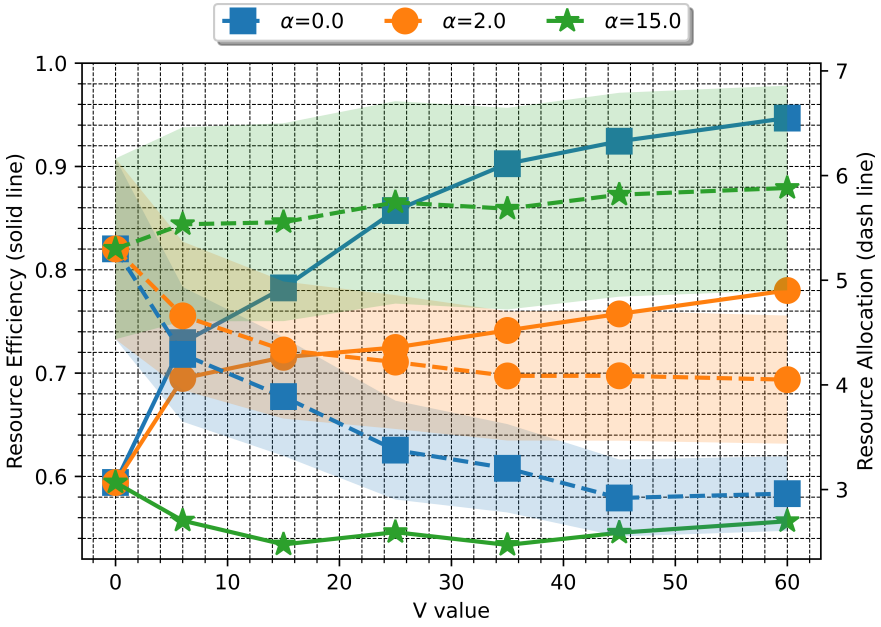

FIGURE 3.16: *Resource allocation and efficiency*

number of radio resources allocated (dash line) as the function of ν and α. We selected three values of α (i.e. 0, 2 and 15) that depend on the awareness of reliable transmission's objective. When reliability is not considered ($\alpha = 0$), our algorithm tends to spend less radio resources for each action and thus, resources are used efficiently. When ν increases, we put more emphasis on minimizing resource allocation, so resource efficiency is further improved. When α appears and grows, the goal of reducing packet loss is also taken into account. The decision maker adapts to the channel conditions and allows more generous allocations for each action and this leads to high resource allocation with high standard deviation and low resource efficiency.

Reliability of communication is guaranteed at the cost of low resource utilization as shown in Figure 3.17. When α is high, the transmission error is significantly low and the communication reliability no longer depends on the ν-value (i.e., 99.5% and 98.5% of the total packets successfully reach the IP layer at the UE side for $\alpha = 15$ and $\alpha = 2$, respectively). However, the dependent relationship between the transmission reliability and the ν-value is observed for $\alpha = 0$. In this case, we barely follow the minimization of the number of resources allocated for each action rather than the reliability of its transmission, thus, we noticed more error-prone transmissions when ν increases.

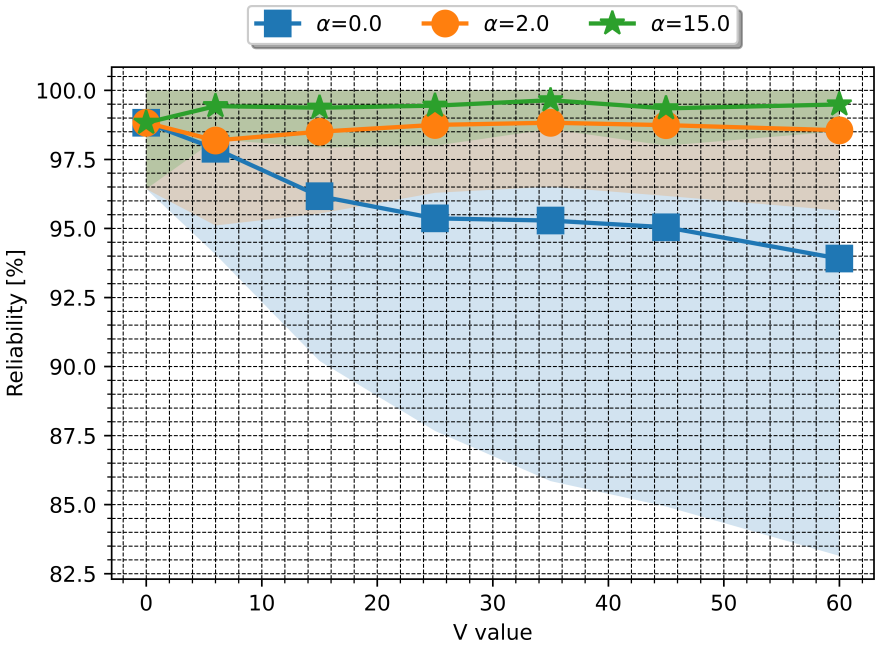

FIGURE 3.17: *Reliability performance*

Figure 3.18 shows that the average latency at the RAN mainly depends on α. Redundant radio resources are scheduled when α is high to improve reliability, but this can result in increased queuing delay as incoming packets must wait longer in the queue before being served.

Figure 3.19 compares the Cumulative Distribution Function (CDF) of latency for different HARQ-based allocation schemes: *(i)* **Classic allocation** procedure, *(ii)* Fixed number of **parallel allocation**, *(iii)* **Adaptive allocation** with a fixed maximum number of RTX ($K_{max} = 10$) as defined in [77] *(iv)* our proposed optimization (**Adaptive, reliability aware allocation**) in which $K_{max} = \sum_{a_0}^{a_{max}} r_{n;a_j}$. According to those figures, we select two pairs of (ν,α) parameters: $(25, 2)$ for good reliability and considerably low latency and $(60, 0)$ for very good resource efficiency and latency. As expected, the latency of Classic allocation is the highest and spreads out over time. 2-parallel and 5-parallel allocation improve latency at the cost of decreasing resource efficiency to 0.8 and 0.6, respectively due to the lack of adaptation when needed. Our adaptive, reliability-aware allocation scheme offers two trade-offs. When $\nu = 60$ and $\alpha = 0$, we improve resource efficiency and latency but not reliability. When $\nu = 25$ and $\alpha = 2$, we improve reliability at the cost of a slight degradation in latency in the best case.

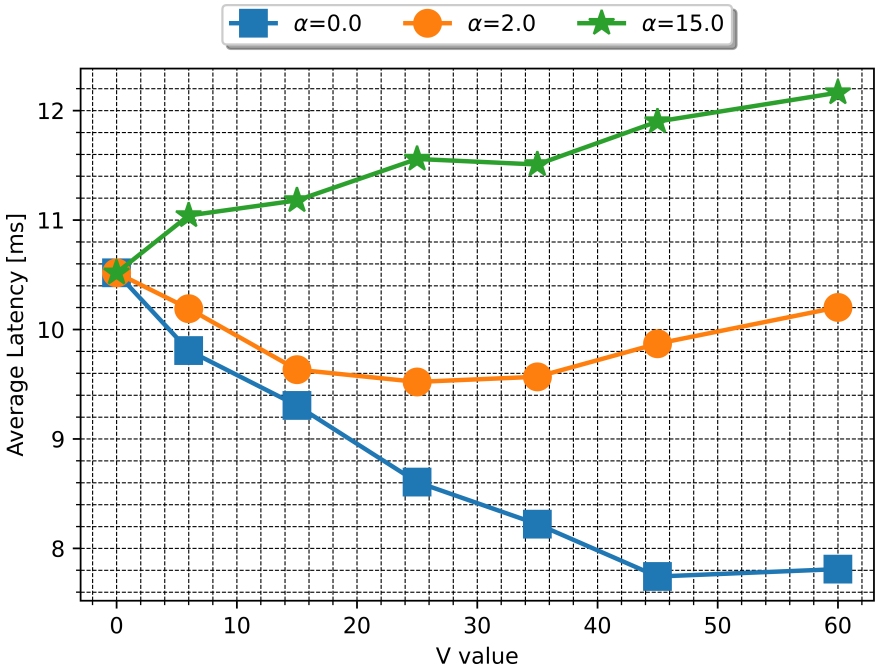

FIGURE 3.18: *Average latency*

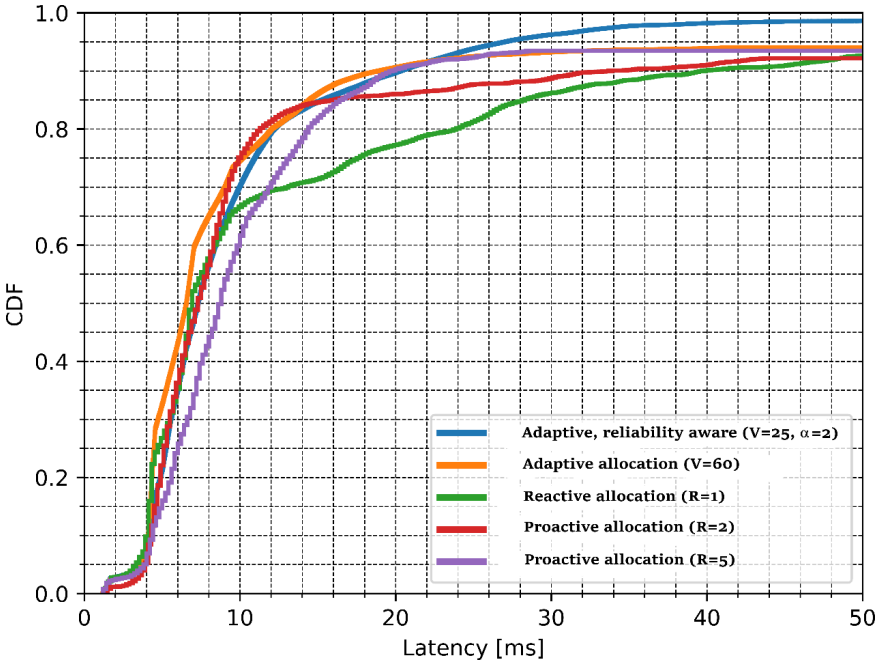

FIGURE 3.19: *CDF of latency for different HARQ-based allocation schemes*

Remarks 3.3.2

In this section, we show the decision maker framework in mission critical communication where the short-term reliability is crucial to be guaranteed in a time-varying channel and intermittent traffic source rate. By well calibrating our proposed optimization to obtain optimal control parameter pairs (ν,α), we successfully choose the one which reduces latency, maintain good resource efficiency and improve reliability by means of system level simulation. In the next section, we would like to validate our proposed algorithm in a 5G-compliant hardware architecture which is based on OpenAirInterface (OAI).

3.4 From system simulation to practical experimentation

In this section, we are going to apply our resource optimization frameworks, which are previously evaluated in system level simulator NS3, into experimentation with the helps of OpenAirInterface (OAI) [78]. In order to provide an 3GPP compliant reference implementation of UE, gNb and Core Network (CN), this open-source platform captures the most fundamental cellular network functions. The core development of this work is mainly carried out at the MAC layer, where the scheduler is implemented. Besides, this layer is responsible for the data transfer between logical channels at RLC layer and transport channels at PHY layer, thus it is important for two-state queuing system managements. This collaboration work is done thanks to the enormous help of Mr. Rodolphe Bertolini, who is responsible for the experimental parts and verifying our proposed algorithms on OpenAirInterface (OAI). The results of our joint simulation and experimental work are published in [79].

3.4.1 Open-Air-Interface Architecture (OAI)

The OpenAirInterface (OAI) software provides an open-source, standard-compliant implementation of a 3GPP NR stack which is exploited on top of x86 CPU and a USRP radio device. This platform is oiginally initiated by Eurocom [80] with the aim at contributing a full experimental LTE implementation which runs in real-time and supporting commercial LTE handsets. Currently, the OAI software spans the full protocol stack of 3GPP LTE, NR standards which include implementations of EUTRAN (both gNb and UE) and EPC (MME, S+P-GW and HSS).

The gNb physical layer software implements 3GPP 36.211 [81], 36.212 [82], 36.213 [83] and provides following features:

- FDD and TDD configurations: 1 and 3
- Transmission modes: 1, 2 (stable); 3, 4, 5, 6, 7 (experimental)
- Max number of antennas: 2
- PRACH preamble format 0
- All downlink (DL) channels supported: PSS, SSS, PBCH, PCFICH, PHICH, PDCCH, PDSCH, PMCH
- All uplink (UL) channels supported: PRACH, PUSCH, PUCCH (format 1/1a/1b), SRS, DRS
- HARQ support for UL and DL
- Highly optimized baseband processing (including Turbo decoder)
- X2 interface and handover.
- Release 12 Dual Connectivity (DC)

Then, gNb MAC implementation includes:

- RRC interface for CCCH, DCCH, and DTCH
- Proportional fair scheduler, round robin scheduler soon.
- DCI generation
- Support for HARQ
- RA procedures and RNTI management
- RLC interface
  - Acknowledged Mode (AM)
  - Unacknowledged Mode (UM)
- UL power control
- Link adaptation

The PDCP layer is header-compliant with 3GPP 36.323 **[84]** and have following functionalities:

- User and control data transfer
- Sequence number management
- RB association with PDCP entity
- PDCP entity association with one or two RLC entities
- Integrity check and encryption using the AES and Snow3G algorithms

The RLC layer implements the full specification of 3GPP 36.322 **[85]**.
the RRC layer is based on 3GPP 36.331 **[86]** and have following features:

- System Information Broadcast (SIB) formats 1, 2, 3, and 13
- RRC connection establishment, reconfiguration, release, re-establishment
- RRC inactivity timer
- Inter-frequency measurement collection and reporting
- eMBMS for multicast and broadcast
- X2 Handover
- Paging

3.4.2 Experimentation test-bed

Our experimentation consists of several components:

- Two powerful machines which are the representatives of UE and gNb, each has 2×14 CPUs and 64 GB of RAM.
- Two Software Defined Radio (SDR) cards illustrate the radio head of terminal and access point (USRP b210)
- Two variable attenuators showing the path loss in DL and UL direction.
- SMA cables which is responsible for the signal transmission at 3.61 GHz.

As shown in Figure 3.20, the computer on the left hand side (red rectangle) is used to run an instance of OAI gNb. The ones on the left hand size (green rectangle and yellow rectangle) model the instances of OAI UE and CN, respectively. We consider that in a situation with few UE, the UE and CN are the parts that have less processing time needed, so we gathered both into a single computer and let the processing time-hungry gNb make full use of a two-CPU machine.

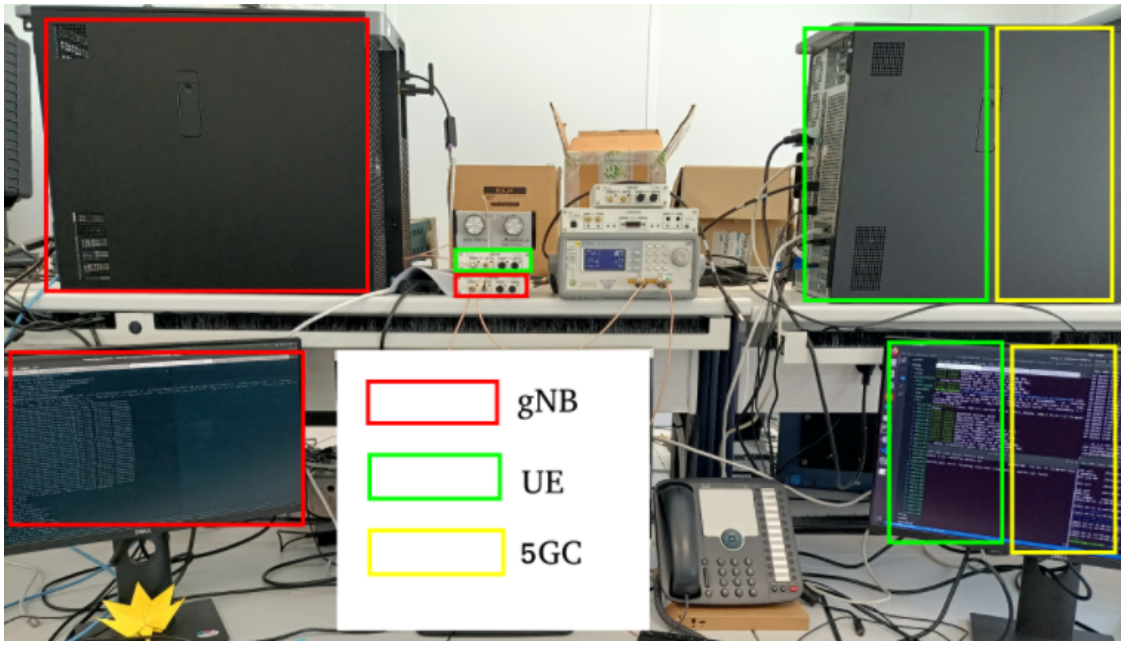

FIGURE 3.20: *OAI experimentation Testbed*

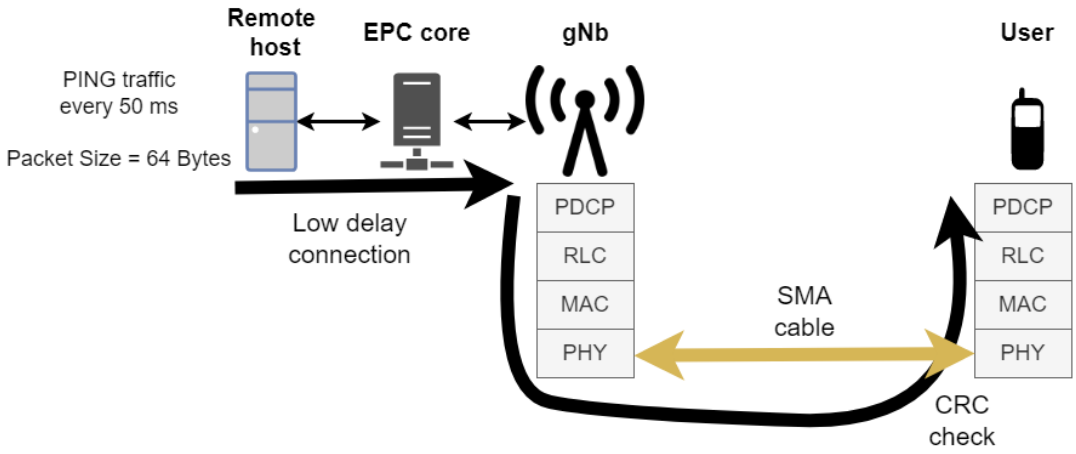

FIGURE 3.21: *Demonstration of OAI experimentation*

In our experimental scenario, the variable attenuators are used to experiment several channel conditions between the UE and the gNb. Packets generated are PING of 64 bytes every 50ms.

In our performance evaluation, we have implemented and compared several HARQ-based allocation schemes: **(i) Classic allocation** refers to the wait-NACK-before-re-transmit scheme, **(i) Fixed proactive resource allocation** (from 2 to 5 parallel retransmissions) as Section 3.2, **(ii) Adaptive resource allocation** with long-term reliability guarantee as defined in Section 3.3 and **(iii) Adaptive, reliability-aware resource allocation** with short-term reliability guarantee. In order to find the suitable values of ν (Proactive and Long-Term Dynamic) and α (Short-Term Dynamic), we run with different values of ν and α and select the value for which a given metric is optimised.

We have implemented the different resource scheduling optimization schemes in the develop version of OAI NR RAN. This includes (but is not limited to) modifying the redundant version of a TB and the HARQ process to handle parallel retransmissions of the same TB by gNb and UE MAC entity and collecting metrics needed to our algorithm, such as DL channel status, buffer length (i.e. Q_1 and Q_2) in the RLC layer and risk probability based on SINR measurements.

3.4.3 Deviations from simulation

Our simulation scenario is demonstrated in Figure 3.21 where the current status of 5G implementation in OAI coupled with the limited capabilities of the USRP do not offer the same freedom as the NS3 simulator. In this work, our aim is not to directly compare the performance between simulation and experimentation, but to verify if our solution is feasible in a real environment and to show the gain brought by our solution with real time hardware constraints.

Our implementation starts from the developed version of OAI NR RAN. The main difference with simulation concerns the spectrum usage technique. While simulations use an FDD, OAI uses a TDD. In a 10-slot frame, the first 6 slots are dedicated to DL and the last 3 slots to UL. The seventh slot is a flexible slot (Flex) composed of 6 DL symbols and 4 UL symbols.

According to this implementation, the RRC layer of the UE sets K_1 to a minimum of 6 slots to allow OAI sufficient processing time, and the gNb scheduler sets a delay L_{12} of at least 3 slots. Moreover, in our experiment, each TB contains a PING packet instead of aggregated application packets. Table 3.4 summarizes OAI experimentation parameters.

For the implementation of our different resource scheduling optimization schemes, Equation (3.31) needs the DL channel status. The CQI is usually calculated with the SINR of the transmission occurring in the PDSCH to be acknowledged in the current UCI. As the CQI in the UCI is not implemented in the current version of OAI, we extrapolate it using the UL channel estimation performed by the gNb in our TDD configuration and we estimate the risk (i.e $\zeta(a_j)$) based on online statistics.

In order to illustrate the deviation from system simulation to hardware experimentation based on OAI, Table 3.5 shows the different parameters between them.

Table 3.5: *Simulation and Experimentation Parameters*

Parameters	**Simulation**	**Experimentation**
DL/UL Duplex	FDD	TDD
DL/Flex/UL	N/A	6/1/3 slots per Frame
(f_c, BW)	(3.61 GHz, 50 MHz)	
Numerology	1	
(m, CR)	(5, 0.3701)	(3, 0.2451)
P_{tx}	8 dBm	-8 dBm
(U_{tx}, S_{rx})	$(4 \times 4, 2 \times 2)$	(1, 1)
a_{max}	5	
(r_{min}, r_{max})	(1, 5)	
(K_1, L_{12})	(2,2) time slots	(6,3) time slots
Packet Decoding	Risk $\zeta_0 = 10^{-4}$	CRC check
Traffic Type	ON-OFF	PING
Traffic Parameters	t_{on}/t_{off}= 1/3 Data Rate= 1.5 Mbps	Every 50 ms 64-Byte packet

3.4.4 Performance Evaluation

Firstly, we depicts the evolution of the resources allocated for each transmission and its associated efficiency as a function of ν. As shown in Figure 3.22, a downward trend is observed when ν increases. In

particular, the degree of better resource efficiency is negatively proportional to the amounts of resources allocated. Then, we decide to choose a value of $\nu = 55$ that maximizes the resource allocation efficiency.

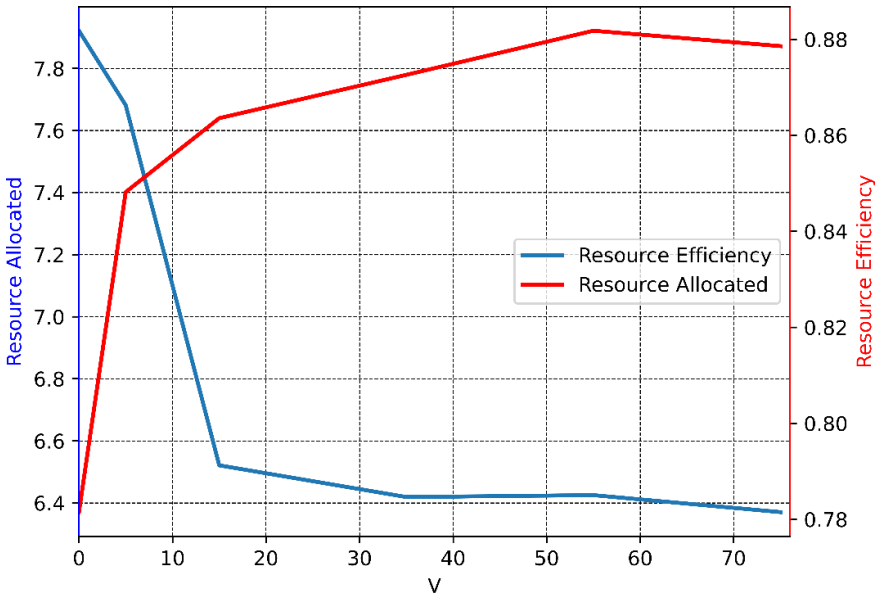

FIGURE 3.22: *Resource Allocation and Resource Allocation Efficiency depending on* ν

Figure 3.23 compares the CDF of latency between ***(i)* classical, reactive** allocation schemes, ***(ii)* fixed proactive** allocation schemes in Section 3.2 and ***(iii)* adaptive** allocation schemes where long-term reliability is taken into account as in Section 3.3. We can see first that fixed parallel 4 (4-4-2) and 5 (5-5) are very close to each other, which means that in the current testbed with the current channel state, allocating 4 or 5 resources does not improve by factors the performance latency wise: for both, 100 % of the packets are located under 42.5ms. On the other hand, we can see that Classic allocation gives the worst results with 100% of the packet being under 72.5 ms. Parallel 2 performs better with 100% of the packet under 60 ms, Parallel 3 records 52.5 ms. Our algorithm which optimises resource allocation at $\nu = 55$, has 100% under 52.5 ms. If we focus on the CDF at 60%, we have the following observations:

- Classic, reactive allocation reaches 47.5 ms
- Fixed proactive 2-2-2-2-2 allocation reaches 42.5 ms
- Fixed proactive 3-3-3-1 allocation reaches 37.5 ms
- Fixed proactive 4-4-2 allocation reaches 27.5 ms
- Fixed proactive 5-5 allocation reaches 30 ms
- Adaptive allocation with $\nu = 55$ reaches 32.5 ms

In the upcoming part, we will compare those results with the one where **short-term reliability** with regards to **adaptive, reliability-aware algorithm** is taken into account. Table 3.6 shows the BLER, latency, resource usage and resource usage efficiency depending on the values of ν (columns) and α (lines). The most importance numerical values will be highlighted as following: Firstly, with extremely low ν or extremely high α the algorithm allocates a high amount of resources. As we are over allocating in current scenario, the resource usage is the highest, from 7 to 9; resulting in the lowest resource efficiency – that is the ratio of resources needed over resources allocated – of around 0.72 to 0.78, which means 28 to 22 % of the allocated resources are wasted. However, in such configuration, the BLER is at its lowest: around 2%. Secondly, in orange, with α = 0 so we do not consider the reliability, and with an extremely high ν. In such configuration, we are close to the standard HARQ that would allocate 1 retransmission per cluster. Without surprise, the resource usage is at its lowest, at 3.6; the resource efficiency at its

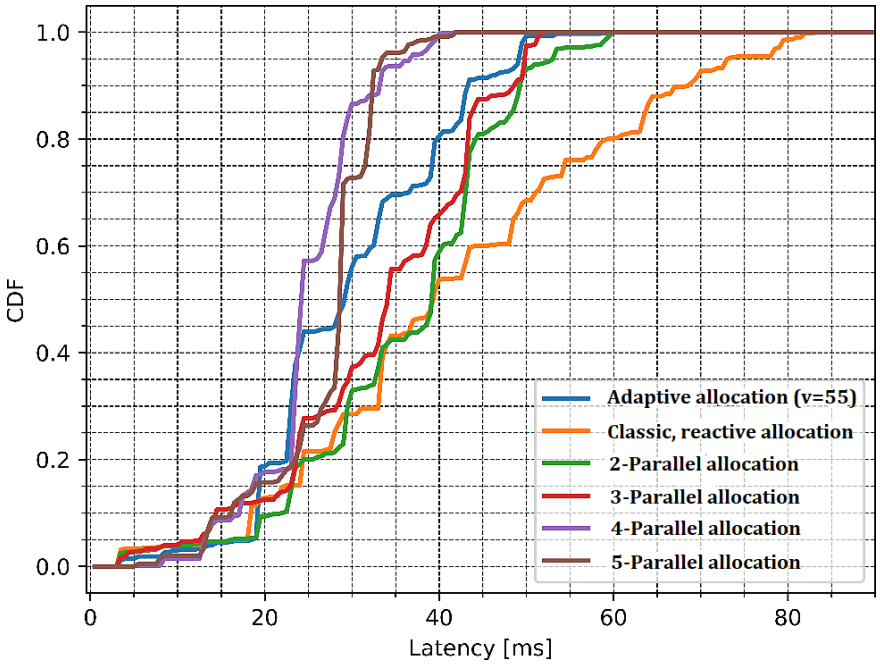

FIGURE 3.23: *Cumulative Distribution Function of Latency completion of HARQ processes for classical, fixed proactive and adaptive schemes*

highest, at 0.95; and the BLER is at its lowest with 40 % HARQ error. Finally, in blue, with $\alpha = 1$ and $\nu = 5$, we have a BLER that is around that of what we see in scenarios with statically allocated 2 and 5 retransmissions. In such case, we see a mean resource usage of 5 and a resource efficiency of 0.89.

Figure 3.24 and Figure 3.25 respectively shows the relation between resource allocation, resource allocation efficiency and the reliability in terms of BLER, resource allocation efficiency with different values of α and ν. We can see clearly that the higher the α, the more resources we allocate per action, the less efficient we are in exchange for better reliability.

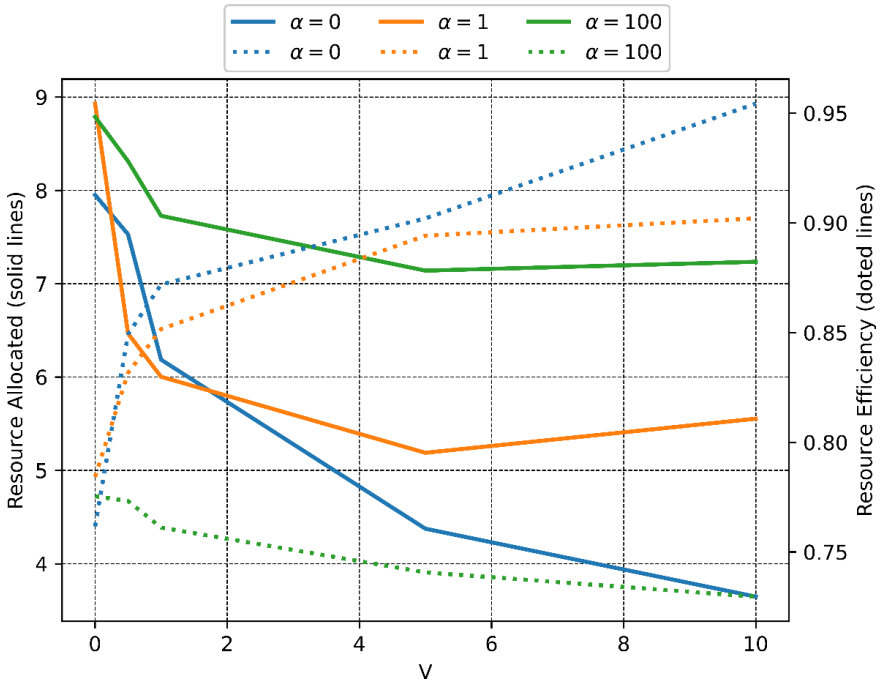

FIGURE 3.24: *Resource allocation and resource allocation efficiency depending on ν value for different values of α*

TABLE 3.6: *Performance of BLER, Latency, Resource Allocation, Resource Allocation Efficiency depending on α and ν*

BLER [%]	ν=0.001	ν=0.5	ν=1	ν=5	ν=10
$\alpha = 0$	2.23048327	9.3117409	22.163908	27,478754	40,437788
$\alpha = 1$	2.53560264	5.2167783	13,411765	15,201192	21,046611
$\alpha = 10$	2.34246123	2.9767041	3,2193827	4,1683694	4,1200407
$\alpha = 100$	2.27120908	2.4323084	1,6408814	1,9742883	2,1047228
LATENCY [ms]	ν=0.001	ν=0.5	ν=1	ν=5	ν=10
$\alpha = 0$	19,5289303	25,370268	24,19107	20,536601	21,314922
$\alpha = 1$	22,9312745	22,356983	22,705722	24,841549	26,204719
$\alpha = 10$	20,8545892	20,941006	20,595434	22,084314	20,492283
$\alpha = 100$	22,4786868	21,196891	20,47451	18,221003	18,563746
RESOURCE USAGE	ν=0.001	ν=0.5	ν=1	ν=5	ν=10
$\alpha = 0$	7,95089456	7,5285616	6,1858775	4,3751634	3,6472868
$\alpha = 1$	8,92859693	6,4655493	6,0027248	5,1883803	5,553539
$\alpha = 10$	8,11859746	7,0828139	6,5372468	6,7738562	7,0766365
$\alpha = 100$	8,7872121	8,3122939	7,7267974	7,1397223	7,2345226
RESOURCE EFFICIENCY	ν=0.001	ν=0.5	ν=1	ν=5	ν=10
$\alpha = 1$	0,76157418	0,8491635	0,8719154	0,9020018	0,9543039
$\alpha = 15$	0,78419769	0,8316244	0,851566	0,8941296	0,9020261
$\alpha = 10$	0,76627006	0,7922429	0,8130547	0,8214975	0,795217
$\alpha = 100$	0,77540881	0,7732192	0,7611233	0,7407012	0,7296396

Figure 3.26 depicts the resource allocation and resource efficiency as a function of optimization parameters (i.e. ν and α). Due to the implementation deviations detailed in the subsection 3.4.3, the size of the queues, and thus the weights ν and α, are different from those of the simulations. For $\alpha = 0$, the decision maker does not consider reliability and mainly optimises resource efficiency. When ν increases, the decision maker allocates fewer resources, which leads to greater resource efficiency. For this experiment, the right average level of the number of resources allocated is between 3 and 4. When α increases, the decision maker trades off reliability (which needs more resources) and efficiency (which limits the number of resources allocated). So the larger α is, the less efficient the scheduling is.

Figure 3.27 shows the latency CDF obtained by experimentation with OAI for different HARQ-based allocation procedures (i.e. **Classic, reactive allocation**, **Fixed proactive 2,5-Parallel allocation**, **Adaptive, proactive allocation** with $\nu = 0.06$ and **Adaptive, reliability-aware allocation** with $\nu = 5$ and $\alpha = 100$). In this experiment, the latency is measured at the MAC layer, instead of the upper application layer. The approximate 3 ms staircase shape of the curve is explained by the TDD DL/UL duplexing of our experiment. Indeed, since (K_1, L_{12}) is (6,3) time slots, there are, for example, 3 ms (i.e. 6 consecutive DL slots) before the UL transmission.

We can see that the stronger the action, the lower the latency. The gain in latency is 30% and 55%

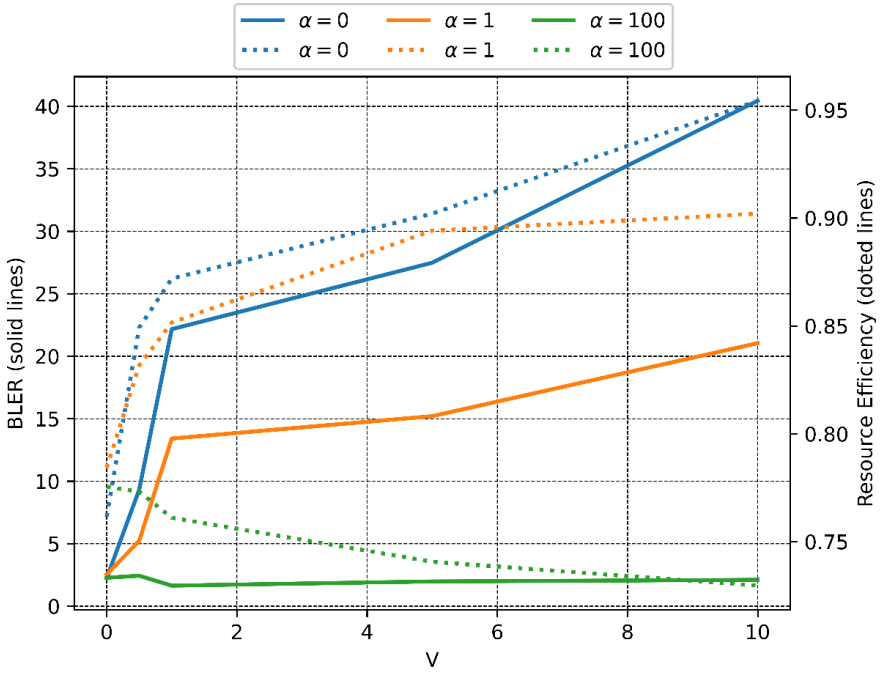

FIGURE 3.25: *Reliability and resource allocation efficiency depending on v value for different values of α*

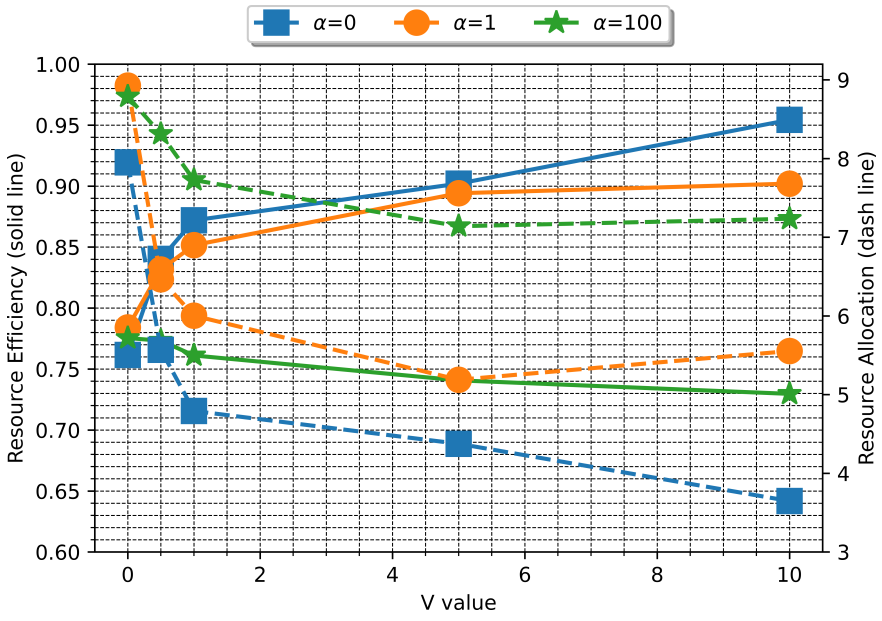

FIGURE 3.26: *Resource efficiency and resource allocation per TB as a function of optimization parameters*

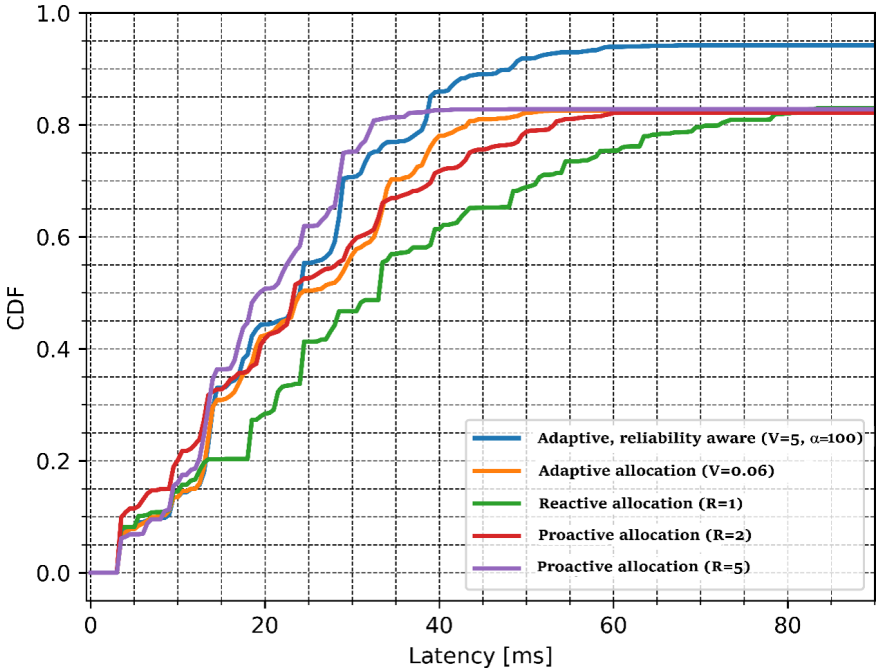

FIGURE 3.27: *Experimental latency CDF for different HARQ schemes*

between Classic, reactive allocation and Fixed proactive 2,5-Parallel schemes, respectively. The average latency of successfully completed HARQs is 29.5 ms, 21.7 ms, and 17.8 ms for Classic, reactive allocation and fixed proactive 2,5-Parallel allocation, respectively. Adaptive allocation automatically adjusts the intensity of each action and trades off resource efficiency and latency, but it is limited by the maximum number of RTXs ($K_{max} = 10$). Due to this limitation, it achieves the same upper bound of 88% HARQ completion as the other schemes. The average latency of adaptive allocation with $\nu = 0.06$ is 21.7 ms, similar to that of 2-Parallel allocation. A close upper bound is achieved by our adaptive, reliability-aware allocation with $\nu = 5$ and $\alpha = 1$. By setting α greater than 1 (i.e. 100), our decision maker outperforms other reliability, since it reaches 95% of completion, while ensuring an average latency of 18.2 ms.

Finally, we examine the RAN latency with the inclusion of RLC layer functionalities. Figure 3.28 shows the CDF of the latency at the RLC layer. First we note that the values of the latency are higher than previously. Indeed, HARQ processes sometimes fail to transmit a packet. As it is Acknowledged Mode (AM), the packet remains in RLC until it is well transmitted, thus it may need few HARQ processes to well transmit a packet, and thus the RLC latency is greater than MAC latency by factors. Finally we also note that the RLC latency in the case of Dynamic algorithm is lower than others by factors (x axis is log scale). This is thanks to the high reliability ensured by the algorithm. If less HARQ fail, then packet spend less time in RLC layer.

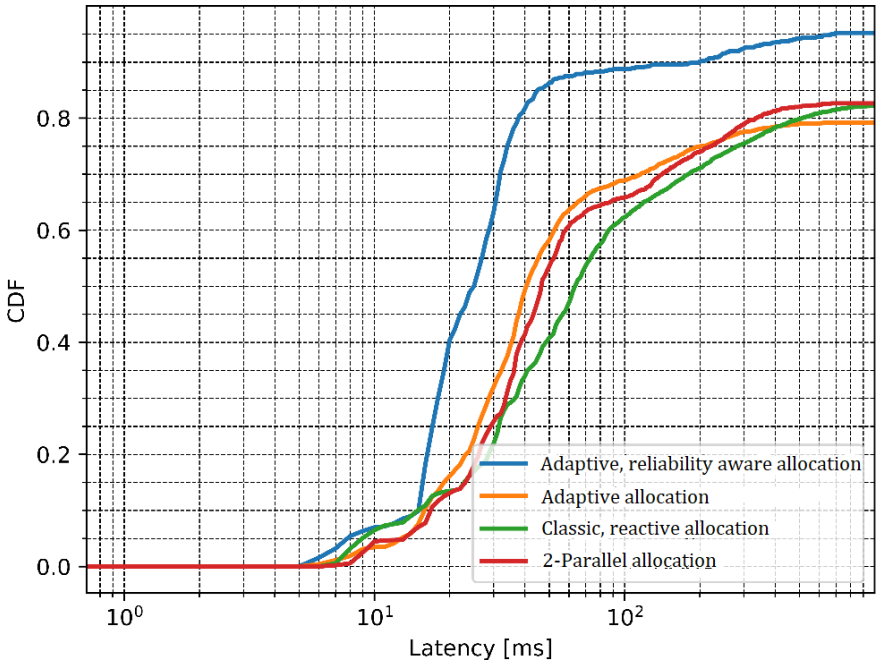

FIGURE 3.28: *Cumulative Distribution Function of RAN Latency for different HARQ schemes*

Remarks 3.4.1

In this work, we successfully validate our proposed algorithms (adaptive allocation and adaptive, reliability-aware), which have been demonstrated with System Level Simulation (SLS) (Network Simulator NS-3 (NS3)), using OpenAirInterface (OAI) hardware experimentation. Our performance results compare several allocation regimes relying on HARQ-based allocation procedures: ***(i) Classical, reactive allocation, (ii) Fixed proactive allocation*** *with 2-Parallel and 5-Parallel allocations,* ***(iii) Adaptive allocation*** *where long-term reliability is considered and* ***(iv) Adaptive, reliability-aware allocation*** *where short-term reliability is taken into account. Under the real-time hardware constraints and deviations from the preliminary simulation, performance gains in terms of latency, resource efficiency and reliability are validated.*

3.5 CONCLUSIONS

In this chapter, we proposed an exploitable approach for RAN delay enhancement, guaranteed reliability in long-term (adaptive allocation)/ short term (adaptive, reliability-aware) and reasonable resource efficiency in solving optimization problem of HARQ protocol. Our proposed algorithms are based on Lyapunov's optimization, where the prior knowledge of stochastic processes with regards to channel statistics and traffic behaviours are not required. Instead, by tracing the evolution of the RLC queue which measures the tendency of IP packet arrival process, and HARQ process queue at MAC layer which implies the dynamic of channel states, we are able to design a decision maker framework which optimally allocates proactive resources for speeding up packet decoding process without overreacting. The performance of our proposed algorithms are evaluated in both system level simulation and hardware experimentation using OpenAirInterface (OAI), where complete, full stack protocols for NR interface and EPC core network are also taken into account. Our obtained results suggest that a calibration process is indispensable for selecting optimal control parameter(s). Once they are well chosen, the performance gains are obtained at both simulation level and experimentation level.

Despite all these above-mentioned features and performance gains, our propose solutions still lacks,

to some extent, the automation, scalability and adaptability. In particular, optimal control parameters are imperatively determined before the algorithms can be applied. In case of objective function changes, the calibration process needs to be redone and requires the fast adaptation of the network in terms of computing resources. Furthermore, the performance gains of our proposed algorithms under multi-user scenario, mixture of UL and DL communication which share the same resources are still mission. Then, the following questions are still opened: *(1) Can we apply the framework in multi-user scenario where a mixed setting between UL and DL are applied* and *(2) Can we adapt the control parameters on-the-fly with respect to the changes of objective functions?* These open questions unroll new perspectives to build a more comprehensive algorithm to solve these problems.

In the next chapter, we investigate how Lyapunov's optimization framework can be exploited for Uplink (UL) communications where massive number of users try to select between Grant-Based (GB) transmission scheme and Grant-Free (GF) transmission scheme in heterogeneous service requirements and dynamic environment. Also relying on Multi Agent Reinforcement Learning (MARL), each user (agent) participates to the training process together with centralized gNb, which aim at maximizing the global objective of latency, reliability and network throughput.

The technical contributions of this chapter have been validated by the following publications:

[C1] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Proactive Resource Scheduling for 5G and Beyond Ultra-Reliable Low Latency Communications,"* IEEE 95th Vehicular Technology Conference: (VTC2022-Spring, doi: 10.1109/VTC2022-Spring54318.2022.9860872.

[C2] **L. N. Dinh**, I. Labriji, M. Maman, and E. Calvanese Strinati, *"Toward URLLC with Proactive HARQ Adaptation,"* in 2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), pp. 220–225. doi: 10.1109/EuCNC/6GSummit54941.2022.9815615.

[C3] **L. N. Dinh**, R. Bertolini, M. Maman, *"Dynamic Resource Scheduling Optimization for Ultra-Reliable Low Latency Communications: From Simulation to Experimentation,"* in 2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), sept. 2022, p. 1026-1031. doi: 10.1109/PIMRC54779.2022.9977893.

CHAPTER 4

Hybrid resource management for URLLC communications

CHAPTER CONTENTS

4.1 Introduction

4.1.1 Motivations

This chapter provides a novel vision on the needs of semi-distributed solutions for effective radio resource allocation in UL communications. Traditionally, in the scenario where multi-user need radio resources for their UL communication, centralized resource allocation approaches (Grant-Based (GB)) are considered. In this centralized method, radio resource management functions are implemented at the gNb side and it adequately provides radio resources for each user that is associated with gNb. However, this design no longer meets today's low latency requirements in Ultra Reliable and Low Latency Communications (URLLC) due to the long handshaking procedures each time an user requests radio resources from the access point. In order to tackle with this problem, decentralized approaches (Grant-Free (GF)) are proposed in which the lengthy procedure establishments is not necessary, thus reduce the latency. However, the missing of arbitrator in managing shared radio resources causes the non-negligible collision between users acquiring the same resources and influence the reliability. In 5G and beyond networks where the heterogeneous services with various QoS requirements are coexisting, the decentralized resource allocation problem is even more complex. Currently, there are several methods which involve the reservation of dedicated radio resources while opportunistically using the shared spectrum, for example: transmission without grant, preemption of radio resources for immediate use (e.g., mini-slot preemption), semi-distributed allocation (e.g., shared resource pool of Device to Device communications), or overlapping transmissions (e.g., Non-orthogonal multiple access). In this work, we propose to enhance Ultra Reliable and Low Latency Communications (URLLC) performance in terms of latency, reliability and network throughput by hybrid GB/GF architecture.

By using Multi Agent Reinforcement Learning (MARL) in hybrid GF/GB resource allocation, we offload resource allocation function to the each user (agent) without requiring global observation. The ultimate objective is to make use of the advantages of each allocation schemes to guarantee the low latency, high reliability and high throughput in URLLC communication.

4.1.2 Related works

An hybrid Grant-Free (GF)/Grant-Based (GB) resource allocation regime was discussed by Zhou et al. **[87]** and showed an improvement with up to 70% reduction in resource inefficient utilization when compared to the conservative transmission scheme. Based on each UE's channel condition and their recorded activities, an optimal resource allocation was proposed to determine the amount of resources for each allocation mode (grant-based and grant-free). dynamic resource allocation framework is missing and authors assumed the instantaneous global knowledge of UEs at gNb's side. Nomeir et al. in **[88]** considered an hybrid resource allocation scheme dealing with heterogeneous services (URLLC/eMBB) and proposed combinatorial allocation framework in which eMBB traffic is managed by GB scheduling and GF is adopted for URLLC traffic. Their schemes are less complex compared to state of the art solutions while achieving near-optimal performance. However, the exact delay calculation in a full stack system is omitted and they assumed the perfect knowledge of users at gNb which is impractical in the real system. Huang et al. **[89]** proposed a Reinforcement Learning (RL) framework to jointly optimize the communication delay and energy consumption under a high URLLC load. In their work, they considered both hybrid spectrum access of licensed and unlicensed mmWave band and used the policy gradient method to update the approximate policy that then helps to achieve optimal results. In this work, RL is employed at the gNb to understand the overall knowledge and behaviours of attaching UEs. Thus the larger the number of UEs, the more complex the centralized decision at gNb and the greater the computational resources required.

Liang et al. **[90]** investigated resource sharing as a Multi Agent Reinforcement Learning (MARL) problem. Under coexistence of Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V) connections

in a limited spectrum, each vehicle (agent) must reuse shared resources in a way that minimizes interference while improving payload delivery rate and network capacity performance. The training procedure relies on deep Q-learning with experience replay to help each agent learn action-value functions and obtain its optimal function. They showed that cooperation amongst agents is beneficial for efficient decision making on the shared resources pool. Menes et al. **[91]** presented an Deep Neural Network (DNN) approach that help each agent to predict spectrum occupation of unknown neighboring nodes in a multi-agent setting. Based on offline training, they show a reduction in the number of collisions and an increase of throughput. Naparstek et al. in **[92]** deal with dynamic and multi-user spectrum access based on Aloha-based protocol. The training process is offline and a Long Short Term Memory (LSTM) is used to aggregate the observations which are partially observable in each user. It will help each user to have a better estimation of the state in time. Yang et al. in **[93]** leveraged transfer learning and cooperative learning mechanisms to enable collaboratively distributed access management in guaranteeing URLLC's requirements. Their proposed MARL consists of a centralized training procedure and distributed cooperative implementation procedure. The objective of this framework is to perform energy-efficient channel assignment and guaranteeing URLLC's QoS. Azari et al. **[94]** proposed an interesting distributed risk-aware ML for the coexistence of scheduled, non-scheduled (urgent) URLLC traffic. Their ML solution for Radio Resource Management (RRM) resulted in the growth of data rate (75%) for scheduled traffic and 99.99 % reliability is guaranteed for both scheduled and non-scheduled URLLC users. They conclude that multiple QoS requirements in URLLC demand a novel, scalable and distributed learning approach, thus a distributed learning might be a potential candidate when dealing with massive UL access in cooperation/competitive way.

4.1.3 Contributions

There are not many studies that use MARL framework on the hybrid grant-based/grant-free scheduling with the aim of guaranteeing the QoS of massive UL URLLC such as latency, reliability and throughput. The advantage of GB scheduling is the guaranteed network throughput under a particular scheduling policy at the cost of handshaking latency, but the network throughput and latency under GF is not ensured due to the unexpected collision. Our global optimization is based on distributed decision making. Each agent does not know the behaviour of global network and makes the decision based only on its local observation. Our contributions are (i) the MARL framework on hybrid GB/GF decision, (ii) maximizing the objective function of system (i.e. latency, reliability and throughput), (iii) cooperative distributed training, which means that the training process will take place both at users and gNb and (iv) online decision making. Performance is evaluated through the combination of the MARL framework and full protocols in NS3 with different traffic profiles (i.e. predictable and unpredictable traffic pattern) that can lead to inefficient learning and thus sub-optimal results. The content of this chapter is adapted from our submitted conference paper

[C4] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Hybrid Radio Resource Management based on Multi-Agent Reinforcement Learning,"* accepted in 2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Gothenburg, Jun. 2023.

4.2 Problem Formulation

4.2.1 System Models

The system consists of a single gNb and N UEs. Each UE_i is d_i away from the gNb and generates traffic which follow Poisson process. Let $\mathcal{N}$ be the set of UEs. $A_i(t), D_{1,i}(t)$ are the total packet size [Byte] arriving or departing from $Q_{1,i}$ of agent i at time slot t, respectively. The total bandwidth is

composed of RB_t resource blocks, divided into RB_{GB} for dedicated resources (i.e. formed by N_{GB} OFDM symbols) and RB_{GF} for shared resources (i.e. formed by N_{GF} OFDM symbols). Each UE_i considers 3 queues: $Q_{1,i}$ represents the data packets in the RLC buffer, $Q_{2,i}$, $Q_{3,i}$ show the data packets scheduled for the dedicated allocation and shared allocation, respectively. The dynamic queues in each agent i can be expressed as follows:

$$Q_{1,i}(t+1) = \max\{Q_{1,i}(t) - D_{1,i}(t), 0\} + A_i(t) \tag{4.1}$$
$$Q_{2,i}(t+1) = \max\{Q_{2,i}(t) - \mathbb{1}_i.D_{2,i}(t), 0\} + A_{2,i}(t) \tag{4.2}$$
$$Q_{3,i}(t+1) = \max\{Q_{3,i}(t) - \mathbb{1}_i.D_{3,i}(t), 0\} + A_{3,i}(t) \tag{4.3}$$

where: $A_{2,i}(t) = p_i \times D_{1,i}(t)$, $A_{3,i}(t) = q_i \times D_{1,i}(t)$ are the size of packets in Bytes that start from $Q_{1,i}$ and are sent to $Q_{2,i}$ and $Q_{3,i}$ of agent i, respectively. p_i and q_i (or $p_i^{\pi_i}$ and $q_i^{\pi_i}$) are the probability that the transport blocks are sent to $Q_{2,i}$ and $Q_{3,i}$ according to policy π_i. If the transport blocks are placed in $Q_{2,i}$, the gNb schedules them according to a centralized scheduling policy π_0. Indeed, a quantity equivalent to $D_{2,i}$ (or $D_{2,i}^{\pi_0}$) Bytes will leave $Q_{2,i}$ if agent i is scheduled by gNb. On the other hand, if the transport blocks are queued at $Q_{3,i}$, each agent i will opportunistically perform resource selection from the shared and competitive resource pool and $D_{3,i}(t)$ (or $D_{3,i}^{\pi_i}(t)$) Bytes will be removed from $Q_{3,i}(t)$. In both cases, an indicator function $\mathbb{1}$ shows that the packet transmission was successful and that the receiver was able to transfer it to the upper layer. Thanks to the ACK message, $D_{2,i}$ and $D_{3,i}$ will be deducted from $Q_{2,i}(t)$ and $Q_{3,i}(t)$, respectively. The details on hybrid radio resource management scheme is displayed in Figure 4.1.

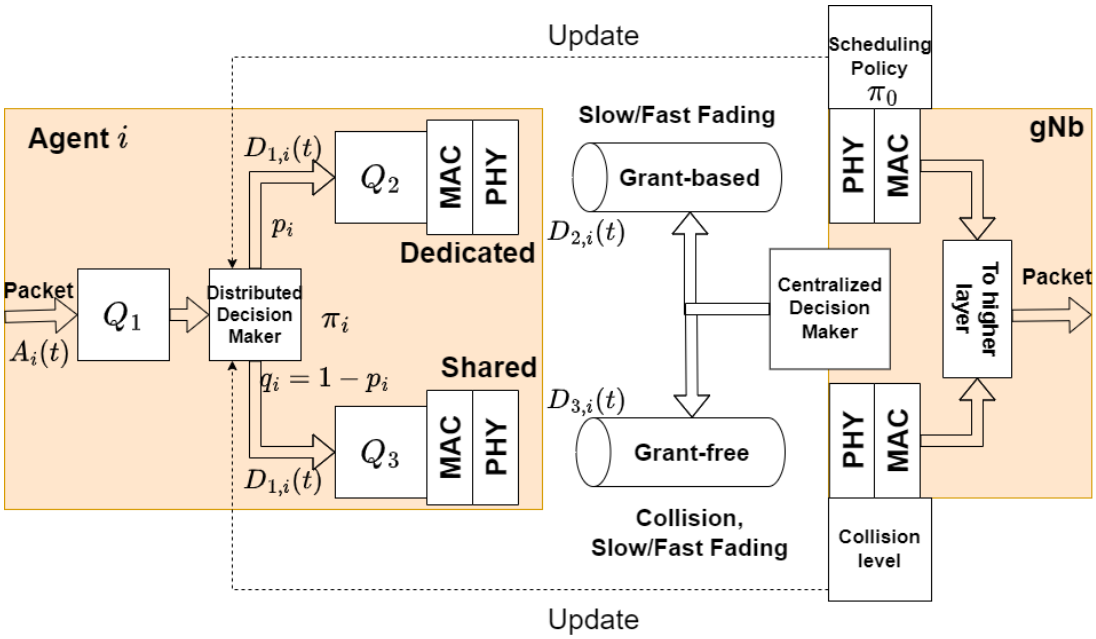

FIGURE 4.1: *Hybrid radio resource management scheme*

4.2.2 HYBRID RADIO RESOURCE MANAGEMENT PROTOCOLS

In this work, total resources in terms of total bandwidth and OFDM symbols are divided into 2 parts: scheduled resources (grant-based) and shared resources (grant-free) as the illustration in Figure 4.2. At first, the partition between scheduled resources and shared resources is defined by π_0 (i.e. ρ_{π_0}) and is fixed at the beginning.

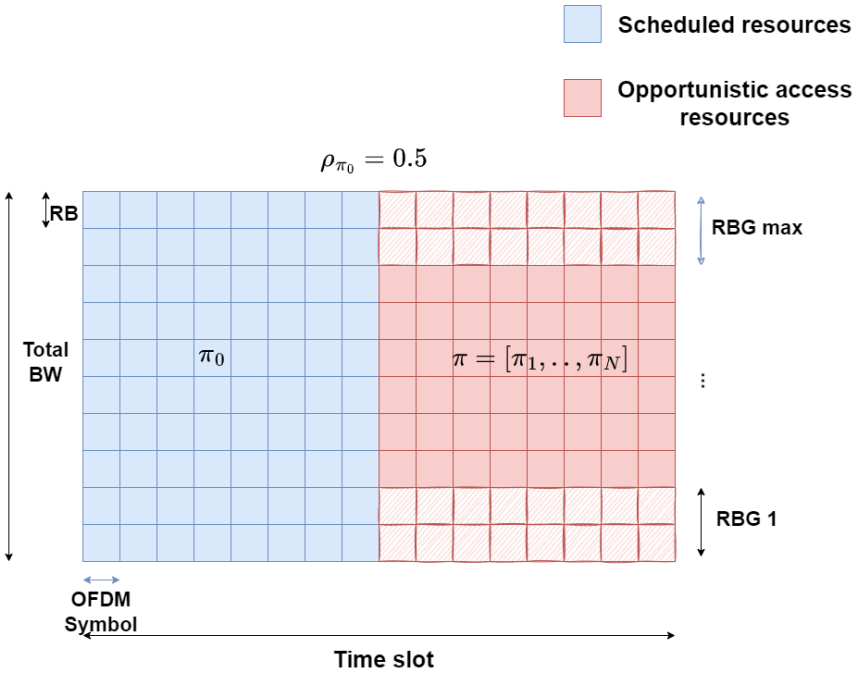

FIGURE 4.2: *Resource partition for hybrid GB/GF access.*

In our hybrid allocation regime, centralized policy π_0 manages scheduled resources in grant-based manner. Each agent, who wants to access the scheduled resources, has to perform a 5-step (grant-based) procedure as follows:

- Step 1: Each agent i sends the status of its RLC queue $Q_{1,i}$ in the Scheduling Request (SR).
- Step 2: Upon SR reception, gNb sends Signaling Grant (SG) accompanied with few resources for requested UEs.
- Step 3: After receiving the SG, each agent sends its Buffer Status Report (BSR) to the attached gNb.
- Step 4: At this step, the gNb has a global view of the amount of pending data for each user. According to the predetermined total scheduled resources, resources are allocated to corresponding users following the scheduling policy π_0 (e.g, Round-Robin, Proportional-Fair).
- Step 5: User sends both data and BSR based on allocated resources. The next scheduling phase begins in step 4 until there is no more data queued in the user.

Hence, GB scheduling is centralized and managed by gNb which has global view of user activity. The advantage is the guarantee of allocated resources and collision-free communication. However, the long handshaking procedure (5-step) cause high delay and may not be suitable for URLLC.

Meanwhile, shared resources are opened for opportunistic access of every user such that collisions between users are minimised under decentralized policy π_i. Opportunistic access resources are equally divided into multiple RBGs in frequency domain (from RBG 1 to RBG max). Based on slotted ALOHA method and decentralized policy of each agent i, π_i, each user selects resources in such way that collision (i.e. more than two users choosing the same resource) is minimal. Then, the group of local policy at each user i is formed as $\boldsymbol{\pi} = [\pi_1, ..., \pi_N]$. By doing so, we can overcome the high delay caused by 5-step handshaking. On the other hand, resource allocation will no longer be managed between users and each communication will face a non-negligible collision probability.

4.2.3 Objective Function

The average delay is proportional to the average queue length, so we can formulate the minimization of global latency as the maximization of the function $d(t)$.

$$\begin{aligned} d(t) &= \lim_{T\to\infty} \frac{1}{T} \sum_{t}^{T} \sum_{i=1}^{N} -\mathbb{E}_{\pi} \left[Q_{1,i}(t) + Q_{2,i}(t) + Q_{3,i}(t) \right] \\ &= \lim_{T\to\infty} \frac{1}{T} \sum_{t}^{T} \sum_{i=1}^{N} \sum_{j=1}^{3} -\mathbb{E}_{\pi} \left[Q_{j,i}(t) \right] \end{aligned} \tag{4.4}$$

In this case, $\pi = [\pi_1, .., \pi_N]$ is global policy and it contains the policy of each agent i, i.e π_i.

The throughput of each agent i is measured by the total time average of transport blocks in bytes that are sent in grant-free and grant-based channels. Next, we defined $r_i = \lim_{T\to\infty} \frac{1}{T} \sum_{t}^{T} \mathbb{E}_{\pi_i}[D_{2,i}(t) + D_{3,i}(t)]$. Maximizing the UL throughput of the network is equivalent to maximizing the total sum-rate $e(t) = \sum_{i=1}^{N} \mathbb{U}_i(r_i)$ where $\mathbb{U}_i(.)$ is a non-decreasing and concave utility function.

To trade-off the maximization of the average delay with the network throughput, we introduce the control parameter $\nu \geq 0$ to form a weighted-sum objective function $f_{\pi}(t)$. If ν is very large, we put more attention to the network sum-rate optimization rather than network delay minimization and vice versa. Our global objective function that jointly optimizes latency and throughput is then derived as follows:

$$f_{\pi}(t) = d(t) + \nu e(t) \tag{4.5}$$

Instead of maximizing the objective function $f_{\pi}(t)$, we will maximize its lower bound function (i.e $g_{\pi}(t) \leq f_{\pi}(t)$). It is achieved based on the fact that the departure rate of the packets from the Q_2 and Q_3 (i.e. r_i) should higher than their arrival rate z_i ($z_i = \lim_{T\to\infty} \frac{1}{T} \sum_{t}^{T} \mathbb{E}_{\pi_i}[A_{2,i}(t) + A_{3,i}(t)]$) to stabilise the queue dynamics.

$$\begin{aligned} g_{\pi}(t) &= d(t) + \nu \sum_{i=1}^{N} \mathbb{U}_i\{z_i\} \\ &= \lim_{T\to\infty} \frac{1}{T} \sum_{t}^{T} \sum_{i=1}^{N} \sum_{j=1}^{3} \nu \mathbb{E}_{\pi}[\mathbb{U}_i(z_i(t))] - \mathbb{E}_{\pi}[Q_{j,i}(t)] \end{aligned} \tag{4.6}$$

Proof.

In order to guarantee both $Q_{2,i}$ and $Q_{3,i}$ of each user i are mean rate stable, the mean arrival rate to both queues must be inferior than their departure rate.

$$z_i^{\pi_i} = z_i \leq r_i \tag{4.7}$$

Also, we assumed that $\mathbb{U}_i(.)$ is an increasing and concave utility function, thus:

$$\mathbb{U}_i(z_i) \leq \mathbb{U}_i(r_i) \tag{4.8}$$

Therefore, our designed new objective function $g_{\pi}(t)$ is the lower bound of the initial objective function $f_{\pi}(t)$. By maximally optimizing $g_{\pi}(t)$, we can bring the sub-optimal solution for our opening problem. ■

Without loss of generality, a negative Lyapunov drift-term $NLD(t)$ is added with no impact on the overall problem because the solved optimal solution pushes the queues to a minimal congested state.

$$NLD(t) = \mathbb{E}_{\pi}[-\nu_1(Q_{1,i}^2(t+1) - Q_{1,i}^2(t)) - \nu_2(\sum_{j=2}^{3} Q_{j,i}^2(t+1) - Q_{j,i}^2(t))] \tag{4.9}$$

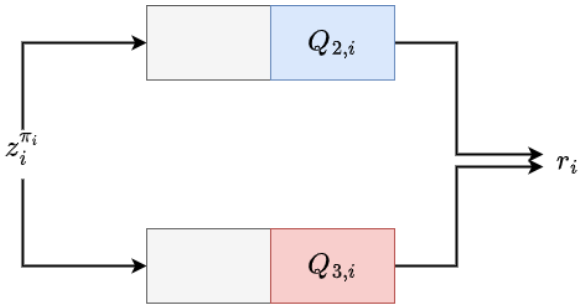

FIGURE 4.3: *Average departure/ arrival rate at each user i.*

Thus, if all the queues are stable, this add-on will converge to 0 as t goes to infinity. Then, without changing the optimal solution, the objective function becomes $h_\pi(t)$.

$$\begin{aligned} h_\pi(t) = \lim_{T\to\infty} \frac{1}{T} \sum_{t}^{T} \sum_{i=1}^{N} \Big(\sum_{j=1}^{3} \mathbb{E}_\pi[-Q_{j,i}(t)] \\ + \mathbb{E}_\pi \Big[\nu \mathbb{U}_i \left(z_i(t)\right) - \nu_1 \left(Q_{1,i}^2(t+1) - Q_{1,i}^2(t)\right) \\ - \nu_2 \sum_{j=2}^{3} \Big(Q_{j,i}^2(t+1) - Q_{j,i}^2(t) \Big) \Big] \Big) \end{aligned} \tag{4.10}$$

Where $\nu, \nu_1, \nu_2 > 0$.

Finally, we transform Equation (4.10) into a discounted dynamic programming problem with discount factor $0 \leq \gamma \leq 1$. We show that the optimal policy of this problem can be approximated by the policy of original average problem when γ is close to 1 **[95]**. The reward function in the MARL framework is similar to the function to be optimised.

$$\begin{aligned} h_\pi^\gamma(t) = \lim_{T\to\infty} \frac{1}{T} \sum_{t}^{T} \gamma^t \sum_{i=1}^{N} \Big(\sum_{j=1}^{3} \mathbb{E}_\pi[-Q_{j,i}(t)] \\ + \mathbb{E}_\pi \Big[\nu \mathbb{U}_i \left(z_i(t)\right) - \nu_1 \left(Q_{1,i}^2(t+1) - Q_{1,i}^2(t)\right) \\ - \nu_2 \sum_{j=2}^{3} \Big(Q_{j,i}^2(t+1) - Q_{j,i}^2(t) \Big) \Big] \Big) \end{aligned} \tag{4.11}$$

Proof.

For any value of $\gamma \in [0,1]$, we will obtain the following lower bound:

$$\gamma.h(x) \leq h(x) \tag{4.12}$$

For any arbitrary function $h(x)$.

Then, $h_\pi^\gamma(t)$ is a lower bound of $h_\pi(t)$ whose value should be maximised. ■

4.2.4 Problem Formulation

Our problem can be formulated as follows:

$$
\begin{aligned}
&\underset{\pi}{\text{maximize}}\ h_{\pi}^{\gamma}(t) && (\mathcal{P})\\
&\text{s.t. } \nu, \nu_1, \nu_2 \geq 0, && (\mathcal{C}_0)\\
&\quad RB_t = RB_{GB}^{\rho_{\pi_0}} + RB_{GF}^{1-\rho_{\pi_0}} && (\mathcal{C}_1)\\
&\quad 0 \leq \rho_{\pi_0} \leq 1 && (\mathcal{C}_2)\\
&\quad RB_i = \pi_{0,i}(RB_{GB}^{\rho_{\pi_0}}) + \pi_i(RB_{GF}^{1-\rho_{\pi_0}}), \forall i && (\mathcal{C}_3)\\
&\quad \mathbb{P}_{\pi_0}[\gamma_i^{gb} \leq \gamma_t] \leq \epsilon_t, \forall i && (\mathcal{C}_4)\\
&\quad 1 - (1 - \mathbb{P}_{\pi_i}^{col})(1 - \mathbb{P}_{\pi_i}[\gamma_i^{gf} \leq \gamma_t]) \leq \epsilon_t, \forall i && (\mathcal{C}_5)\\
&\quad \overline{A_{2,i}} \leq \overline{D_{2,i}} \\
&\quad \overline{A_{3,i}} \leq \overline{D_{3,i}} && (\mathcal{C}_6)\\
&\quad \overline{A_{1,i}} \leq \overline{A_{2,i}} + \overline{A_{3,i}}
\end{aligned}
$$

The constraints ($\mathcal{C}_1$), ($\mathcal{C}_2$) and ($\mathcal{C}_3$) limit the number of resource blocks (RB) allocated to each agent i under the policy π_i or π_0. In particular, ($\mathcal{C}_{1,2}$) states that the total number of resource blocks RB_t is partitioned into GB, i.e. $RB_{GB}^{\rho_{\pi_0}}$ and GF, i.e. $RB_{GF}^{1-\rho_{\pi_0}}$. ($\mathcal{C}_2$) defines this separation, managed by the gNb under policy π_0 with the ratio ρ_{π_0}. The constraint ($\mathcal{C}_3$) reveals that the resources RB_i of each agent i can be either scheduled by policy π_0 (i.e. $\pi_{0,i}(RB_{GB}^{\rho_{\pi_0}})$) or competed under policy π_i (i.e. $\pi_i(RB_{GF}^{1-\rho_{\pi_0}})$). Then, the constraints ($\mathcal{C}_4$) and ($\mathcal{C}_5$) relate to the reliability requirements. The transport blocks will be successfully decoded at the receiver when their SINR is above a predefined target (i.e.,γ_t). Regardless of the GF channel (γ_i^{gf}) or the GB channel (γ_i^{gb}), the outage probability must be less than a target ϵ_t. In the GB channel, only fast/slow fading channel is the source of impairment, so ($\mathcal{C}_4$) guarantees a transmission error below a threshold ϵ_t under the scheduling policy π_0. In the GF channel, the collision due to uncoordinated resource selections between agents, which is characterised by $\mathbb{P}_{\pi_i}^{col}$ is also considered in addition to fast/slow fading. Thus, ($\mathcal{C}_5$) takes into account both impairments simultaneously.

The constraint ($\mathcal{C}_6$) guarantees the stability of the queues in each agent i under any policy π_i. The operator $\overline{X}$ is $\lim_{T\to\infty} \frac{1}{T}\sum_t^T \mathbb{E}_{\pi_i}[X(t)]$. According to queue theory, the time average of arriving process should be smaller than or equal to the one departing from the queue. The average amount of departure process at $Q_{2,i}(t)$ depends on the centralized scheduling policy of gNb π_0, while others depend on its decentralized policy π_i.

4.3 Uplink Grant-Based (GB) / Grant-free (GF) Protocols

As the illustration in Figure 4.2, the total resources in terms of OFDM symbols in time domain and RBs in frequency domain is divided into 2 parts: *(i)* scheduled resources under the management of centralized gNb which are accessible under 5-step procedure and *(ii)* shared resources which are accessible for every user. In centralized scheduling method, gNb receives the information related to BSR and SR of each user. Then, the allocation of resources for demanded users are based on several policy π_0 such as: RR, MR or PF. On the other hand, the immediate access to the shared resources for each user is possible under collision-prone probability. Then a local policy π_i is attached for each user (agent) to control the optimal shared resource selection. In the following, we will briefly describe the possible centralized scheduling π_0 for GB resources and access protocol for GF resources.

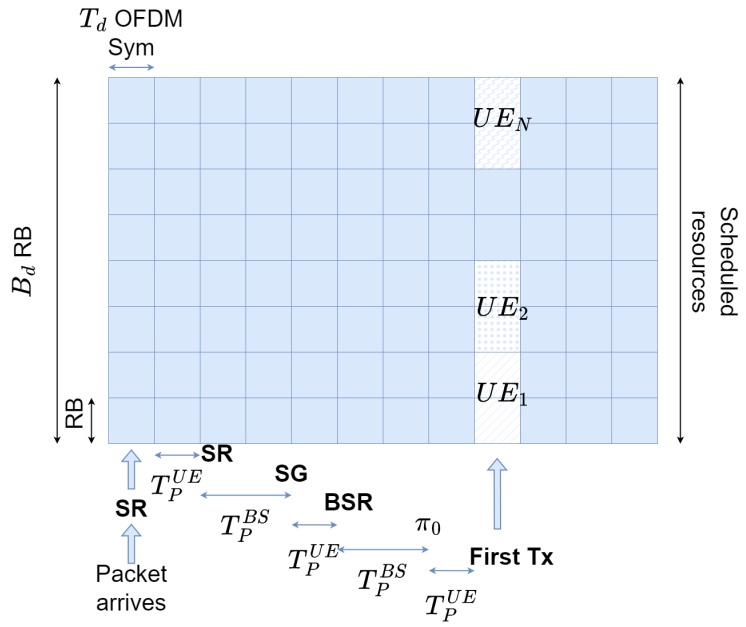

FIGURE 4.4: *Time diagram of the GB scheme*

4.3.1 CENTRALIZED SCHEDULING POLICY π_0

In our target scenario, there are no priority of access between user in GB channel, thus, for the sake of simplicity, we choose RR policy to allocate radio resources for users on demand.

4.3.1.1 ROUND ROBIN SCHEDULING POLICY

The time diagram of GB RR policy is illustrated in Figure 4.4. In this case, the dedicated Bandwidth in GB channel is divided into B_d RBs, spanning over T_d OFDM symbols and fully managed by gNb. In 5-step access, the processing delays of UE and gNb are taken into account and are represented by T_P^{UE} and T_P^{BS}, respectively.

The RR scheduler is probably the simplest scheduler found in the literature. The involvement of RR policy is in the Step 4 where gNb has access to the BSR of every users. Then, it works by equally dividing the available resources among the active flows, i.e., those logical channels which have a non-empty RLC queue. If the number of RB to be allocated for every users is greater than the number of active flows, all the flows can be allocated in the same subframe. Otherwise, if the number of active flows is greater than the number of RB (saturation), not all the flows can be scheduled in a given subframe; then, in the next subframe the allocation will start from the last flow that was not allocated. The MCS to be adopted for each user is done according to the received wide-band CQI.

4.3.1.2 RETRANSMISSION UNDER POLICY π_0

In case of imperfection channel environment, the message transmission faces to the non-zero corrupted probability (e.g. packet error probability is greater than the predefined error target), then the receiver will feedback NACK message to the sender for the packet retransmission. In this case, HARQ protocol is considered to manage the retransmission process in PHY/MAC layer.

For what concern the HARQ, RR implements the non adaptive version, which implies that in allocating the retransmission attempts RR uses the same allocation configuration of the original block, which means maintaining the same RBG and MCS. Then, UE that are allocated for HARQ retransmissions will not be allowed for the transmission of new data in case they have a transmission opportunity available

in the same TTI. It explains the fundamental trade-off between communication latency and reliability in error-prone wireless medium.

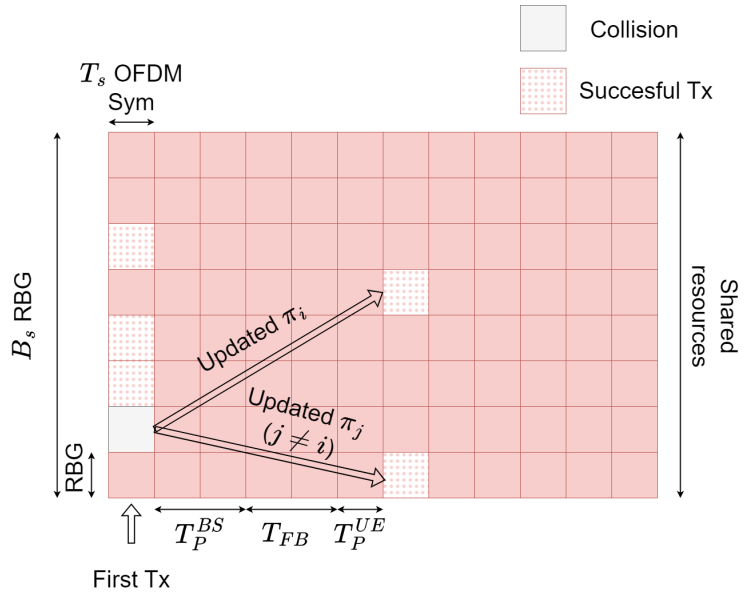

FIGURE 4.5: *Time diagram of the GF scheme*

4.3.2 OPPORTUNISTIC RESOURCE ACCESS IN *GF* CHANNEL

With opportunistic, GF channel access, UE directly selects channel resources without the coordination of centralized gNb, thus reduce the access delay due to 5-step message exchange. It can be a potential protocol candidate to support URLLC applications. However, the collision-prone transmission over shared resources is a big bottleneck when a plethora of IoT use cases with massive access is considered and significantly degrades the transmission reliability. The grant-free transmission protocol which is based on slotted-ALOHA is derived as follows:

In each communication slot, the shared Bandwidth in GF channel is equally divided into B_s RBG which are spanned over T_s shared OFDM symbol as depicted in Figure 4.5.

If UE_i sees the holding packet in its Q_3, it will opportunistically select RBG_i based on its local policy π_i for data transmission. In case of another UE_j accidentally select the same resource block group RBG_i in the shared channel, collision happens and gNb will feedback the NACK to both UE_i and UE_j after a processing delay $T_P^{BS} + T_{FB}$.

In this case, both UE_i and UE_j will respectively update their local policy to perform other resource selection procedure. Then, the upcoming opportunistic transmission will be performed after T_P^{UE} delay.

This procedure will be iterated until there is no collision or the maximum number of retransmission is reached. In the latter case, the packet is considered as lost and irretrievable.

4.4 Hybrid Resource Scheduling based Multi-agent reinforcement learning framework

4.4.1 Single-Agent Reinforcement Learning

4.4.1.1 Markov Decision Process

In single-agent system, an RL agent tries to take optimal decisions and maximises its long-term reward. The sequential decision making process of an agent in the environment is formulated as Markov Decision Process (MDP) which defines a model of transition states. Originally, an MDP is a tuple of $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma)$ where $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces, respectively; $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ denotes the transition probability from any state $s \in \mathcal{S}$ to any state $s' \in \mathcal{S}$ for any given action $a \in \mathcal{A}; \mathcal{R} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathcal{R}$ is the reward function that determines the immediate reward received by the agent for a transition from (s, a) to s'; $\gamma \in [0, 1)$ is the discount factor that trades off the instantaneous and future rewards. Each transition state in an MDP follows the Markov property which shows that the future only depends on current state and not on the history of precedent states and taken actions. Furthermore, each agent under the assumption of MDP has full observation of the states and the environment. In this case, the transition probabilities and reward distributions persist over time.

The ultimate goal of each agent is to solve MDP or to find an optimal policy to maximises the rewards over time. A policy π is defined as a mapping from underlying states s to probabilities of selecting a particular action a in the action space $\mathcal{A}$. In a episodic (i.e., finite T) or non episodic (i.e., infinite T) trajectory $\tau = (s_0, a_0, s_1, a_1, ...)$, the rewards $r = (r_0, r_1, ...)$ are obtained accordingly in every time step. The expected return will be expressed as follows:

$$\mathbb{E}_\tau \left[\sum_{t \in \tau}^{T} \gamma^t r\left(s_t, a_t, s_{t+1}\right) | a_t \sim \pi(.|s_t), s_0 \right] \tag{4.13}$$

Where discount factor γ controls how much intention we put on the future rewards (γ closes to 1) and immediate rewards (γ closes to 0).

Based on discounted cumulative reward function in Equation 4.13, under a given policy π, we can define the value function (i.e., $V_\pi(s)$) and state-action value function (namely Q-function, $Q_\pi(s, a)$) as follows:

$$V_\pi(s) = \mathbb{E}_\pi \left[\sum_{t \in \tau}^{T} \gamma^t r\left(s_t, a_t, s_{t+1}\right) | s_0 = s \right] \tag{4.14}$$

$$Q_\pi(s, a) = \mathbb{E}_\pi \left[\sum_{t \in \tau}^{T} \gamma^t r\left(s_t, a_t, s_{t+1}\right) | a_0 = a, s_0 = s \right], \forall s \in \mathcal{S}, a \in \mathcal{A} \tag{4.15}$$

where $\mathbb{E}_\pi$ is the expectation under the policy π over the set of long state-action trajectories $\tau = (s_0, a_0, s_1, a_1, ...)$.

4.4.1.2 Methods to solve MDP

Given the fact that the agent learns the optimal policy by a trial-and-error process during its interaction with real world, then we can turn these gained experiences through their interaction into knowledge about the dynamic of the environment. Then, solutions for MDP can be categorised into 2 types: value-based methods and policy based methods.

Value-based methods are introduce to guide agent selecting optimal Q-function (i.e, Q^*) that maximises Equation 4.15. Then, the optimal policy π^* will be derived with respect to the greedy action as follows:

$$\pi^* = \arg\max_a Q^*(s, a) \tag{4.16}$$

In the environment where complex states and actions are involved, it is challenging for the agent to learn its Q values because its behaviours are not always linear. Instead, by using neural network, non-linear Q-function can be approximated. Mihn et al. [96] first introduced the concept of DQN to optimise the following equations:

$$\min_{\theta} \mathbb{E}_{s_t,a_t,r_t,s_{t+1}\sim\mathcal{D}}\left[\left(r_t+\gamma\max_{a} Q_\theta(s_{t+1},a)-Q_\theta(s_t,a_t)\right)^2\right] \tag{4.17}$$

where a replay buffer $\mathcal{D}$ records the previous samples of the trajectory [97], a network parameter θ is fitted by using this experience buffer in a supervised learning fashion and Q_θ is a slowly updated target network which supports training stabilization.

In terms of policy-based methods, optimal policy π^* will be learnt directly rather than greedy-action learning. Specifically, the optimal policy will be parameterized by θ (i.e, $\pi^* = \pi_\theta(.|s)$) and the update of parameters θ will be towards the direction in which cumulative rewards are maximized (i.e., $\theta \leftarrow \theta + \gamma\nabla_\theta V^{\pi_\theta}(s)$). The well-known policy gradient $\nabla_\theta V^{\pi_\theta}(s)$ is expressed as follows [98]:

$$\nabla_\theta V^{\pi_\theta}(s) = \mathbb{E}_{s\sim\mu^{\pi_\theta},a\sim\pi_\theta(.|s)}\left[\nabla_\theta \log \pi_\theta(a|s).Q^{\pi_\theta}(s,a)\right] \tag{4.18}$$

where μ^{π_θ} is introduced as the measured state under policy π_θ and $\nabla_\theta \log \pi_\theta(a|s)$ is the updating scores of the target policy.

Deterministic Policy Gradient (DPG) theorem is introduced by Silver et al. [99] to take the deterministic policy and continuous actions into account.

$$\nabla_\theta V^{\pi_\theta}(s) = \mathbb{E}_{s\sim\mu^{\pi_\theta}}\left[\nabla_\theta \pi_\theta(a|s).\nabla_a Q^{\pi_\theta}(s,a)|_{a=\pi_\theta(s)}\right] \tag{4.19}$$

4.4.2 Multi-Agent Reinforcement Learning

4.4.2.1 Markov Game

Multi Agent Reinforcement Learning (MARL) refers to the interactions of multiple autonomous agents within the same environment to learn how to achieve their individual/global objectives. In single agent RL problem, the understanding of MDP is beneficial to model optimal decision-making process in stochastic environment. Nevertheless, different representations are required in multi-agent system when each agent's joint action changes state dynamic and distribution of rewards. Thus, the decision-making process that considers many agents is usually modelled as Markov game [100].

In multi-agent system, at each time slot t, each agent $i \in N$ observes a state s_t^i locally or globally and select an action $a_t^i \sim \pi_i(a_t^i|s_t^i)$. The payoffs (possibly, not intermediate) which agent i receives is r_t^i. Each agent i aims to maximize its own total expected return $R_i = \sum^T \gamma^t * r_i^t$. A Markov game in multi agent system is defined by a tuple $(\mathcal{N}, \mathcal{S}, \{\mathcal{A}^i\}_{i\in\mathcal{N}}, \mathcal{T}, \{\mathcal{R}^i\}_{i\in N}, \gamma)$, where:

- $\mathcal{N}$ = 1, $\cdots$, N denotes the set of $N > 1$ agents.
- $\mathcal{S}$ denotes the state space observed by all agents.
- $\mathcal{A}^i$ denotes the action space of agent i. Let $\mathcal{A} := \mathcal{A}^1 \times ... \times \mathcal{A}^N$ is the joint action space of agents in the system.
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ denotes the transition probability from any state $s \in \mathcal{S}$ to any state $s' \in \mathcal{S}$ for any joint action $a \in \mathcal{N}$.
- $\mathcal{R}^i : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the reward function that determines the immediate reward received by agent i for a transition from (s, a) to s'
- $\gamma \in [0, 1)$ is the discount factor.

At time t, each agent $i \in \mathcal{N}$ executes an action a_t^i according to system state s_t. The system then transition to state s_{t+1}, and rewards each agent i by $R^i(s_t, a_t, s_{t+1})$. The goal of agent i is to optimize its

own long term reward, by finding the policy $\pi^i : \mathcal{S} \to \Delta(\mathcal{A}^i)$ such that $a_t^i \sim \pi^i(.|s_t)$. As a consequence, the value function $V^i : \mathcal{S} \to \mathbb{R}$ of agent i becomes a function of the joint policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ defined as $\pi(a|s) = \prod_{i \in \mathcal{N}} \pi^i(a^i|s)$. In particular, for any joint policy π and state $s \in \mathcal{S}$, the value function (i.e, $V^i_{\pi^i, \pi^{-i}}(s)$) can be expressed as follows:

$$V^i_{\pi^i, \pi^{-i}}(s) := \mathbb{E}\left[\sum_{t \geq 0} \gamma^t R^i(s_t, a_t, s_{t+1}) | a_t^i \sim \pi^i(.|s_t), s_0 = s\right] \tag{4.20}$$

where $-i$ represents the indices of all agents in $\mathcal{N}$ except agent i. Thus, the solution for MDP relies not only on a single agent, but also the choices of all other agents of the game.

4.4.2.2 Methods to solve Markov game problems

The solutions for Markov Game is also based on MDP solution for single agent: value-based MARL and policy-based MARL.

In value-based MARL, the update of Q-learning values has been modified considering the environment states of multi-agents.

$$Q^i(s_t, a_t) \leftarrow Q^i(s_t, a_t) + \gamma . \left(r_t^i + \gamma . \textbf{eval}^i \left(\{Q^i(s_{t+1}, .)\}\right) - Q^i(s_t, a_t)\right) \tag{4.21}$$

where $\textbf{eval}^i \left(\{Q^i(s_{t+1}, .)\}\right)$ represents the evaluation of agent i at time step $t+1$ considering other agent's behaviours.

Regarding the policy-based MARL, each agent learns its own policy $\pi^i_{\theta^i}$ by updating the parameter θ^i. Then, we will assume $\boldsymbol{\theta} = (\theta^i)_{i \in \mathcal{N}}$ as the collection of policy parameters for all agents and $\boldsymbol{\pi_\theta} = \prod_{i \in \mathcal{N}} \pi^i_{\theta^i}(a^i|s)$ is the collective policy. In order for each agent i to learn its optimal policy, the policy gradient in Equation 4.18 can be extended as follows:

$$\nabla_{\theta^i} J^i(\boldsymbol{\theta}) = \mathbb{E}_{s \sim \mu^{\boldsymbol{\pi_\theta}}, a \sim \pi_\theta(.|s)} \left[\nabla_{\theta_i} \log \pi_i(a^i|s) . Q^{i, \boldsymbol{\pi_\theta}}(s, a_1, ..., a_N)\right] \tag{4.22}$$

where $Q_i^\pi(s, a_1, ..., a_N)$ is the evaluated Q-function of agent i at a particular state s given the action sets of other agents $(a_1, ..., a_N)$.

If continuous action sets with deterministic policy are considered, Lowe et al. **[101]** proposed MADDPG update which is written as follows:

$$\nabla_{\theta^i} J^i(\boldsymbol{\theta}) = \mathbb{E}_{s \sim \mu^{\boldsymbol{\pi_\theta}}} \left[\nabla_{\theta^i} \log \pi^i_\theta(a^i|s) . \nabla_{a^i} Q^{i, \boldsymbol{\pi_\theta}}|_{a_i = \pi^i_{\theta^i}(s)}\right] \tag{4.23}$$

4.4.2.3 Observability in multi-agent reinforcement learning

In multi-agent setting, each agent participates to the optimal decision making process and requires novel representations of MDP which are modelled depending on full or partial visibility of them. When each agent has full observability of the transition state, our problem will be formed as represented by a Markov game. In case of all agents can not observe the state $s \in \mathcal{S}$ but they share the same reward function $\mathcal{R}$, it is represented by Dec-POMDP. Finally, POMG assumes that agents do not have global access to the environmental state but only an observation of the state through an local observation of each individual agent i **[102]**. Figure 4.6 shows the different observability in multi-agent system.

In Dec-POMDP, a tuple of $(\mathcal{N}, \mathcal{S}, \{\mathcal{A}^i\}_{i \in \mathcal{N}}, \{\Omega^i\}_{i \in \mathcal{N}}, \mathcal{O}, \mathcal{T}, \{\mathcal{R}^i\}_{i \in \mathcal{N}})$ is defined. In additional to the Markov game's tuple, joint observation set $\Omega = [\Omega_1, ..., \Omega_N]$ and the observation probability function $\mathcal{O} : \Omega \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ are included. The former contains the set of observation for each agent i (i.e Ω_i), the latter is the probability of agents seeing observations $o \in \mathcal{O}$, given the state is $s \in \mathcal{S}$ and agents take actions $a \in \mathcal{A}$. Then a common observation $\mathcal{O}(o_1, .., a_N | a_1, ..., a_N, s')$ are observed by N agents given each action tuple $(a_1, ..., a_N)$ was chosen and transition state s' is recorded. At every time step, each

agent takes an action and receives a local observation corresponding to the state and receives a joint reward. Then, a local policy in each agent will map local histories of observations to optimal actions. However, computation complexity is the main drawback of Dec-POMDP when a massive number of agents and highly dynamic environments are considered **[103]**. Then, a centralized controller to collect all the agent's private information is essential to reduce the noise of knowledge collection and reach global minimum solution. It is known as the "centralized-training and decentralized execution" fashion **[104]**. In wireless system where multiple UEs are attached to the same gNb, this scheme is relevant for the applications of MARL in achieving performance convergence.

With regards to POMG, it is considered as the counterpart of the Dec-POMDP where each agent tries to maximise their individual reward functions in a partially observable environment. In this case, it is formalized as a tuple of $(\mathcal{N}, \mathcal{S}, \{\mathcal{A}^i\}_{i\in\mathcal{N}}, \mathcal{O}, \mathcal{T}, \{\mathcal{R}^i\}_{i\in\mathcal{N}})$. In this case, $\mathcal{O} = \mathbb{O}^1 \times ... \times \mathbb{O}^N$ is the joint observation set for each of agent $i \in \mathcal{N}$. An observation function $O(\boldsymbol{o}|\boldsymbol{a}, s')$ represents the probability of observing $\boldsymbol{o}$ given the action $\boldsymbol{a} \in \mathcal{A}$ and state $s' \in \mathcal{S}$.

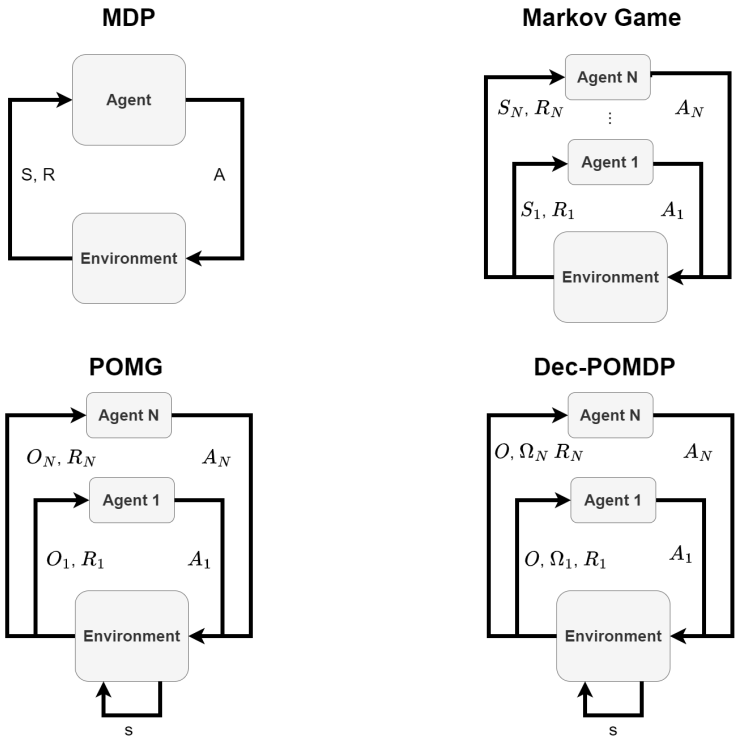

FIGURE 4.6: *Observability in multi-agent reinforcement learning*

4.4.3 PROPOSED SOLUTIONS

In this chapter, we design MARL algorithms to solve the problem $\mathcal{P}$. Two RL approaches to learn an optimal policy are compared: Multi-agent Deep-Q Learning (MADQL - value-based) and Multi-agent Deterministic Policy Gradient (MADDPG - policy-based). The former algorithm tries to find global optimum based on local observation of each agent (POMG) in a fully distributed manner. On the other hand, the latter algorithm is based on centralized training-decentralized execution (actor-critic) manner where centralized critic justifies the action value of each agent and update their weights for decentralized, individual agent (semi-distributed architecture).

The State/Observation space $\boldsymbol{s_i}$ for each agent i, at time slot t has to consider some information about its queues, $\boldsymbol{Q_i(t)} = \{Q_{1,i}(t), Q_{2,i}(t), Q_{3,i}(t)\}$, the traffic pattern (low/high rate, aperiodic/periodic traffic) λ_i, which will be used to estimate $A_i(t)$ at particular time slot t, the scheduling policy π_0 in GB access,

which indicates how much resources in Bytes will be allocated for scheduled users and occupancy level information $\boldsymbol{occ}_i(t)$ in GF access which measures the probability of collision under policy π_i. Thus, at each time slot t, $\boldsymbol{s_i(t)} = \{\boldsymbol{Q_i(t)}, \lambda_i, \pi_0(t), \boldsymbol{occ}_i(t)\}$. Afterwards, we define a subset $\mathcal{K} \in \mathcal{N}$ of agents in which each agent $k \in \mathcal{K}$ can observe other agent's states, $\mathcal{K} = \{1, ..., K\}$. If only local observation is permitted, then $K = 1$ and each agent observes itself to make decision. If $K = N$, each agent can access to global state. The global state is formed as follows:

$$\mathcal{S}(t) = \bigcup_{k=1}^{K} \boldsymbol{s_k(t)} \tag{4.24}$$

Similarly, we can define the global action of each agent in the network as follows:

$$\mathcal{A}(t) = \bigcup_{k=1}^{K} \boldsymbol{a_k(t)} \tag{4.25}$$

where on each agent i, at time slot t, the action vector $\boldsymbol{a_i(t)}$ includes the flow control and the resource selection. First, from $Q_{1,i}(t)$, agent i transmits a quantity of $D_{1,i}(t)$ bytes to $Q_{2,i}(t)$ with probability p_i and to $Q_{3,i}(t)$ with probability q_i. Second, the data in $Q_{2,i}(t)$ will be framed as transport block for the scheduled transmission according to policy π_0 whereas in grant-free channel, each agent i will automatically select resources to serve the data in $Q_{3,i}(t)$.

The reward function quantifies how good an action is taken under a particular state. For each agent i, we define the reward under state $\boldsymbol{s_i(t)}$ and action $\boldsymbol{a_i(t)}$, $R_i(\boldsymbol{s_i(t), a_i(t)}) \in \mathbb{R}$ according to the objective function in Equation 4.10 as follows:

$$\begin{aligned} R_i(\boldsymbol{s_i(t), a_i(t)}) = &\sum_{j \in 3} -Q_{j,i}(t+1) + \nu \log(z_i(t)) \\ &- \nu_1 \left(Q_{1,i}^2(t+1) - Q_{1,i}^2(t)\right) \\ &- \nu_2 \sum_{j=2}^{3} \left(Q_{j,i}^2(t+1) - Q_{j,i}^2(t)\right) \end{aligned} \tag{4.26}$$

Accordingly, the global reward observed by gNb for the set $\mathcal{N}$, i.e $\mathcal{R}(\mathcal{S}(t), \mathcal{A}(t)) \in \mathbb{R}^N$ of agents is derived as follows:

$$\mathcal{R}(\mathcal{S}, \mathcal{A}) = \bigcup_{i=1}^{N} \boldsymbol{R_i}\left(\boldsymbol{s_i(t), a_i(t)}\right) \tag{4.27}$$

4.4.3.1 Optimal policy learning using Multi agent Deep Q networks (MADQL)

In this algorithm, each agent has its own memory play, which is stored locally. Also, we assumed that Partially Observable Markov Decision Process (POMG) is considered where each agent does not have full observation dynamic **[105]**. Under local policy π_i, the action value function of agent i, i.e., W_{π_i}-function is defined as: $W_{\pi_i}(s_i, a_i) = R_i(s_i, a_i) + \gamma W_{\pi_i}(s_i', \pi_i(s_i'))$. Where s_i and a_i are respectively current state and action of agent i which returns a corresponding rewards $R_i(s_i, a_i)$ and turns agent i into new state s_i By estimating this function, we will guide an agent in selecting the optimal action in a particular state. However, due to the complex dynamic state of the multi-agent system, it is viable to approximate their value function with the parameter ω using a neural network. The approximation of W-function will be expressed by the algorithm 2 and the value function will be updated each time slot period.

In this algorithm, we use a technique called Experience Replay **[96]** to efficiently utilize the collected samples and eliminate their correlation. Also, two separate networks are used to independently select the action and learn the value function to avoid overestimation **[98]**. As we used ω-parameterized network to evaluate the action-value function, the network which is responsible for action selection will be

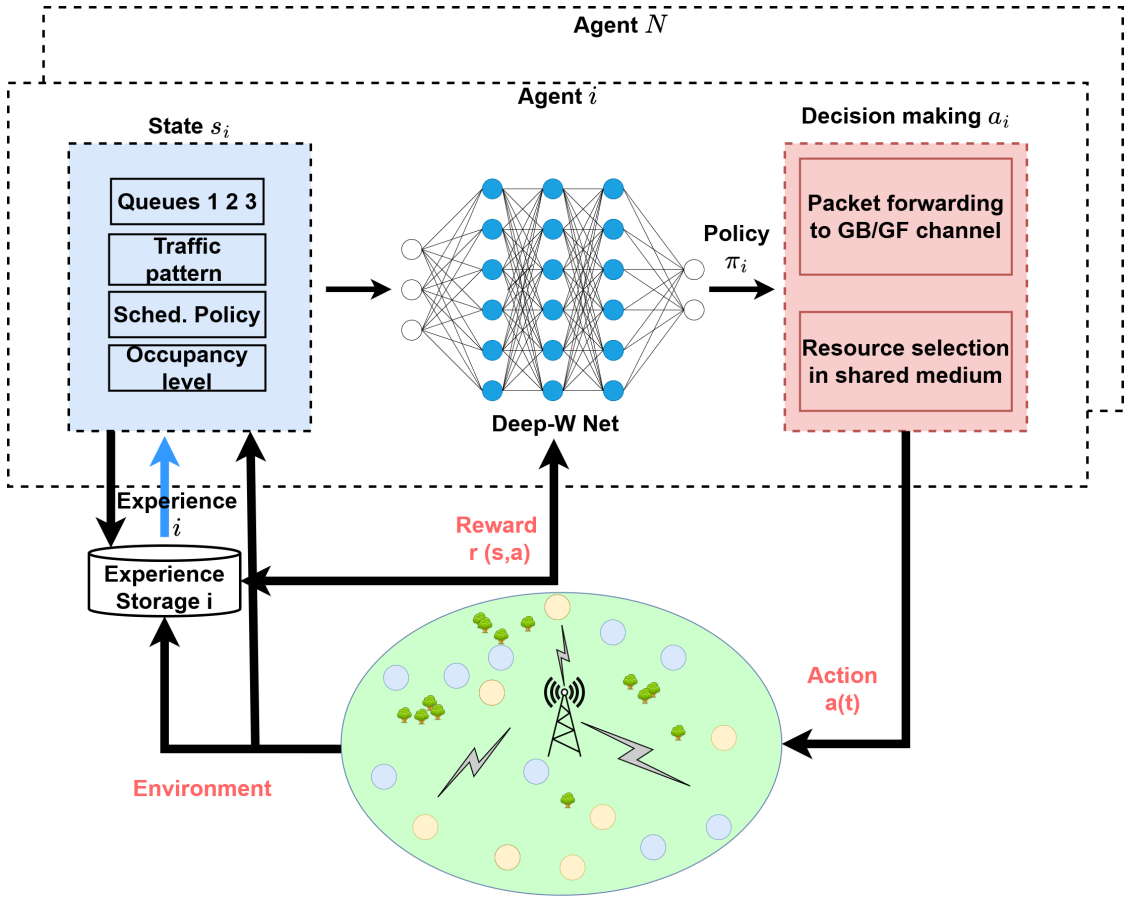

FIGURE 4.7: *DQN Multi Agent Decision Maker framework*

Algorithm 2 W(Q)-learning based on W(Q)-value

Initialization Replay memory *Mem*
Random initialization $W_{\pi_i}(s_i, a_i, \omega)$, $W_{\pi_i}(s_i, a_i, \phi)$ $\forall s_i, a_i \in \mathcal{S}, \mathcal{A}$
Define M-batch size
for *time slot t* **do**
  for *each agent i* **do**
    if ***inactive*** **then**
      continue
    end
    Given state $\boldsymbol{s_i(t)} \in S(t)$
    Select $\boldsymbol{a_i(t)}$ as the output of $\varepsilon - greedy$
    Observe next state $\boldsymbol{s_i(t+1)}$
    Store $(\boldsymbol{s_i(t), a_i(t), R_i(t), s_i(t+1)})$ to *Mem*.
    if *More than M samples are collected* **then**
      Randomly sampling mini-batch M from *Mem*
      Given $(\boldsymbol{s_i(t), a_i(t), s'_i(t+1)}) \in M$
      Calculate $y_i = \max_{a_i}[R_i(s'_i, a_i) + \gamma W_{\pi_i}(s'_i, a_i, \phi)]$
      Perform gradient descent on (learning rate α)
      $\sum_{(s,a,s',y)\in M}(y_i - W_{\pi_i}(s_i, a_i, \omega))^2$
      if *mod(t, T) == 0* **then**
        $\phi = \omega$
      end
    end
  end
end

parameterized by ϕ. When local action of each agent i is executed at time slot t, each agent will observe the next state at time slot $t+1$ and store sample $(s_i(t), a_i(t), R_i(t), s_i(t+1))$ to the buffer. The reward will be calculated accordingly because it is a function of state. After M samples collected (batch-size), we use fixed target technique which holds the target W-function parameterized with ϕ and updates the target every steps [97].

4.4.3.2 Optimal policy using Multi agent semi-distributed learning

In this section, we propose another algorithm to solve problem $\mathcal{P}$ which is based on optimal policy learning rather than action-value function learning. It turns out that value function based learning is sensitive to the high variance of the multi-state environment. Specifically, the use of W(Q)-learning in a multi-agent environment faces a challenge when policy of each agent changes over time and the environment is non stationary. Thus, the convergence of multi-agent algorithms based on Q-learning in dynamic environment is often time consuming. On the other hand, the policy gradient method often requires the coordination of several agents and leads to high training variance. In this section, we propose an algorithm based on Actor-Critic approach (Policy-Learning) that directly learns the policy leading to the optimal solution. Both the actor and the critic are approximated by neural networks. The training process is centralized at the gNb, where a centralized critic part learns the shared, global states, actions and policies of all agents. Then, each agent i attached to the gNb can obtain the training knowledge to derive its own policy in the decentralized actor part. Since we need to learn/approximate the individual policy and value function of each agent, we parameterize its policy and value function W as θ_i and w_i, respectively. And let's assume $\boldsymbol{\theta} = \{\theta_1, ..., \theta_N\}$, $\boldsymbol{\pi} = \{\pi_1, ..., \pi_N\}$. The objective of our algorithm is to maximize the reward function $\sum_i^N R_i$. The framework can then be displayed on Figure 4.8.

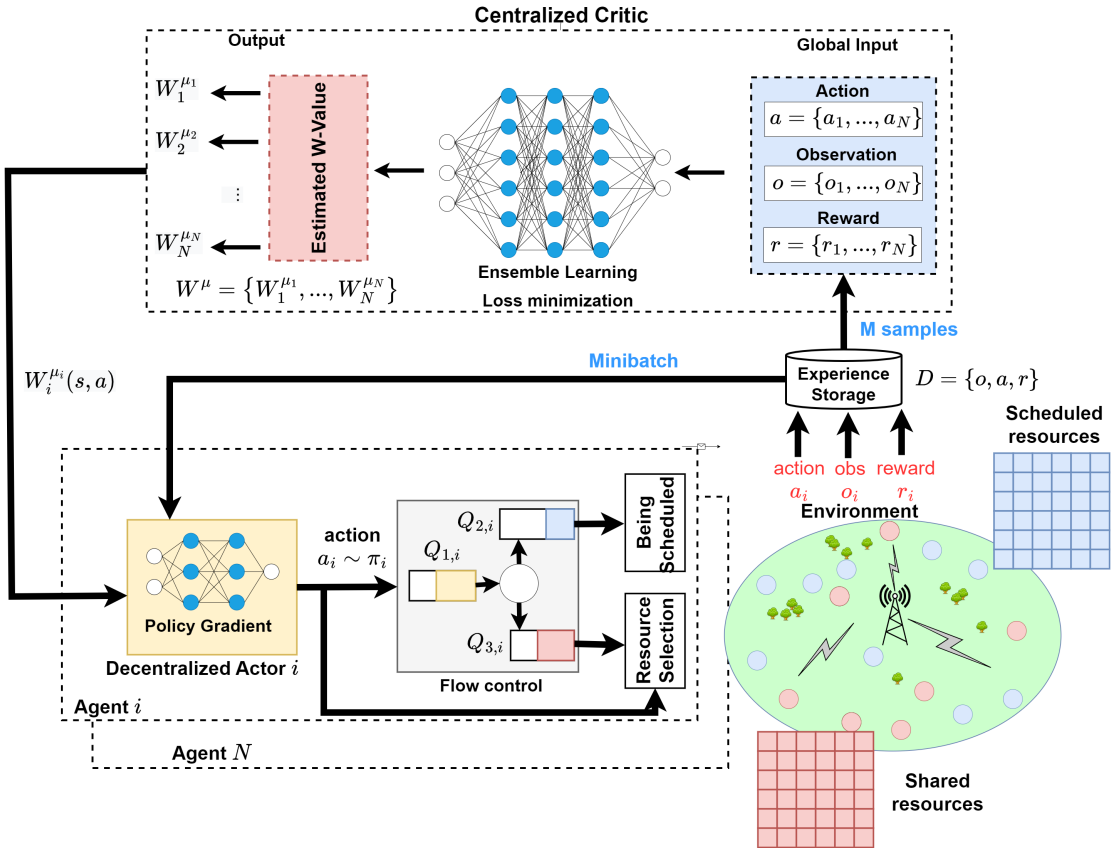

Figure 4.8: *Multi-Agent Decentralized-Actor, Centralized-Critic Decision Maker framework*

Algorithm 3 Semi-distributed Learning Algorithm

Replay memory *Mem*
Random action initialization (Exploration)
for *each time slot t* **do**
  gNb observes global action $\mathcal{A}$(t), reward $\mathcal{R}$(t) and state $\mathcal{S}$(t), $\mathcal{S}$(t+1)
  Store $(\mathcal{S}(t), \mathcal{A}(t), \mathcal{S}(t+1), \mathcal{R}(t))$ in *Mem*
  Set $\mathcal{S}(t) \leftarrow \mathcal{S}(t+1)$
  for *each agent* $i \in \mathcal{N}$ **do**
    if ***inactive*** **then**
      continue
    end
    Perform $a_i = \mu_{\theta_i}(s_i)$ according current policy π_i
    Sample M samples (S^m, A^m, R^m, S'^m) in *Mem*
    Set target $\boldsymbol{y_i^m} = R_i^m + \gamma W_i^{\mu'}(S'^m, A'^m)$
    gNb updates Critic part (Gradient descent, learning rate α_c)
    $\mathcal{L}(\theta_i) = \frac{1}{|M|}\sum_m (y_i^m - W_i^{\mu}(S^m, A^m))^2$
    Agent i updates Actor part (Policy gradient, learning rate α_a)
    $\Delta_{\theta_i} J \approx \frac{1}{|M|}\sum_m \Delta_{\theta_i}\mu_i(s_i^m)\Delta_{a_i} W_i^{\mu}(S^m, A^m)$
    Update target network parameters
    $\theta_i' \leftarrow \tau\theta_i + (1-\tau)\theta_i'$
  end
end

The motivation of this architecture is the separation of centralized critic part (embedded in gNb) which has global observation of all users, and decentralized actor part (embedded in each agent i) which has only local observations. Then, centralized critic will help us to approximate the value function $W_i^{\mu_i}(s_i(t), a_i(t))$ of agent i following its parameterized policy π^{μ_i} taking action $a_i(t)$ at state $s_i(t)$ at time slot t. This information will be sent into Agent i and it will use such information to estimate or optimize policy π^{μ_i}. centralized critic improves the estimation of state-action value learning of which decentralized actor use to improves the policy evaluation. To the end, this approach will converge to optimal policy without requiring global observation of agent i in the network. The algorithm can be expressed in Algorithm 3. In this algorithm, $\mu' = \{\mu_{\theta_i'}\}$ is the set of target policies with delayed parameters θ_i'. The approximate policy of each agent i is learned by maximizing the log probability of agent $i's$ actions. τ is a parameter for updating the target network, γ is discounted factor for the future reward awareness.

4.4.4 Results and discussion

The performance gain of our proposed algorithms will be illustrated in 2 different scenarios: *(1)* Fixed number of UEs (N=30) and various homogeneous traffics and *(2)* Different number of UEs and homogeneous traffic. In the former scenario, we demonstrate our algorithms when *(i)* only collision in shared bandwidth is considered and *(ii)* more complex scenario where both collision in shared channel and deep fading in both shared/scheduled channel are taken into account. In the latter one, we extended our studies towards the scenario where more associated agents are considered (N=30-90).

4.4.4.1 Fixed number of UEs (N=30) with various homogeneous traffics

Our network contains a single gNb and $N = 30$ UEs, all placed at the same distance from the gNb ($d = 80m$). Each user's traffic is generated using a Poisson process with a fixed packet size of 20 Bytes. After being encapsulated with a header in internet protocol and packet data convergence protocol layers, they arrive at the RLC Layer (Q_1) for transmission to the GB channel (Q_2) or the GF channel Q_3. In this

work, we consider a total Bandwidth of $50MHz$ in the Sub-6GHz band and half of it is dedicated to the GB channel ($\rho_{\pi_0} = 0.5$). The remaining BW is divided equally into 15 groups (RBGs) for opportunistic access to resources. In order to model dynamic channel and antenna model, the 3GPP Indoor-Factory scenario [44] is considered and gNb and each of agent are equipped with 4×4, 2×2 planar linear antennas, respectively. The details of simulation parameters are given in Table 4.1.

TABLE 4.1: *Simulation parameters*

Nb of Users	30
Reliability target ϵ_t	0.01
Tx Power	8 dBm
Distance	80 m
Max re-transmission K_{max}	5
Central frequency	3.61 GHz
Bandwidth	50 MHz
Scheduling Policy π_0	Round-robin
ρ_{π_0}	0.5
Shared resources in shared BW	15 RBGs
Channel Dynamics	3GPP Indoor-Factory

Concerning the hyper-parameters of learning algorithms, Table 4.2 shows their numerical values in our simulation. In semi-distributed algorithm based on MADDPG, both actor (user) and critic (gNb) have to learn its local policy and global action reward, respectively. Then, the actor/ critic learning rate (i.e. α_a and α_c) are set at 10^{-3} and 10^{-2}, accordingly. Then, the history of previous state, action, reward (experience) are stored for each user at gNb. The batch size of 64 shuffled experience are then used for learning and updating the action-value function/ local policy of each user towards the optimal one. Delay network update rate parameter $\tau = 0.1$ is used to indicate how frequently the parameterized policy is updated. As $\tau \ll 1$, the target networks have a stable and slow update ameliorating the divergence problem. With regards to the applied neural networks which approximate the action value function at both actor and critic parts, two layer of 64 fully connected nodes are used.

In W(Q)-learning (MADQL), the experience are stored locally at each agent, the learning rate for updating the value function is 2.10^{-2}. Decay exploration technique is applied to help each agent to explore their action value function. Particularly, the initial exploration rate is set as $\epsilon = 0.99$ and will decay at rate $\epsilon_d = 5e-4$ until reaching its minimum $\epsilon_m = 0.01$. The batch size (64) as well as neural network architecture (2 layer of fully connected 64 nodes) are similar to semi-decentralised learning algorithm. A discount factor of 0.99 is considered for future rewards. Besides, a period of $T = 0.1(s)$ is selected to update the target network for learning.

With regards to parameters ν, ν_1, ν_2 in the objective function $\mathcal{P}$, We will apply these hyper-parameters in 2 different scenarios where only collision is considered (retransmissions are deactivated) and both collision and channel fading happen (retransmissions are activated), respectively. In this case, the queue dynamic changes and suggests us to select different values of ν, ν_1, ν_2 . In the scenario where the network performance under collision is examined, the ν, ν_1, ν_2 values are considerably lower than in the case where both collision and fading effects are considered. It is because that we would like to further promote the good actions which result in successful transmission as well as further penalise the bad actions which cause packet corruption and failed transmissions. Then, because of the differences between the unit of

TABLE 4.2: *Hyper-parameters*

Semi-decentralized learning (MADDPG)		**W(Q)-learning (MADQL)**	
Actor learning rate α_a	10^{-2}	Learning rate α	2.10^{-2}
Critic learning rate α_c	10^{-2}	Exploration rate ε / decay rate ε_d	0.99 / 5e-4
Delay network update rate τ	0.1	Minimum exploration rate ε_m	0.01
Batch size	64	Batch size	64
Discount factor γ	0.9	Discount factor γ	0.99
Hidden layers	2	Hidden layers	2
Dimension of hidden layer	64	Dimension of hidden layer	64
		Time period T	0.1 s
ν, ν_1, ν_2 (Collision)	(1000,100,50)	ν, ν_1, ν_2 (Collision)	(1000,100,50)
ν, ν_1, ν_2 (Collision + Fading)	(10000,1000,500)	ν, ν_1, ν_2 (Collision + Fading)	(10000,1000,500)

the queues (queue size in bytes) and the sum total rate which is measured by the number of allocated resources per time unit, the values of ν, ν_1, ν_2 are not identical.

In the following, we will examine our algorithms (MADQL and MADDPG) in two different scenarios: *(i)* Only collisions are considered and *(ii)* collisions and fading effects are considered .

Performance under collisions

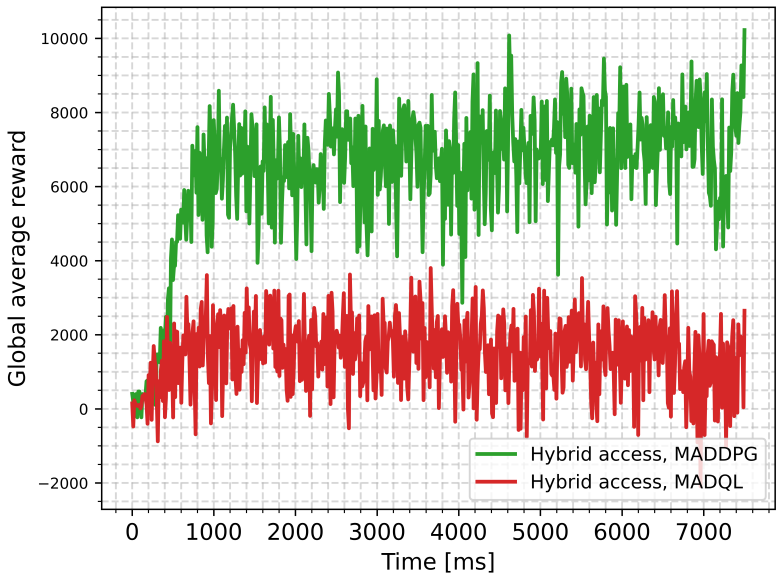

FIGURE 4.9: *Reward convergence for MADDPG and MADQL at load=0.5Mbps (only collision).*

Figure 4.9 compares the obtained rewards between MADQL-based and MADDPG-based algorithm when each user has 0.5 Mbps traffic load. It shows the convergence of both algorithms and better average rewards are obtained with MADDPG algorithm. In practice, we observed that MADDPG algorithm either safely selects shared resources or rather following GB resource access for guaranteed resource allocation. Thus, the error-prone transmission is minimised and resulted in better rewards. On the other

hand, MADQL algorithms put more favours in opportunistic shared resource access to minimize the transmission latency. It leads to the frequent collision between decentralized users and penalizes the obtained rewards.

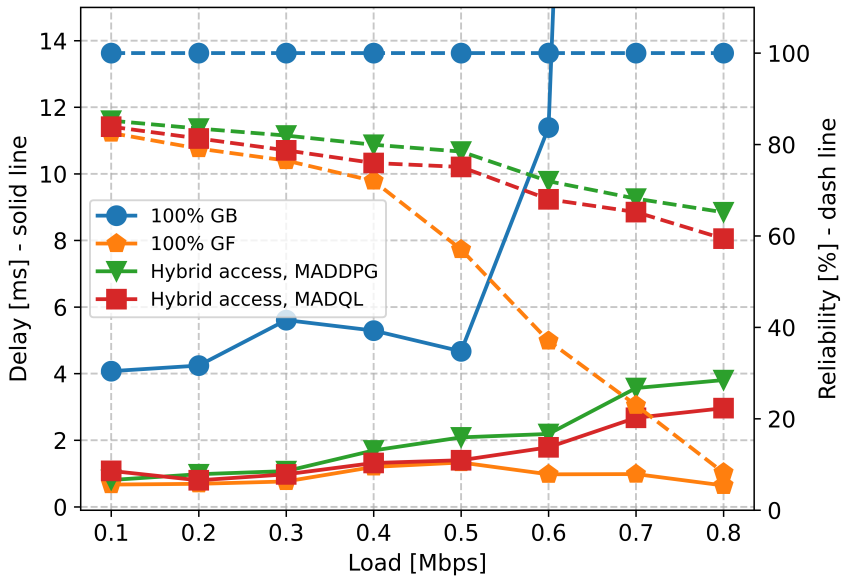

FIGURE 4.10: *Comparison of RAN Delay and transmission Reliability between access protocols (only collision).*

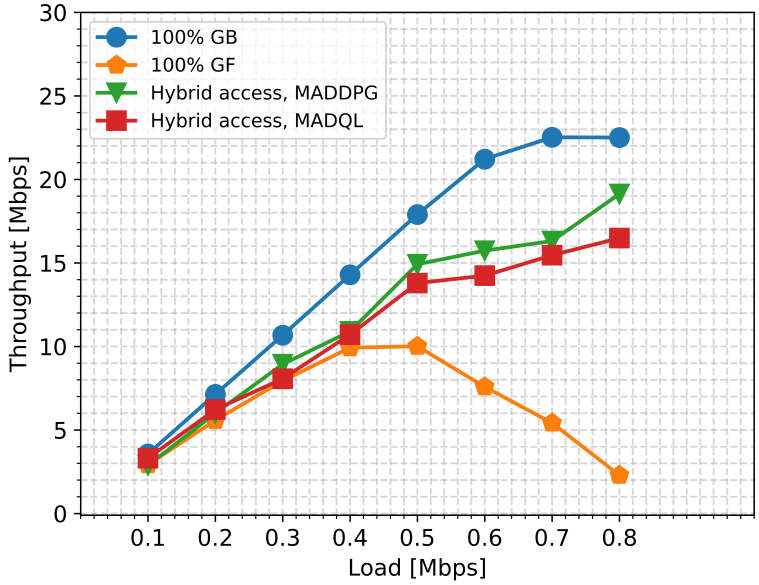

FIGURE 4.11: *Average throughput performance between access protocols (only collision).*

Figure 4.10 illustrates the performance of RAN delay and transmission reliability between *(i)* 100 % GB ($\rho_{\pi_0} = 1$), *(ii)* 100 % GF ($\rho_{\pi_0} = 0$), *(iii)* MADQL-based hybrid access scheme ($\rho_{\pi_0} = 0.5$) and *(iv)* MADDPG-based hybrid access scheme ($\rho_{\pi_0} = 0.5$). Generally speaking, guaranteed radio resources are well managed by centralized gNb scheduling at GB method when packet transmission achieves 100 % reliability in good channel condition. In exchange, access latency is always large due to the 5-step

resource allocation procedure. The upsurge of RAN latency is also observed when each user generates higher traffic load (0.6 Mbps). In this case, network capacity is almost reached and each user has to wait longer to access the requested radio resources.

The downward trend can be observed if 100 % GF method is applied. In this case, access latency is maintained at around 1 ms when the RBG radio resources are occupied immediately regardless the traffic load. However, the significant decreasing of transmission reliability due to the transmission collision between users are seen. Starting from $0.5Mbps$, the collision becomes more severe and reduces the successful transmission rate of each user.

Our MADQL algorithm provides a better trade-off when low latency is maintained and transmission reliability is much improved. Our MADDPG algorithm improves the reliability when compared to MADQL because critic part provides better global observation for each user to re-evaluate its policy. Thus, less collision between users are observed. On the other hand, the access latency in this case is slightly higher than MADQL because each agent is likely to to select 5-step GB resource when traffic load is high to minimize the risk of colliding with other transmissions.

Finally, Figure 4.11 depicts the average network throughput between access protocols. In 100 % GB, radio resource allocation is guaranteed and linearly increasing as the function of traffic load until reaching saturation (network capacity). On the other hands, average network throughput increases at low traffic rate and considerably drop at high traffic rate at 100 % GF. By exploiting our MADDPG and MADQL algorithm, network throughput is greatly improved at high traffic when transmission efficiency, inversely proportional to collision rate, is improved.

Performance under collisions and channel dynamics

In this section, we discover the flow control and resource selection when packet retransmission in bad radio channel is considered. Now, the involvement of deep fading effects and resource collision will be taken into account and negatively reduce packet transmission reliability. In case of packet error, NACK is sent back to the agent. If the packet transmission is initially performed in GB and error, then the simple retransmission is carried out until packet is successfully received or the maximum number of retransmission is surpassed. In case of error due to packet collision and/or deep fading in GF channel, shared resource selection procedures will be repeated for the next retransmission.

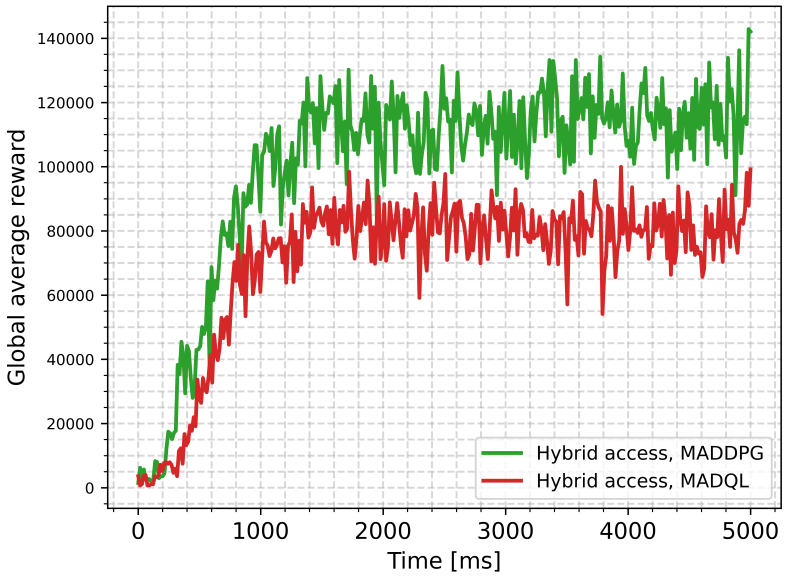

FIGURE 4.12: *Reward convergence for MADDPG and MADQL at load =0.1 Mbps (collision + fading).*

Figure 4.12 compares the obtained rewards between MADQL-based and MADDPG-based algorithm

when each user has 0.1 Mbps traffic load. It shows the convergence of both algorithms and the best rewards of MADDPG. Figure 4.13 and 4.14 displays the performance in terms of **latency**, **reliability** and **network throughput** between *(i)* 100 % GB ($\rho_{\pi_0} = 1$), *(ii)* 100 % GF ($\rho_{\pi_0} = 0$), *(iii)* MADQL-based hybrid access scheme ($\rho_{\pi_0} = 0.5$) and *(iv)* MADDPG-based hybrid access scheme ($\rho_{\pi_0} = 0.5$) according to the traffic load of each user. As expected, the latency due to the handshaking procedure of 100 % GB is the most important to ensure resource access to all users (maximum reliability). On the other hand, the latency of successful communications of 100 % GF is the best, but the shared resources are mismanaged (i.e. collision between users) when user accesses are frequent (worst reliability). Our MADQL-based algorithm better exploits shared resource to maintain low latency and significantly reduce users collisions (higher reliability than 100 % GF). Finally, through the coordination of the centralised gNb in estimating each agent's action, our MADDPG-based algorithm improved the management of the shared resources (i.e fewer collisions are observed) while maintaining low latency. 100 % GB guarantees radio resources for each user according to the round-robin policy π_0. Thus, the network throughput increases linearly when higher traffic loads are generated for each user, until saturation. Nevertheless, this is not the case for 100 % GF when the higher number of opportunistic accesses causes a higher collision probability and accidentally reduces the network throughput at some traffic load generation ($0.15 Mbps$). In a more efficient way,MADQL and MADDPG significantly improve the network throughput as the optimal action-value/policy guides user to select a safer action for opportunistic use of shared resources. With the gNb coordination, MADDPG achieves better performance when traffic load is high.

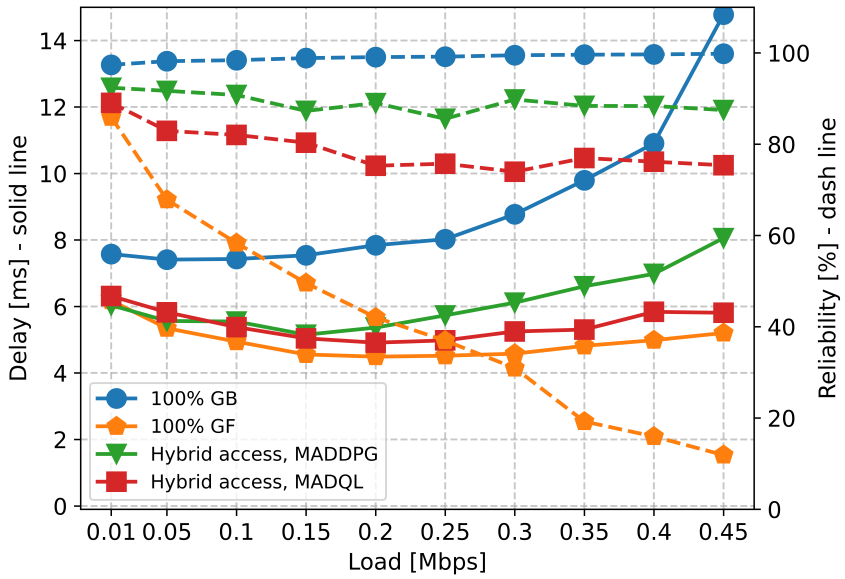

FIGURE 4.13: *Comparison of RAN Delay and transmission Reliability between access protocols (collision + fading).*

4.4.4.2 HIGHER NUMBER OF UES WITH HOMOGENEOUS TRAFFIC

In the second scenario, we evaluate our algorithms MADQL and MADDPG under more multiple associated agents (users) attached in the single gNb. Then, two sub-scenarios where low homogeneous traffic (Average source rate = 0.005 Mbps) and medium homogeneous traffic (Average source rate = 0.1 Mbps) are considered. In both sub-scenarios, N=30,45,60,75 and 90 associated users are placed in the system where both collisions and channel dynamics are applied. In terms of simulation (hyper)parameters, we use similar values as the illustrations in Table 4.1 and Table 4.2.

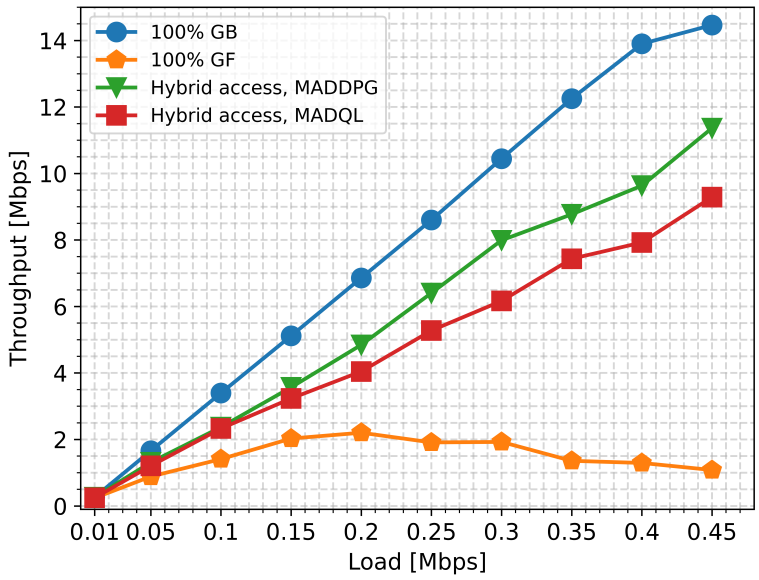

FIGURE 4.14: *Average network throughput between access protocols (collision + fading).*

Low traffic

Figure 4.15 and 4.16 respectively demonstrate performance in terms of delay, reliability and network throughput as a function of the number of users. In this case, each user generates traffic = 0.005 Mbps (low) in their application level. Generally speaking, since the probability of a packet existing at the layer is low, the likelihood of the user being disabled in a time interval is high and this leads to fewer effective users participating in the management of hybrid resources. As the results, the development of latency in 100% GB, 100% GF, MADQL and MADDPG algorithm-based access protocols, when more users are involved, is insignificant. Latency under 100% GB is largest due to 4-step based access while 100% GF access facilitates immediate access for users in the shared channel. Our MADQL and MADDPG based access exploit the hybrid GB/GF channel, then network latency will be between 100% GB and 100% GF. On the other hand, the participation of more users in the shared channel leads to a slight decrease in reliability due to a higher probability of collision, except for 100% GB when radio resources are reliably guaranteed under centralised scheduling policy (Round-Robin).

With respect to average throughput performance, the total throughput of users in this scenario is far from reaching the total capacity of the system, the linear average throughput of the network is related to a higher number of users in the network. Also, the opposite trend is observed in relation to latency performance when users performing 100% GB achieve better bit rate through an more reliable, controlled channel where resources are flexibly allocated for users accordingly. For other access schemes where a fixed number of shared resources in channels is considered, a larger number of users involved will result in a non-negligible collision and cause throughput degradation. By exploiting our algorithms MADQL and MADDPG, throughput improvement is observed when each user learns to coordinate with the others to optimise the global objective.

Medium Traffic

Figure 4.17 and 4.18 respectively demonstrate performance in terms of delay, reliability and network throughput when each user generates traffic = 0.1 Mbps (medium) in their application level. In this scenario, total sum rate of users is comparable to the network capacity and it explains why allocation resources are overloaded and latency significantly increases as a function of number of users in 100 % GB scheme. On the other hand, the reliability is remarkably degraded when most of the time shared

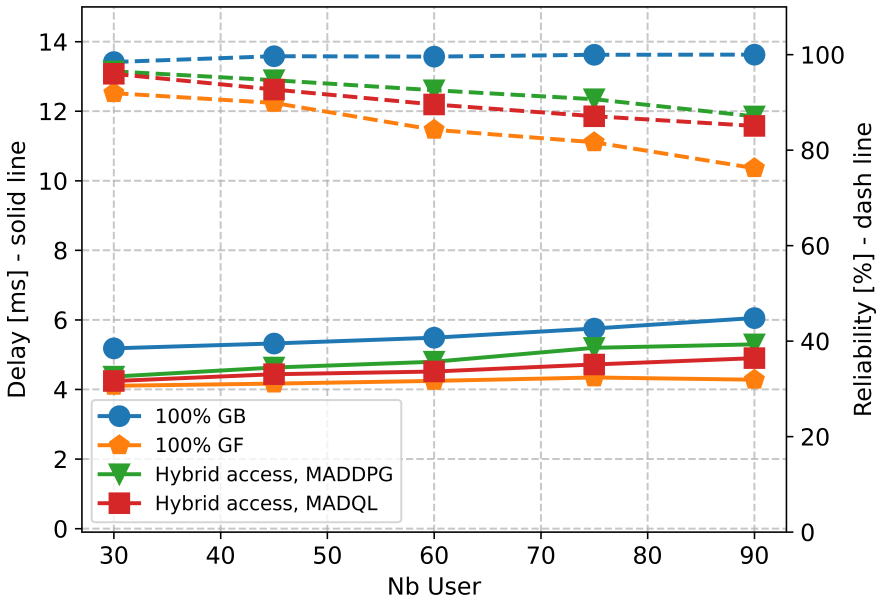

FIGURE 4.15: *RAN delay and reliability as a function of the number of users - low traffic (collision + fading).*

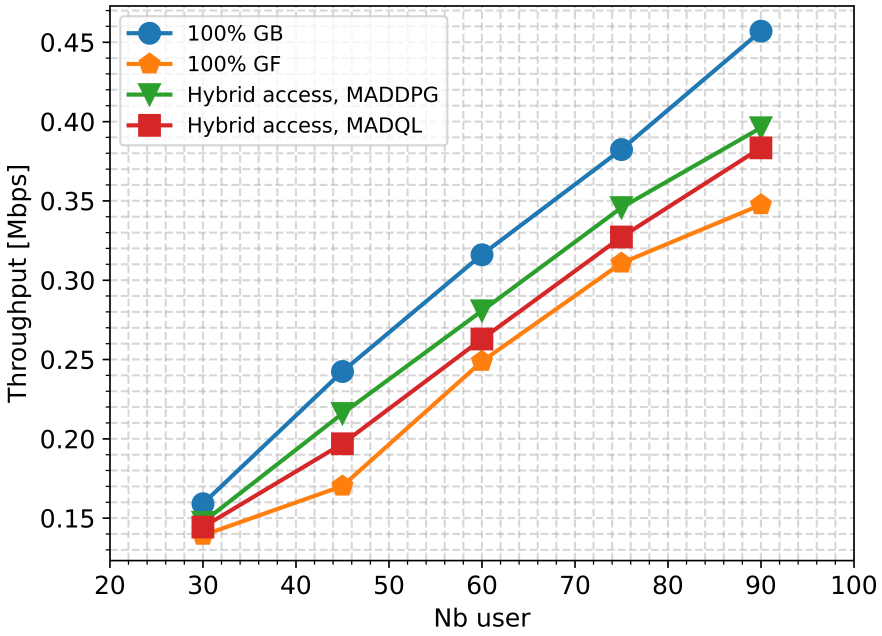

FIGURE 4.16: *Average network throughput - low traffic (collision + fading).*

resources are selected by more than two users and collision happens with those users. By using our algorithm MADQL, associated users better learn how to coordinate with other agents and thus maintain low latency with improved reliability. In addition, MADDPG based algorithm leverage the coordination of centralised gNb to effectively use scheduled resources and reduce collision without severely impacting latency. The impact of each access scheme on network throughput is also demonstrated and it show similar behaviours as in Figure 4.14.

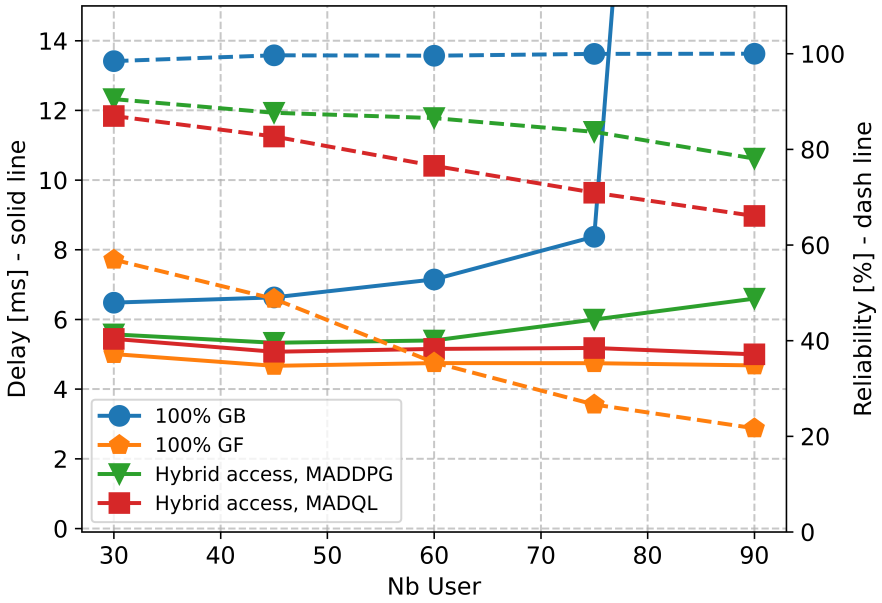

FIGURE 4.17: *RAN delay and reliability as a function of the number of users - medium traffic (collision + fading).*

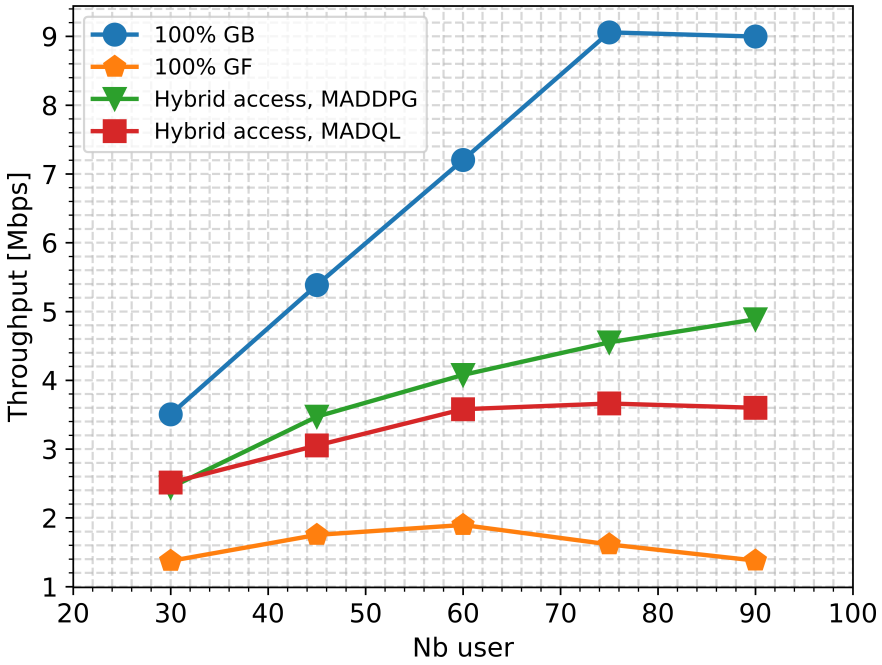

FIGURE 4.18: *Average network throughput - medium traffic (collision + fading).*

4.4.5 Conclusion

In this chapter, we propose two different multi-agent based algorithms for hybrid GB/GF access schemes that are capable of empowering shared resources in the GF channel to improve communication latency and network throughput with low impact on reliability. By means of a system-level simulation, where a full protocol stack is considered, it has been demonstrated that the use of semi-distributed approach (MADDPG), with the support of centralized gNb (critic) having the full evaluation of each associated agent (actor), provides better opportunistic access with fast uplink delay and less collision between users. However, the application of the MADQL approach where only local observation is possible should not be discarded when the algorithm favors agents that exploit shared spectrum access to minimise their uplink latency at the cost of collisions and thus reduce transmission reliability and throughput. The performance gains of our proposal are confirmed when compared with the typical centralized, round-robin scheduling policy (100% GB) and the decentralized, slotted ALOHA protocol (100 % GF).

Based on the promising findings presented in this work, in our future work, we will evaluate more complex scenarios with the participation of *(i)* larger number of users with heterogeneous traffic and *(ii)* distributed access using collaborative MARL will be studied. In essence, the association of a larger number of users with heterogeneous traffic introduces enormous complexities into the system because the priority of resource allocation for each user is not identical and learning performance is likely to diverge in an overly complex scenario. Then, fine tuning of the algorithm is essential to achieve optimal performance in a heterogeneous system. In this case, communication between agents for collaborative policy exchange could be a solution to reduce the complexity of the learning space and result in an optimized solution.

The technical contribution of this chapter has been submitted to the following conference:

[C4] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Hybrid Radio Resource Management based on Multi-Agent Reinforcement Learning,"* accepted in 2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Gothenburg, Jun. 2023.

CHAPTER 5

Conclusions and Future perspectives

5.1 Summary of thesis research

Ultra reliable and low latency communications (URLLC) in 5G networks enable novel applications in industrial automation, medical healthcare, intelligent transportation, etc., where exchange information between nodes are wirelessly transmitted with extremely low delay and ultra-high reliableness. However, in the context of dynamic wireless medium, protocol designs for ensuring ultra-reliability is entangled with the requirement for low latency and the ideal accomplishment of URLLC is very challenging. Moreover, cellular networks are complex systems, which involve multiple communication protocols and layers, so the activation cross-layer protocols exert impacts on the formation of E2E latency as well as system reliability. In some cases, the poor coordination of available network protocols causes not only inefficient usage of resources (in terms of radio, power, computing and caching storage) but also latency violation when the activation of additional protocols is not instantaneous. It introduces the necessities of an orchestrator which effectively manages cross-layer protocols to ensure URLLC communications and deal with possible changes of network topology.

In this thesis, we focused on radio resource management for URLLC communication. More specifically, we first proposed an architecture of network orchestration which takes classified cross-layer protocols, target KPIs and relevant changes in the radio networks (e.g. channel fading, medium interference, time varying traffic, etc) into account in Chapter 2. With the integration of data-driven feature blocks, it allows the self-reorganisation of network functions towards the optimal one by appropriately applying network protocols (mechanisms) in dealing with rapid changes of radio environment. The AI-enabled core helps with predicting the predictable events for the better orchestration and bound the performance in case of unpredictable events are involved. Afterwards, we provide two orchestration methods *(i)* early decision making based on mechanisms scheduling and *(ii)* jitter-aware scheduling to turn latency/jitter dependent environment into dependent application control.

Our first research direction towards the embodiment of early decision making is presented in Chapter 3 where radio resources are effectively managed at RAN level in guaranteeing low latency, high reliability and high spectrum efficiency. At first, by means of system level simulation based on NS3, we prove how proactive resource scheduling can be implied in the regular HARQ procedure at MAC layer to reduce global RAN latency. In this work, we clarified the correlation between traffic source rate and PHY/MAC layer configurations to serve the data over the air in a highly dynamic channel and highlighted the compromise between RAN latency, transmission reliability and radio resource efficiency by analysing the application of different proactivity resource scheduling levels to re-transmit corrupted packets. Although the RAN latency is reduced as the function of higher proactive resource scheduling, the lacking of adaptation is the main source causing poor radio usage efficiency. In order to deal with that, we propose an

adaptation control algorithm to improve proactive resource scheduling. Based on Lyapunov stochastic optimization tool, a mathematical framework is proposed to understand the performance-delay trade-off by minimising the objective function of the total resource allocation and the total queue length that is parameterized by a ν value. Our results reveal that an appropriate selection of ν enables the dynamic selection of proactive retransmission to overcome the defect of a long RTT in the reactive scheme while maintaining a good level of resource efficiency that is considered a drawback of the fixed proactive schemes. Following this research direction, we propose an adaptive, reliability-aware resource scheduling algorithm dealing with dynamic scenarios (e.g., random bursty traffic, time-varying channel). It takes into account the traffic arrival at the network layer, the queue behaviours at the data link layer and the risk that the applied decision might trigger packet loss. The trade-off between the resource efficiency, latency and reliability is achieved by the timing and intensity of decisions and can be parameterized with ν and α. Afterwards, our queue-aware and channel-aware solution is evaluated in a system-level simulator and validated by an experimental testbed using OAI.

In chapter 4, we exploit AI-based orchestration architecture in hybrid GB/GF radio resource management. In this research, we explore the utilization of opportunistic, shared radio resources in GF channel, where each user performs random access for shortening resource access latency, in addition to scheduled resources in GB channel which is centralized managed by gNb. Now, each user that wants to access radio resources has 2 different options at RLC layer. Firstly, it will defer the resource scheduling function to gNb by sending Scheduling Request (SR) via GB channel. After 5-step message exchange between UE and gNb, resource allocation for requested user is guaranteed with perfect efficiency at the cost of high delay. However, this scheme might not be suitable for URLLC applications. On the other hand, each user directly decides which resource group in shared resource pool it will occupy to transmit the packet holding in RLC buffer and significantly reduce the access latency. However, it will face to the non-negligible transmission collision when other user selects the same resource for its transmission. Then, we proposed 2 different algorithms based on MARL framework to minimize the access latency and transmission error due to collision while maintaining good average network throughput: *(i)* MADQL and *(ii)* MADDPG to solve the hybrid access optimization problem. The former algorithm is able to solve the optimization problem with only local observation of each user (agent). In the latter, gNb plays an important role in criticizing the action of each associated user (agent) in the network. This semi-distributed architecture helps gNb having the full evaluation of each associated agent (actor) and provides better opportunistic access with fast uplink delay and less collision between users than the former algorithm. However, the application of the MADQL approach where only local observation is possible should not be discarded when the algorithm favors agents that exploit shared spectrum access to minimise their uplink latency at the cost of collisions and thus reduce transmission reliability and throughput. The performance gains of our proposal are confirmed when compared with the typical centralized, round-robin scheduling policy (100% GB) and the decentralized, slotted ALOHA protocol (100 % GF).

5.2 Future Perspectives

In this thesis, we proposed 2 main innovative solutions for better radio resource management. Although we introduced advanced features to support URLLC communication, there are still rooms for the future developments.

The first solution, which is presented in Chapter 3, addresses DL communication, where single gNb fully controls radio resource allocation for single UE in optimizing the single performance trade-off between latency, reliability and resource allocation effectiveness in proactive resource allocation strategy. The mathematical optimization, which is based on Lyapunov optimization framework, is then transformed into protocol designs for system level simulation and experimental test-bed based on OAI. The immediate extension of our work concerns the management of multi-user scenario where the more advanced scheduling functions are considered to guarantee the accessibility of proactive radio resources for

each user. Furthermore, towards some extents, the automation, scalability and adaptability are missing features of our solution in dealing with unexpected changes of network topology (e.g: the dependent of obtained performance on manually selected parameters ν and α or the consideration of single-user only, etc). Then, the future research direction will answer the following questions:

- (1) *Can we apply the framework in multi-user scenario where a mixed setting between UL and DL are applied.*
- (2) *Can we adapt the control parameters on-the-fly with respect to the changes of objective functions in time?*
- (3) *The compatibility of proposed solutions with the full protocol stacks communication.*

These open questions unroll new perspectives to build a more comprehensive algorithm to solve the optimization problems.

The second solution introduced in Chapter 4 concerns UL direction and multiple users are attached in a single gNb. In this work, multi-user with homogeneous Poisson traffic attempt to explore the opportunistic, shared resource pool to reduce latency access with possible collision, in additions to scheduled resources without resource collusion but high delay access. Our solution based on MARL framework: fully-distributed decision maker model (MADQL) and semi-distributed decision maker model (MADDPG), are proposed to reduce transmission collision, reduce access latency and maintain a good average network throughput. The future extension of this research can be followed by different axes:

- **(1) Heterogeneous support:** *When network needs to be self-orchestrated to satisfy heterogeneous requirement, then how our optimization framework need to be tuned to bring optimal solutions? Also, our framework should be validated with various centralized scheduling policy (e.g. proportional fairness, maximum rate besides classical round robin)* and decentralized multiple access protocol (e.g. CSMA protocol). Then, the evaluation of our proposed algorithm with different set of centralized scheduling policy in GB channel and decentralized multiple access protocols in GF channel are considered.
- **(2) Transfer learning:** *Based on centralized training and decentralized execution framework, we are able to decentralize the intelligence to different access nodes in the network and deal with the scalability dimension when more nodes are added to the communication network. It paves the way for the transfer learning concept to be integrated for efficient learning. Particularly, when a device joins the network, it can communicate with neighbor agents to search for optimal its learning model or policy without requiring the retraining process. In this case, various associated problems are raised such as learning convergence and communication resources for model exchange between agents.*
- **(3) Efficient training approach:** *In case of massive number of users are involved in the network, the global action state and state space are extremely large and lead to the inefficient training. It imposes a big challenge for the exploitation of URLLC application in practice. With the deployment of our proposed centralised training and decentralised execution, the question of high communication efficiency, data security and privacy are posed. It laid the foundation for the consideration of federated learning framework.*

APPENDICES

ANNEXE A

Résumé étendu de thèse

Les travaux de thèse présentés dans ce manuscrit se concentrent sur les mécanismes avancés pour la gestion et l'orchestration des ressources dans les communications ultra-fiables à faible latence (URLLC). Plus spécifiquement, nous étudions comment à gérer de manière effciente et efficace les ressources radio des réseaux mobiles en se basant sur l'optimization de Lyapunov et sur des approches d'apprentissage (semi)distribuées. Dans la partie suivante, nous présentons un bref résumé des principales contributions de notre travail de recherche. Nous commençons par décrire les contextes de l'étude, puis nous identifions les principaux défis qui y sont associés, avant de proposer la méthodologie pour adresser ces problèmes.

Introduction et contexte d'étude

Les futurs réseaux mobiles sont censés prendre en charge un nombre exponentiel d'appareils connectés et un large champ d'applications. Bien que les technologies de 4G ont été largement déployées commercialement dans le monde entier, nous permettant d'atteindre une meilleure capacité de données et des vitesses de transmission plus élevées, la prolifération de nouveaux cas d'utilisation qui exigent des améliorations en matière de latence, de fiabilité et de scalabilité posent des défis sans précédent pour les systèmes 4G. Les technologies de 5G sont donc introduites pour améliorer les performances des réseaux en termes de mobilité, d'énergie, de vitesse, de diversité des services, de fiabilité améliorée, de latence réduite et de débit plus élevé **[6]**. Les services qui seront pris en charge par la 5G peuvent généralement être divisés en trois catégories principales :

- Enhanced Mobile Broadband (eMBB) : fournit des services de données à haut débit (jusqu'à 20 Gbps en voie descendante **[8]**) pour les applications. Il constitue la base de connexions stables à très haut débit pour le contenu, les services et les données multimédias (par exemple : la vidéo 3D, les écrans Ultra Haute Définition (UHD), la réalité augmentée, etc.)
- Massive Machine-Type Communication (mMTC) : Ces services sont applicables à des scénarios dans lesquels un grand nombre de dispositifs connectés transmettent des données sporadiques (petites charges utiles imprévisibles) et des quantités relativement faibles de données sensibles non retardées (par exemple, réseau intelligent, maison/bâtiment intelligent, villes intelligentes, etc.)
- Ultra Reliable and Low Latency Communications (URLLC) : Il prend en charge les applications dans lesquelles la transmission de paquets courts est soumise à des exigences de faible latence (environ 1 ms) et de fiabilité extrêmement élevée (avec des taux d'erreur de l'ordre de 10^{-5} - 10^{-9}). Ces cas d'utilisation vont de l'automatisation industrielle, les applications critiques, les voitures autonomes, etc.

Parmi ces services clés, la conception du protocole du service URLLC est sans doute considérée comme la plus difficile et la plus problématique parce qu'elle doit répondre à deux exigences compromettantes : une faible latence et une très grande fiabilité. Le respect des exigences strictes de l'URLLC est encore compliqué par des perturbations environnementales imprévisibles telles que le brouillage du canal, la mobilité de l'utilisateur ou le comportement du trafic, etc. Dans le contexte d'un environnement dynamique et d'une exploitation multi-utilisateurs de ressources limitées, la gestion efficace des ressources radio pour répondre aux exigences de l'URLLC est complexe. Donc, j'aimerais proposer des solutions avancées des solutions avancées, flexibles, proactives, évolutives pour résoudre ces problèmes.

L'orchestration centralisée

Dans les communications sans fil, la gestion des ressources radio (RRM) est d'une importance capitale pour permettre des communications ultra-fiables et à faible latence dans des environnements dynamiques (par exemple, le comportement du trafic, la mobilité des utilisateurs, l'évanouissement des canaux, etc.). Essentiellement, la gestion des ressources radio comprend toutes les stratégies, procédures et algorithmes permettant de gérer efficacement les ressources radio (par exemple, la formation de faisceaux, l'attribution des ressources radio, le choix de la modulation et du système de codage des canaux, etc.). A priori, la gestion centralisée des ressources au niveau de la station de base donne les meilleurs résultats car les informations importantes sur les utilisateurs sont souvent collectées et traitées de manière uniforme. Parmi les outils prometteurs pour la communication URLLC au niveau RAN, nous pensons que le protocole HARQ) est le point de blocage qui doit être optimisé pour répondre aux exigences strictes en matière de latence tout en respectant les contraintes de fiabilité et d'efficacité des ressources.

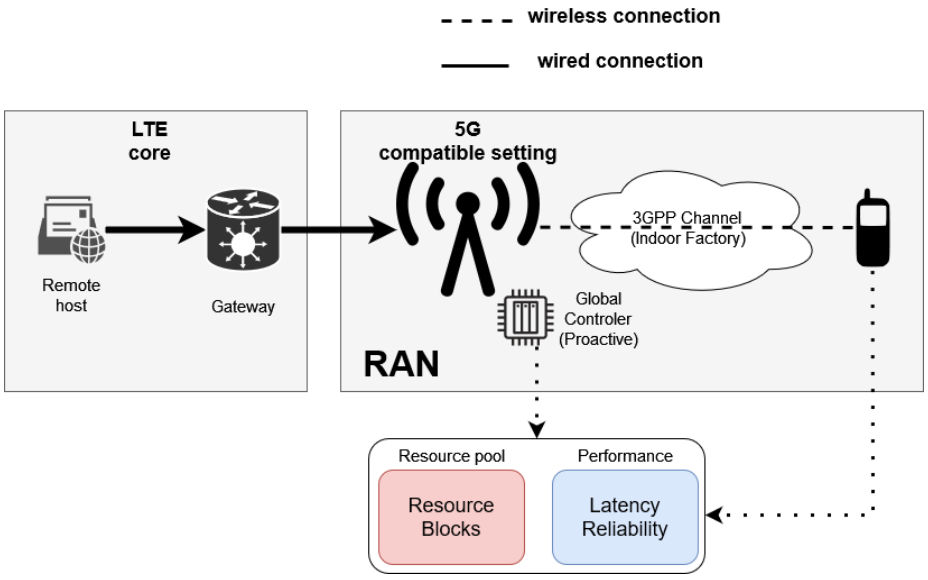

Figure A.1 : *Gestion centralisée des ressources dans les communications sans fil.*

La solution complète que nous proposons dans ce travail est divisée en 4 parties principales.

Tout d'abord, à l'aide d'une simulation au niveau du système basée sur NS3, nous démontrons comment la planification proactive des ressources peut être intégrée dans la procédure HARQ ordinaire au niveau de la couche MAC afin de réduire la latence globale du réseau RAN. Dans ce travail, nous avons clarifié la corrélation entre le débit de la source de trafic et les configurations des couches PHY/MAC pour servir les données dans un canal hautement dynamique. Nous avons ensuite mis en évidence le compromis entre la latence RAN, la fiabilité de la transmission et l'efficacité des ressources radio en analysant l'application de différents niveaux de planification proactive des ressources pour retransmettre les pa-

quets interrompus. Bien que la latence du RAN soit réduite en fonction d'une planification proactive des ressources plus élevée, le manque d'adaptation est la principale cause de la faible efficacité de l'utilisation des ressources radio.

Dans l'étape suivante, un algorithme de contrôle adaptatif est proposé pour améliorer la gestion proactive des ressources. Basé sur l'outil d'optimisation stochastique de Lyapunov (cadre de Lyapunov 1), un cadre mathématique est proposé pour comprendre le compromis performance-délai en minimisant la fonction objective de l'allocation totale des ressources et de la longueur totale de la file d'attente, paramétrée par une valeur de v. Les résultats montrent qu'un choix approprié de v permet la sélection dynamique des retransmissions proactives pour surmonter le défaut d'un long RTT dans l'ordonnancement réactif. Nos résultats montrent qu'un choix approprié de v permet à la sélection dynamique des retransmissions proactives de surmonter le défaut d'un long RTT dans le schéma réactif, tout en maintenant un bon niveau d'efficacité des ressources, ce qui est considéré comme un problème des schémas proactifs fixes.

Dans la troisième partie, nous proposons une solution adaptative et sensible à la fiabilité pour traiter le système dans lequel la perturbation critique se produit fréquemment et où davantage de ressources radio sont nécessaires pour améliorer la fiabilité de la communication. Sur la base de l'optimisation de Lyapunov, nous prenons en compte la dynamique du processus d'arrivée du trafic au niveau du réseau, le comportement de la transmission sans fil au niveau de la couche physique et le risque que la décision appliquée puisse entraîner la perte de paquets.

Finalement, nous validons le déploiement des algorithmes proposés mentionnés précédemment avec le matériel conforme à la 5G. En tenant compte des contraintes matérielles en temps réel et des écarts par rapport à la simulation préliminaire, nous validons les gains de performance en termes de latence, d'efficacité des ressources et de fiabilité.

Les contributions techniques de ce chapitre ont été validées par les publications suivantes :

[C1] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Proactive Resource Scheduling for 5G and Beyond Ultra-Reliable Low Latency Communications,"* IEEE 95th Vehicular Technology Conference : (VTC2022-Spring, doi : 10.1109/VTC2022-Spring54318.2022.9860872.

[C2] **L. N. Dinh**, I. Labriji, M. Maman, and E. Calvanese Strinati, *"Toward URLLC with Proactive HARQ Adaptation,"* in 2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), pp. 220–225. doi : 10.1109/EuCNC/6GSummit54941.2022.9815615.

[C3] **L. N. Dinh**, R. Bertolini, M. Maman, *"Dynamic Resource Scheduling Optimization for Ultra-Reliable Low Latency Communications : From Simulation to Experimentation,"* in 2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), sept. 2022, p. 1026-1031. doi : 10.1109/PIMRC54779.2022.9977893.

La gestion des ressources hybrides

La sixième génération de communications mobiles (6G) élargit le champ d'application des réseaux de communication sans fil à des réseaux efficaces, élastiques et dynamiques. Cependant, les approches centralisées actuelles peuvent ne pas être un bon candidat en raison de la surcharge de signalisation et de la nécessité d'un calcul excessif, ce qui n'est pas pratique en raison du déploiement dense des utilisateurs et des stations de base. En outre, comme indiqué ci-dessus, la gestion des ressources radio (RRM) implique de nombreuses variables d'optimisation qui ne sont pas toujours bien caractérisées mathématiquement

(en raison de la nature dynamique de l'environnement de propagation, de la mobilité de l'utilisateur). Cela pose des difficultés dans la formulation et la résolution des problèmes d'optimisation. Cette situation a stimulé la recherche de solutions plus avancées en matière de gestion des ressources radio. Lorsqu'il est question de stratégies d'attribution des ressources pour répondre aux exigences des communications ultra-fiables et à faible latence (URLLC), les méthodes actuelles impliquent la réservation de ressources radio dédiées alors qu'une utilisation opportuniste du spectre partagé est disponible. En particulier, plusieurs solutions sont proposées pour intégrer les approches opportunistes : transmission sans octroi, préemption des ressources radio pour une utilisation immédiate (par exemple, préemption de mini-lots), allocation semi-distribuée (par exemple, pool de ressources partagées pour les communications d'appareil à appareil), ou transmissions se chevauchant (par exemple, accès multiple non orthogonal). Dans ce travail, nous proposons un schéma hybride d'allocation des ressources basé sur l'accès par subvention (GB) et sans subvention (GF) pour des communications à faible latence, à haute fiabilité et à haut débit. Ensuite, nous tirons parti du cadre d'apprentissage par renforcement multi-agents dans lequel chaque utilisateur associé et chaque station de base gèrent conjointement les ressources limitées d'une manière entièrement distribuée ou semi-distribuée.

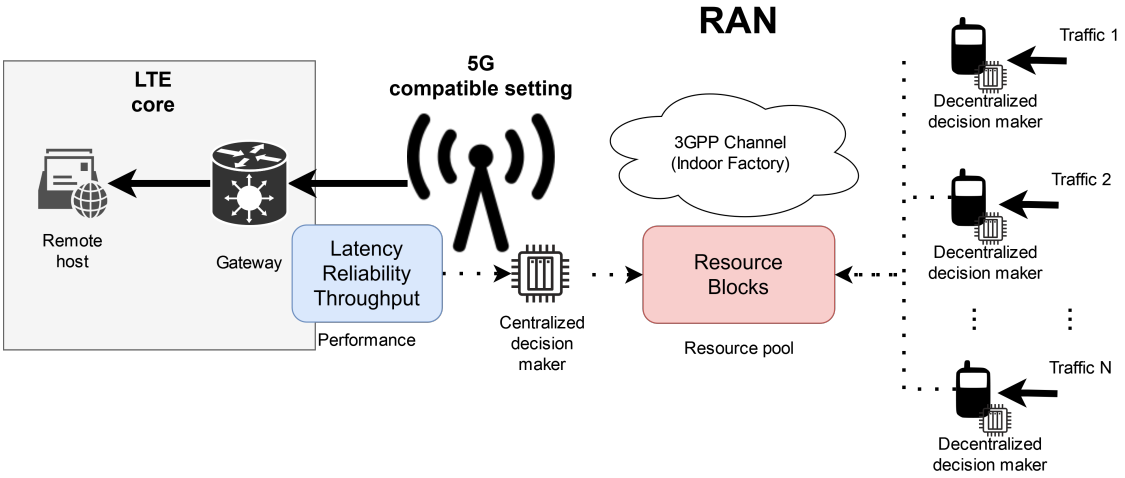

FIGURE A.2 : *Gestion hybride des ressources pour les communications sans fil.*

Notre premier algorithme est basé sur Multi-agent Deep-Q Learning (MADQL), dans lequel chaque utilisateur effectue la sélection des ressources en se basant uniquement sur son observation locale du trafic et des files d'attente de transmission (GB et GF). Étant donné que la sélection des ressources selon la méthode GB fournit un accès garanti aux utilisateurs avec retard, tandis que l'accès opportuniste GF minimise le retard total au prix d'une collision potentielle avec d'autres utilisateurs sélectionnant les mêmes ressources, réduisant ainsi le débit global et l'efficacité de la communication. Ensuite, chaque utilisateur (agent) doit prendre sa propre décision pour optimiser l'objectif global conçu et Multi-agent Deterministic Policy Gradient (MADDPG) est proposé pour guider l'utilisateur dans la sélection des actions de grande valeur. D'autre part, l'algorithme MADDPG est proposé pour exploiter le rôle central du gNB (critique) afin d'évaluer les actions de sélection des ressources de tous les utilisateurs (acteurs). Ensuite, la qualité des bonnes/mauvaises actions prises est transmise à chaque utilisateur (agent) pour qu'il mette à jour sa politique vers la politique optimale. Ensuite, en tirant parti de la formation centralisée et de l'exécution décentralisée, la gestion optimale des ressources hybrides est proposée de manière semi-distribuée.

Nos résultats montrent que l'utilisation de l'approche semi-distribuée (MADDPG), avec le soutien d'un gNB centralisé (critique) ayant l'évaluation complète de chaque agent associé (acteur), fournit un meilleur accès opportuniste avec un faible délai de liaison montante et moins de collisions entre les utilisateurs. Cependant, l'application de l'approche MADQL, où seule l'observation locale est possible, ne doit pas être écartée lorsque l'algorithme favorise les agents qui exploitent l'accès partagé au spectre pour

minimiser leur temps de latence sur la liaison montante au prix de collisions, réduisant ainsi la fiabilité de la transmission et le débit. Les gains de performance de notre proposition sont confirmés lorsqu'on les compare à la politique d'ordonnancement centralisée typique de type round-robin (100% GB) et au protocole décentralisé ALOHA à créneaux (100% GF).

La contribution technique de ce chapitre a été soumise à la conférence suivante :

[C4] **L. N. Dinh**, M. Maman and E. Calvanese Strinati, *"Hybrid Radio Resource Management based on Multi-Agent Reinforcement Learning,"* accepted in 2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Gothenburg, Jun. 2023.

Conclusion

Dans cette thèse, nous avons développé et étudié la gestion des ressources radio pour la communication URLLC. Plus précisément, au chapitre 2, nous avons d'abord proposé une architecture d'orchestration de réseau qui tient compte des protocoles intercouches classifiés, des indicateurs clés de performance et de la dynamique du réseau radio (par exemple, l'évanouissement du canal, l'interférence moyenne, le trafic variable dans le temps, etc.) Grâce à l'intégration de blocs fonctionnels pilotés par les données, il permet l'auto-organisation des fonctions de réseau vers l'optimum en appliquant les protocoles de réseau (mécanismes) de manière appropriée lorsqu'il s'agit de faire face à des changements rapides dans l'environnement radio. Le noyau basé sur l'AI aide à prévoir les événements prévisibles pour une meilleure orchestration et limite les performances en cas d'événements imprévisibles. Nous proposons ensuite deux méthodes d'orchestration (i) la prise de décision précoce basée sur l'ordonnancement des mécanismes et (ii) l'ordonnancement tenant compte de la gigue pour transformer l'environnement dépendant de la latence/gigue en un contrôle dépendant de l'application.

À la lumière de l'architecture du décideur proactif, nous montrons au chapitre 3 comment la latence est réduite en parallélisant l'allocation dans un environnement dynamique. Nos résultats suggèrent qu'un algorithme d'adaptation est nécessaire pour réduire le gaspillage des ressources (c'est-à-dire améliorer l'efficacité des ressources) en raison de la réaction excessive du processus de parallélisation. Nous proposons ensuite un algorithme d'allocation adaptatif pour améliorer l'efficacité de l'allocation des ressources tout en maintenant l'objectif de fiabilité à long terme de la communication. Sur la base des optimisations de Lyapunov pour la gestion des systèmes à deux files d'attente au niveau des couches Radio Link Control (RLC) et Medium Access Control (MAC), nous concevons un cadre d'optimisation qui tient compte de la latence, de la fiabilité et de l'efficacité des ressources du Radio Access Network (RAN). Dans certains cas, lorsque des messages critiques doivent être garantis dans un environnement hautement dynamique, nous adaptons ce cadre pour une allocation tenant compte de la fiabilité. Nous mettons ensuite en œuvre les algorithmes proposés dans le banc d'essai 5G OpenAirInterface (OAI) afin de prouver leur faisabilité dans des conditions de temps réel.

Enfin, dans la dernière partie de notre travail, nous nous concentrons sur les multi-utilisateurs partageant des ressources radio limitées et nous proposons un cadre de gestion des ressources hybrides (schéma d'accès Grant-Based (GB)/Grant-Free (GF)) pour optimiser la latence UL, le débit du réseau et la fiabilité des communications URLLC. Basés sur l'apprentissage par renforcement multi-agents (MARL), les algorithmes Multi-agent Deep-Q Learning (MADQL) et Multi-agent Deterministic Policy Gradient (MADDPG) sont introduits pour guider l'utilisateur associé dans le choix de l'action optimale de sélection des ressources et optimiser l'objectif global.

APPENDIX B

End-to-End Protocol Stack

The architecture of end-to-end simulation consists of EPC Network components, LTE components and NR interface as illustrated in Figure B.1. On the right hand side of architecture, we built a remote host (server) which connects to a SGW/PGW, through a router. Inside the SGW/PGW, there is an application (**EpcSgwPgwApp**), which is responsible for encapsulating the packet through GTP, is installed. By setting up **EpcEnbApp** inside the gNb, the packets exchanged between gNb and SGW/PGW are handled via IP connection which represents the backhaul network. At gNb, packets are received and decapsulated before being forwarded to RAN. The packet, after being sent into a designed channel, will be then passed to higher layers of UE in case of correctly received. Concerning the Uplink pathway, the protocol is the same as the one described above but on the contrary direction.

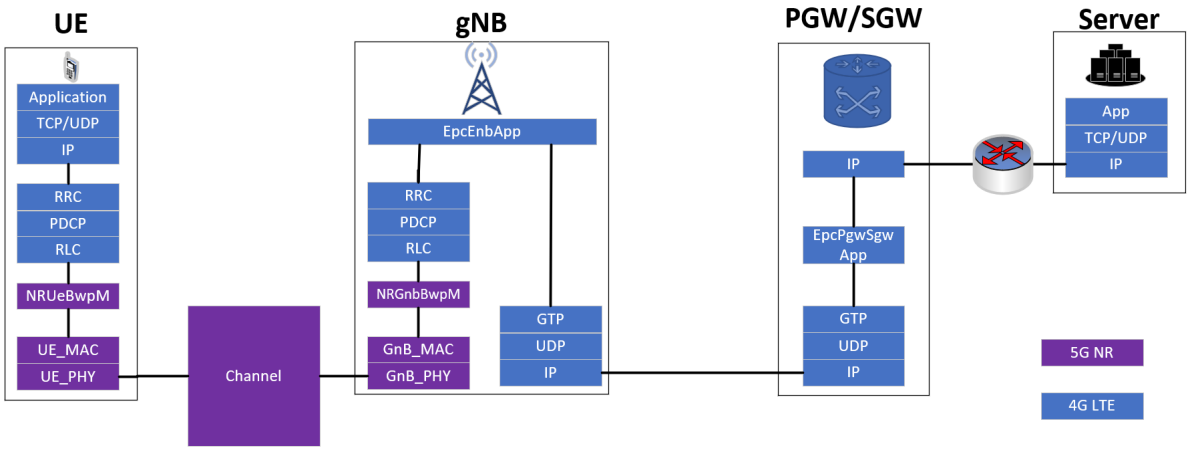

FIGURE B.1: *End-to-end architecture overview.*

In order manage the RAN entities in gNb and UE, **NrGnbNetDevice** and **NrUeNetDevice** are installed at both side, respectively. Hereafter, we are going to detail the RAN protocol stack which is mainly used in our system level simulation. The packet data flow in RAN therefore illustrated as Figure B.2 below.

At first, the IP packets will arrive SDAP layer in order for being mapped into a data radio bearer before being forwarded into the PDCP layer. The PDCP layer will perform header-compression to reduce the number of redundant information bits and be responsible for in-sequence delivery of packets. Afterthat, a PDCP header is added to PDCP SDU, carrying essential information for the deciphering at receiver, to form PDU. The PDCP PDUs will then go through RLC layer which is responsible for segmentation/concatenation, duplicate detection and in sequence delivery of the packets to the higher layers. The RLC provides services to the PDCP under the form of radio bearers in which one RLC entity per radio bearer

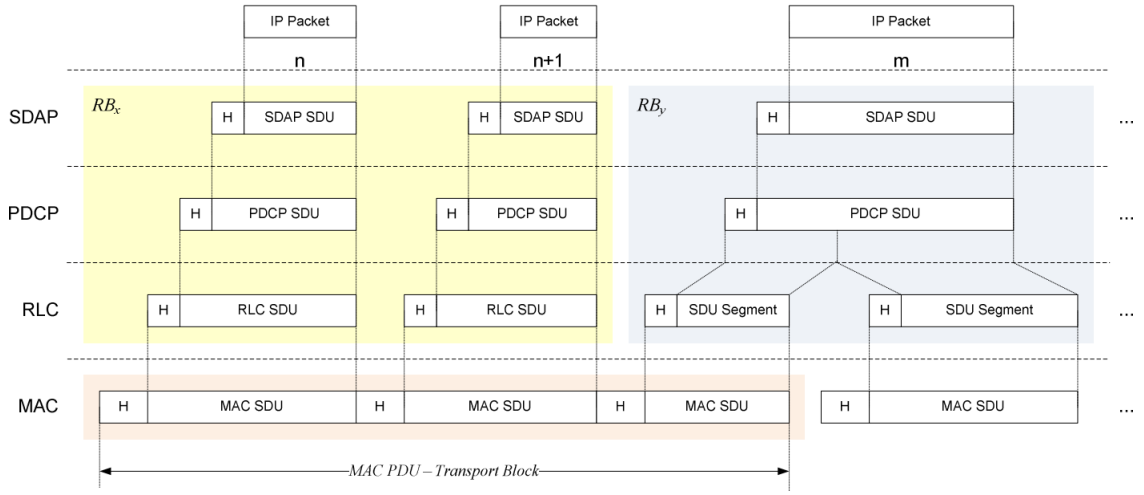

FIGURE B.2: *Packet Data Flow* ***[106]***

is configured for each device. Then, the attachment of RLC header will facilitate the in-sequence delivery of the PDCP PDUs in the case of retransmission. Subsequently, the RLC PDUs are forwarded to the MAC layer, which multiplexes a number of RLC PDUs together and adds MAC header to create a TB. The size of a TB depends on various factors, such as the configuration of frame transmission and the instantaneous rate selected by the scheduler.

B.1 Radio Resource Control (RRC) Protocol

The RRC is a control plane protocol and is in charge of setting crucial exchanges for the communication session. The implementation of RRC is desired by **[107]** and can be illustrated as in Figure B.3. RRC layer is responsible for establishing the radio bearers and configuring all lower layers using RRC signalling between the gNb and UE. The RRC model which is used in our simulator provides the following functionalities:

- Generation (gNb) and interpretation (UE) of System Information (Master Information Block and System Information Block Type 1 and 2).
- Initial cell selection
- RRC connection establishment procedure.
- RRC reconfiguration procedures, supporting: SRS reconfiguration, UE measurement, data radio bearer setup and handover.

B.1.1 Broadcast of System Information

System Information is a critical Downlink information for UE to access network. It consists of:

- **Master Information Block(MIB):** carries information related to the PHY layer, MIB is generated during cell configuration and broadcasted every 10 ms at the beginning of radio frame as a control message.
- **System Information Block Type 1 (SIB1):** brings information regarding network access, it is broadcasted every 20 ms at the middle of radio frame as a control message. Before perceiving SIB1, UE must have decoded MIB.

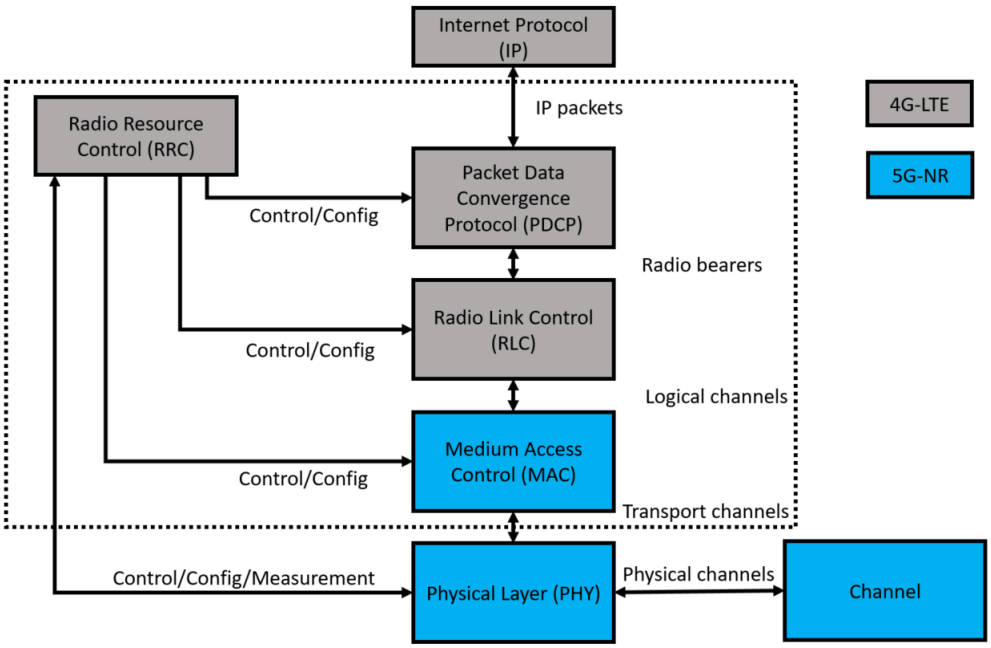

FIGURE B.3: *Illustration of protocol layers in our system level simulation*

- **System Information Block Type 2 (SIB2):** contains UL- and RACH- related settings. it is scheduled to transmit via RRC protocol at 16 ms after cell configuration, and then repeats every 80 ms.

B.1.2 Initial Cell Selection

Initial cell selection is an IDLE mode procedure which is performed by UE when it has not yet attached to a gNb. The objective of the procedure is to look for a suitable gNb and acquire access to the cellular network.

This procedure is typically done at the beginning of the simulation, in which UE finds the strongest cell which gives UE the highest RSRP, i.e $Cell\_ID_{serving-cell} = \arg\max_k \{RSRP(k)\}$. Once suitable cell is found, the initial cell selection procedure will be enacted.

B.1.3 RRC connection establishment

The RRC connection establishment consists of three steps: RRC Connection Request message from UE, the RRC Connection Setup message from gNb and finally, RRC Connection Setup Complete message from UE. Once RRC connection establishment complete, the UE will be provided with periodic alignment command from gNb and will continually perform cell measurements for handover.

B.1.4 Ideal RRC protocol model

In our system level simulator, all RRC messages and information elements are transmitter between the gNb and UE in an ideal fashion, which means that the connection is error free and it does not consume any radio resources to establish the exchange. In our implementation, it is achieved by directly passing the RRC data structures between UE and gNb entities without considering the lower layers (PDCP,RLC,MAC,scheduler).

B.2 Packet Data Convergence Protocol (PDCP)

The implementation for PDCP follows the 3GPP specifications **[107]**. PDCP layer is responsible for the IP header compression of user-plane data packets so as to reduce the number of information transmitted over the air. The functional view of PDCP layer can be shown in Figure B.4. The functionalities of PDCP layer will be expressed as follows:

- From Tx, PDCP SDUs coming from PDCP sub-layer are stored in transmission buffer, then the Sequencing Numbering functionality will add sequence number to each of incoming PDCP SDU.
- the numbered SDUs then go through header compression procedure (only for User Plane Data).
- Afterwards, there are 2 paths that PDCP SDUs can be forwarded: (i) Going through Integrity/Ciphering or (ii) Directly going to Add PDCP Header. Integrity is applied only to signalling messages while both signalling and data messages can be ciphered.
- Then, PDCP header will be added on top of PDCP SDU.
- At Rx, the process will be inverted.

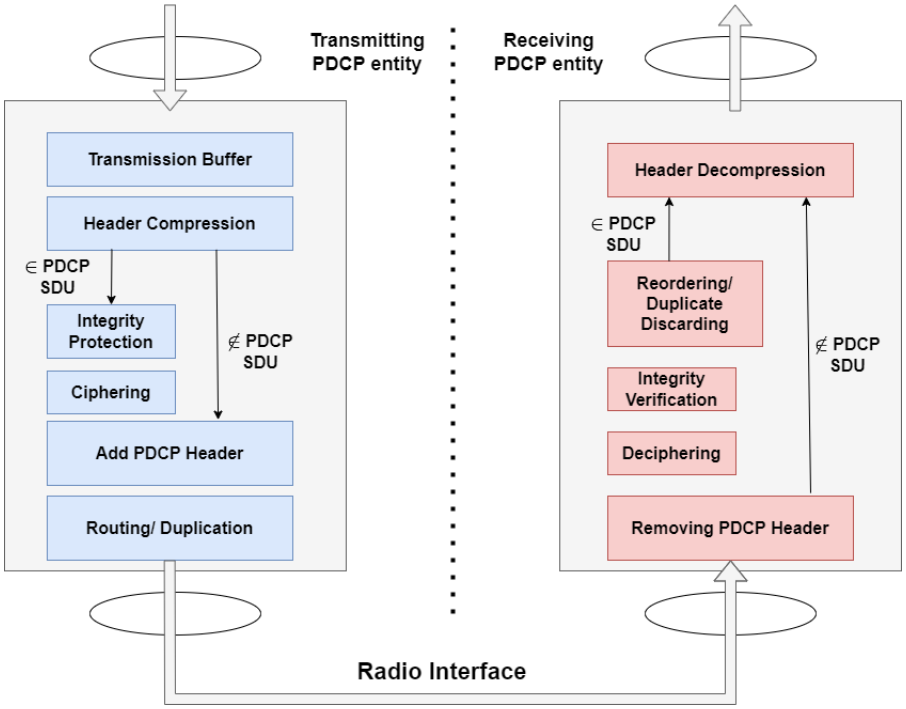

Figure B.4: *Functional review of PDCP layer* **[107]**

.

At the transmitter, it performs header compression using ROHC Protocol as well as ensures the in-sequence packet delivery and remove duplicated packets. At the receiver side, the PDCP performs the corresponding decompression operations. With respect to the PDCP operations, we implemented the following features in our simulation: (i) Transfer of data in control and data plane and (ii) Handling of PDCP sequence numbers (SNs) and their status.

B.3 Radio Link Control (RLC) Protocol

In order to control the transfer of upper layer Packet Data Unit (PDU) generated by PDCP layer, the RRC layer is sandwiched between PDCP and MAC layer. There are 3 modes of operations which are well-

TABLE B.1: *Support of RLC features in each RLC mode*

RLC features	TM	UM	AM
Transfer of upper layer PDUs	X	X	X
Error correction through retransmissions (ARQ)			X
Concatenation/Segmentation of RLC SDUs		X	
Resegmentation of RLC data PDUs			X
Reordering of RLC data PDUs		X	X
Duplicate detection		X	X
RLC SDU discard		X	X
RLC reestablishment	X	X	X
Protocol error detection			X

known in RLC layer: transparent mode (TM), unacknowledgement mode (UM) and acknowledgement mode (AM) in which each one is identical in terms of concatenation/segmentation, re-segmentation, duplicate detection and in-sequence delivery of service data units (SDUs). The features differences of operational modes can be briefly given in the Table B.1 below:

By deferring the retransmission features to the MAC and scheduling layer as well as to model a sufficientRLC functions, we are going to implement Unacknowledgement mode (UM) which is suitable for handling delay sensitive, real time data traffic **[108]**.

The functionalities of RLC in UM mode can be depicted in Figure B.5. At the transmitter, the RLC entity receives RLC PDUs from upper layer through a single UM-SAP, which connect PDCP and RLC layer together, and put them into transmission buffer. When a transmission opportunity triggered by MAC appears, it may perform concatenation/segmentation of RLC SDUs depending on the size of each SDU in transmission buffer and the size of opportunity (TB size). Afterwards, a header will be added to each PDU to mark the information fields (such as its length and SN) before being passed into MAC layer for the over-the-air transmission. Due to the retransmission, the reception order of each RLC PDU may not be maintained, so the UM receiving entity should create a receive buffer to keep them for reordering process purpose. Based on the information of SN in each PDU header, we are able to perform reordering process and reassemble, redo concatenation/segmentation process to recover SDUs in the right order.

In order to give an intuitive example of how reordering process is working receiver RLC entity, Figure B.6 illustrate the whole procedure with the support of 2 parameters which are UM_Window_Size and $T\_reordering$. In this example, amongst 4 PDUs numbering from 0 through 3 in which the transmitter sends (**circle 1**), PDUs 0 and 3 are corrupted by the bad channel. The receiver RLC entity triggers $T\_reordering$ timer once the gap due to PDU 0 is detected (**circle 2**). Thanks to HARQ retransmission protocol, PDU 0 is successfully recovered before $T\_reordering$ timer expires, thus, PDUs 0,1 and 2 are delivered to upper layer and reordering window advances (**circle 3**). After well receiving 3 more PDUs from 4 to 6 (**circle 5**) which transmitter sent (**circle 4**) before, $T\_reordering$ restarts due to the sequence gap caused by PDU 3. For some reason, we are not able to recover PDU 3 and $T\_reordering$ expires, the receiver will stop waiting for it and advance PDUs 4,5 and 6 to the upper layers. In this case, PDU 3 is considered as completely lost (**circle 6**).

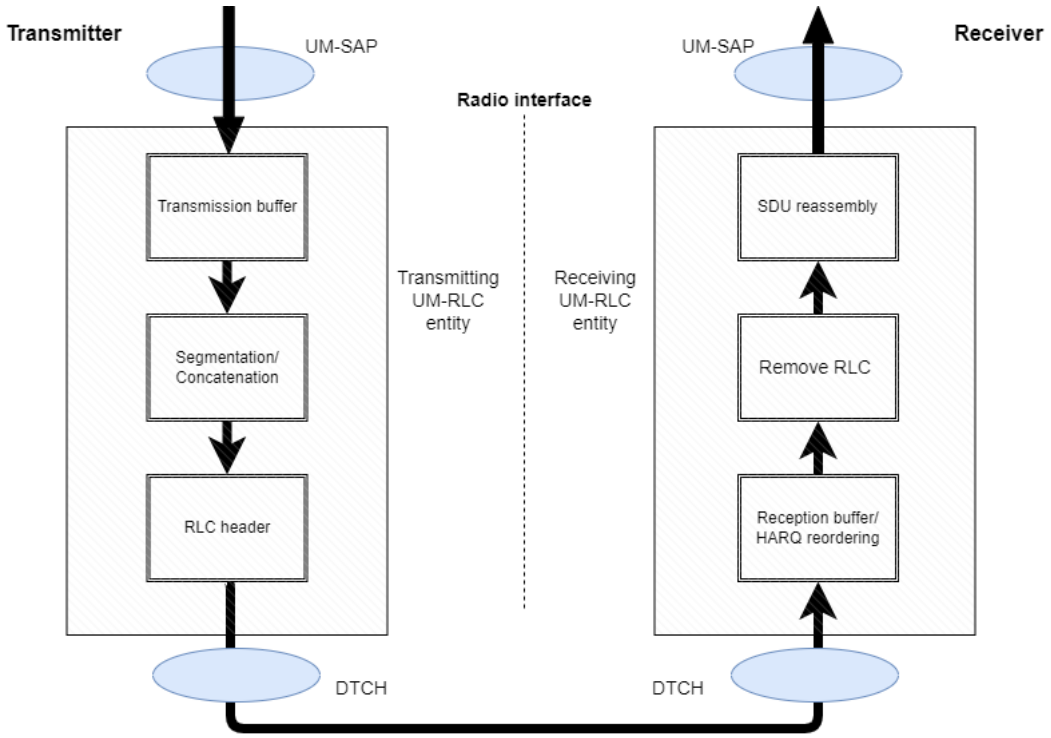

FIGURE B.5: *Illustration of RLC UM entities*

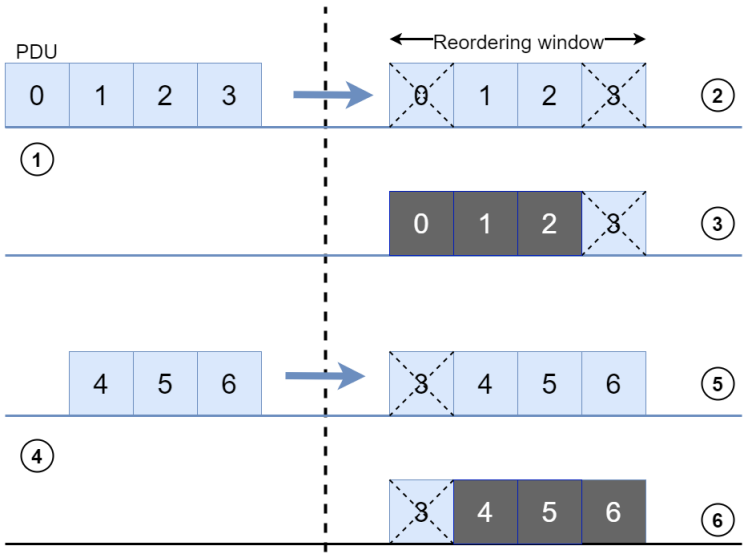

FIGURE B.6: *Illustration of RLC UM reordering process*

B.4 Medium Access Control (MAC) Protocol

In this section, we are going to detail the Medium Access Control (MAC) layer in our simulation, The MAC layer is responsible for:

- Channel Mapping: MAC layer maps logical channels carrying RLC PDUs to transport channels. For transmission, multiple SDUs from logical channel are mapped into the TB which are carried in the transport channels. In the receiver, TBs are demultiplexed and assigned to corresponding logical channels.
- Scheduling: MAC layer performs all related scheduling functions in the UL and DL directions. Thus, HARQ functionality will be managed by scheduler and it is responsible for exchanging scheduling related information with UEs.
- UL Timing Maintenance: Timing synchronisation between UEs and gNb needs to be maintained in the course of communication. Then, MAC layer is responsible for periodic synchronization.

Typically, UL and DL scheduling are centralized and controlled at gNb. DL scheduler will control which UEs to transmit and for each of them, the set of RBs upon which the user's PDSCH should be transmitted. The information exchanged includes transport-block size, modulation scheme and logical-channel multiplexing. On UL, the UEs are responsible for selecting from which radio bearers the data will be served according to their priorities. Then, a set of complementary information from UEs are essential (buffer status, channel measurement, etc.) for gNb performs scheduling functions. The process of UL / DL scheduling will be shown in Figure B.7.

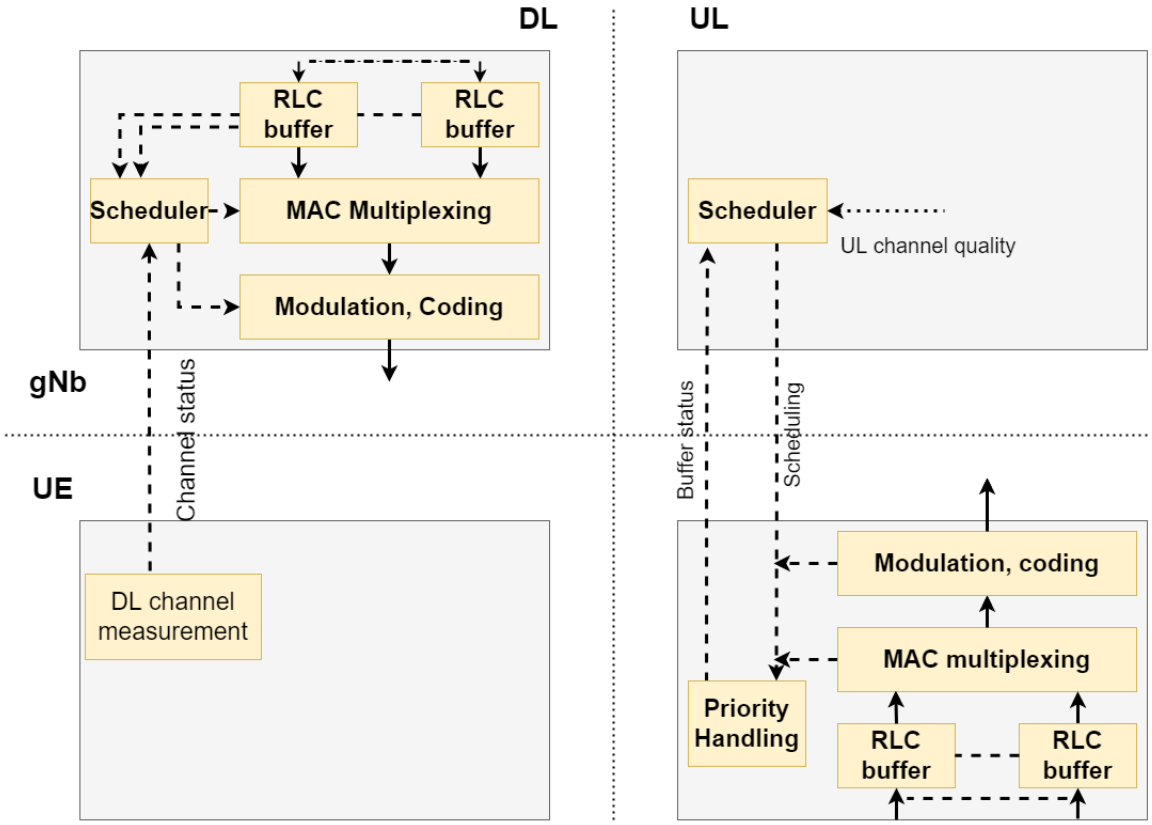

Figure B.7: *Downlink and Uplink scheduling*

Our simulator is inspired by a developed version from mmWave module [109], which additionally support sub-6GHz band and Time-Division Multiple Access (TDMA) as well as OFDMA schedulers. In

OFDMA, we simplify the implementation by restricting the single-beam capability, which means that UEs or gNbs are pointed by only one beam at a particular time instant. However, we do not limit which UEs are served in a single beam, so the scheduler at MAC layer can schedule these UEs at the same time in frequency domain. Basically, the allocation of resources in terms of time and frequency is related to the OFDM symbols and RBGs in side a slot, respectively. There are 3 allocation types that can be used to schedule UE. Figure B.8a depicts a TDMA-based scheme where three UEs equally occupy OFDM symbols with their data are spanned in all RBGs. In contrast, OFDMA-based scheduler allocate entire OFDM symbols for UEs but on different RBGs as in Figure B.8. Figure B.8c provides better level of flexibility with different TTI, allocating each UE with different RBGs in certain OFDM symbols. In order to identify the reserved resources for a particular user, a bitmask method is utilized, in which the indicator of **1's** to show that this resource is for an user u, otherwise, **0's**.

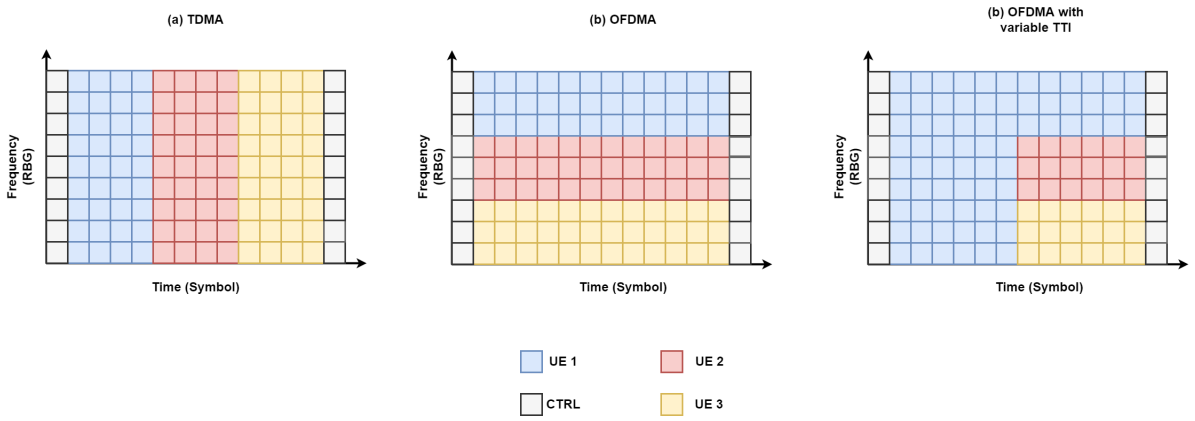

FIGURE B.8: *Allocation schemes for a Slot*

B.4.1 DYNAMIC SCHEDULING IN MAC LAYER

Hereafter, we are going to shed light on the conventional, centralized dynamic scheduled-based access.

In DL, whenever a new TB is created and ready, the scheduler decides to transmit them over the air. However, this operation is not instantaneous, but in a configurable delay of $L1L2delay = L_{12}$ representing the preparation and processing time at PHY/MAC layer of gNb side. Afterwards, each UE monitors the PDCCH at the beginning of each slot to detect whether a valid Downlink Control Information (DCI) is sent to them or not. Subsequently, the authorized UEs decode the DCI to see where they should look for their sending data in PDSCH. To determine whether the TB transmission is successful or not, UE will process the received data. In case of success, UE will forward that TB to higher layer for further processing and send acknowledgement message (ACK) to the gNb. Otherwise, the negative acknowledgement (NACK) will be sent to inform that the current TB is corrupted. The sending of ACK/NACK will be encoded into PUCCH and performed after a delay of $K2delay$. In order to align with the current 3GPP recommendation, the sending of PDCCH and PUCCH will be in charged by the first OFDM symbol and last symbol in a slot, respectively **[110]**. The DL scheduler timing is represented in Figure B.9.

In UL, upon data arrival at the UE RLC queues, the UE sends an SR to the gNb through PUCCH to request an UL grant from its attached gNb. Since there is no information of current buffer size in UE is sent to gNb in this stage, gNb sends an UL grant via DCI in PDCCH to indicate that there is an opportunity for the UE to transmit its buffer size information. In this regard, UE occupies 1 entire OFDM symbol to transmit its data and BSR via PUSCH. Afterthat, in case of a BSR is received at gNb, after performing resource scheduling for UEs which demand radio resources, gNb knows the UE RLC buffer status of each UE and accordingly provide them another opportunity to transmit their remaining data. The UL

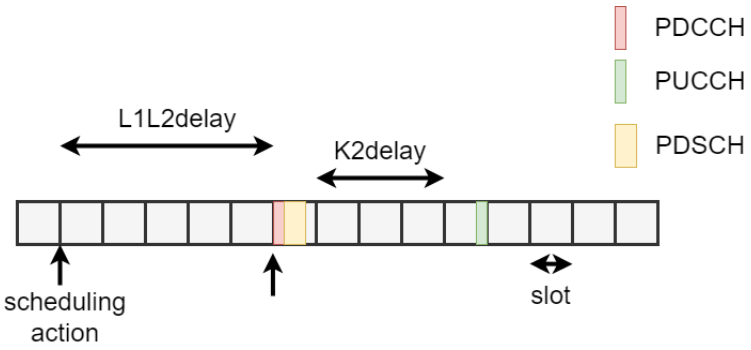

FIGURE B.9: *Scheduler timing for DL*

procedure is presented in Figure B.10 in which the processing delay in gNb and UE are also introduced. Before sending the UL grants, $L1L2delay = L_{12}$ has to be considered at the gNb side. Along with that, upon the reception of UL grant, the UE should sent its UL data and/or BSR after $K2delay$. The value of $K2delay$ illustrates the delay between UL grant reception in DL via PDCCH and corresponding UL data transmission via PUSCH. In our implementation, this delay is seen as the number of time slots required for the UE to decode UL grant in PDCCH and prepare its UL data to transmit in the indicated scheduling opportunity over PUSCH. According to NR specification, $K2delay$ may take any integer value from 0 to 32 slots **[111]**. On the other hand, $L1L2delay = L_{12}$ depicts the time that the PHY/MAC layer at gNb need to encode control and/or data channels and can be seen as a delay between control/data acquisition from the RLC class and the moment at which the control/data is available to be over the air. In short, the UL packet transmission ends up after 5-step process ($SR \rightarrow UL-grant \rightarrow UL-data+BSR \rightarrow UL-grant \rightarrow UL-data$).

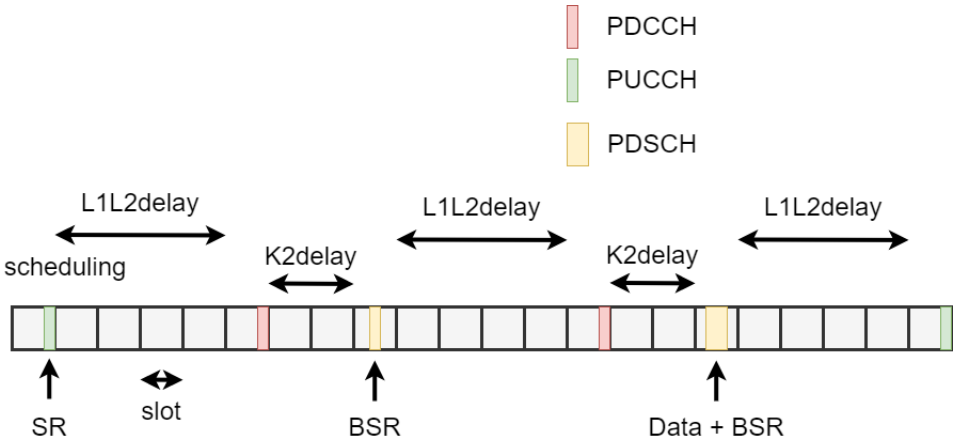

FIGURE B.10: *Scheduler timing for UL*

B.4.2 SCHEDULER

The scheduling process accounts for the resource assignments for active DL/UL flows and control the scheduling decisions at the MAC layer for of corresponding slots. The main output of centralized scheduler at gNb is a list of DCIs for a particular slot, each of which specifies three parameters: *(1)* The transmission starting symbol, *(2)* the duration (in number of symbols) and *(3)* an RBG bitmask , in which a value of 1 in the position x represents a transmission in the RBG number x. In UL/DL direction, the allocation of RBGs in time/frequency domain among active UEs is performed by using specific scheduling algorithms (e.g., round robin, proportional fair, maximum rate, etc). Scheduling procedures for TDMA-

based access can be easily obtained by first applying scheduling algorithms and then assign the numbers of OFDM symbol(s) for the allowed users. Then, the creation of bitmask for each user according to the scheduling phase will be done to initialize DCI control message.

For OFDMA based scheduler, different phases are performed:

- *(1)* Compute the link quality based on the CQI for the DL or SINR measurements for UL data channel. The corresponding MCS index will be derived then to determine the structure of RBGs.
- *(2)* Gather the active UL and DL flows in current slot, perform scheduling algorithm prioritizing UL flows over the DL flow.
- *(3)* If the HARQ retransmission in current slot is triggered, the resource allocation for the retransmission will be prioritized.
- *(4)* Both select an adequate number of OFDM symbol(s) and number of RBGs inside a symbol in a slot to assign to a specific user. The calculation of adequate resources will be based on the Buffer Status Report (BSR) of each user.

B.4.3 Transport block size (TBS) determination

In order to model the amount of data which is passed between PHY and MAC layer of transmitter and receiver, a MAC TB will be modelled as described in **[110]**. A TB is added with CRC and segmented into code blocks (CBs) as Figure B.11 below.

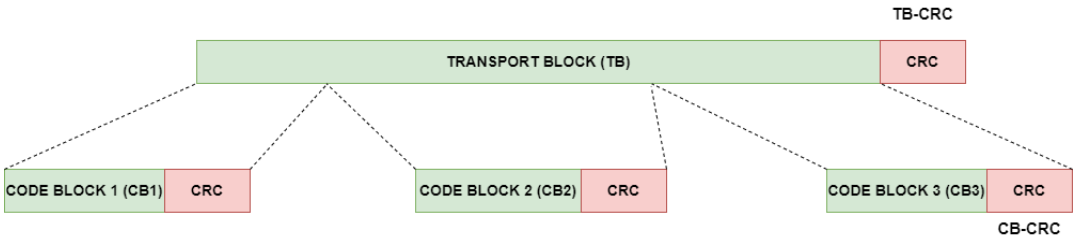

Figure B.11: *Illustration of Transport Block.*

The calculation of TB size in NR depends on the frame structure, as well as the selection of MCS index, resource allocation (in terms of OFDM symbols and RB). Follows the TS 38.214 **[110]**, the TB size will be computed as following:

$$TBS = R_e \times Q_m \times n_s \times n_{rb} \times (12 - n_{refSc}) \tag{B.1}$$

where R_e is the ECR of the selected MCS, Q_m is the modulation order of the selected MCS, n_s is the number of allocated OFDM symbols, n_{rb} is the number of allocated RBs and n_{refSc} is the number of reference subcarriers carrying reference signal per RB. By default, $n_{refSc} = 1$ is set as number of reference subcarriers carrying DeModulation Reference Signal (DMRS) per RB. The details on R_e, Q_m according to each MCS value can be presented in Table B.2.

Given a Bandwidth $BW[Hz]$ and Numerology Num, without considering the guard band between RB, the total number of RB can be calculated as follow:

$$n_{rb} = \frac{n_s \times BW[Hz]}{2^{Num} \times SCS_0[Hz]} \tag{B.2}$$

where $SCS_0 = 15kHz$ is the SCS in Hz at standard.

With the purpose of performing error detection in each TB or CB, an CRC attachment with the size of 24 bits will be reserved at the header of each TB or CB.

MCS Index m	Modulation Order Q_m	Code Rate R_e x 1024	Spectral efficiency	β-value
0	2	120	0.2344	1.6
1	2	157	0.3066	1.61
2	2	193	0.3770	1.63
3	2	251	0.4902	1.65
4	2	308	0.6016	1.67
5	2	379	0.7402	1.7
6	2	449	0.8770	1.73
7	2	526	1.0273	1.76
8	2	602	1.1758	1.79
9	2	679	1.3262	1.82
10	4	340	1.3281	3.97
11	4	378	1.4766	4.27
12	4	434	1.6953	4.71
13	4	490	1.9141	5.16
14	4	553	2.1602	5.66
15	4	616	2.4063	6.163
16	4	658	2.5703	6.5
17	6	438	2.5664	9.95
18	6	466	2.7305	10.97
19	6	517	3.0293	12.92
20	6	567	3.3223	14.96
21	6	616	3.6094	17.06
22	6	666	3.9023	19.33
23	6	719	4.2129	21.85
24	6	772	4.5234	24.51
25	6	822	4.8164	27.14
26	6	873	5.1152	29.94
27	6	910	5.3320	32.05
28	6	948	5.5547	34.28

TABLE B.2: *MCS index table 1 for TBS and SINR determination*

TABLE B.3: *5G New Radio numerologies*

Numerology	Slots/Subframe	Symbol Length (μs)	Slot length (ms)	SCS (kHz)	Symbols/Slot
0	1	71.42	1	15	14
1	2	35.71	0.5	30	14
2	4	17.85	0.25	60	14
3	8	8.92	0.125	120	14
4	16	4.46	0.0625	240	14
5	32	2.23	0.03125	480	14

B.5 PHYSICAL LAYER (PHY) PROTOCOL

B.5.1 FRAME STRUCTURE

In 5G NR, radio frame can be structured in 3 different types: **(i)** Frequency Division Duplex (FDD), **(ii)** Time Division Duplex (TDD) and **(iii)** Licensed-Assisted Access (LAA) (Unlicensed Spectrum). Typically, each of those frames contains 10 subframe of 1 ms length. Then, the flexibility of the frame design is achieved by the introduction of *numerology* concept. Traditionally, each subframe corresponds to a single slot in LTE network and SCS is fixed at 15 kHz. The supported numerologies (0, 1, 2, 3, 4, 5) in NR allow to divide each subframe into multiple slots (1, 2, 4, 8, 16, 32) with corresponding increased SCS (15 KHz, 30 kHz, 60 kHz, 120 kHz, 240 kHz, 480 kHz). In all the case, the number of OFDM symbols per slot is fixed to 14 symbols for normal CP and 12 symbols for extended CP. In frequency domain, the number of subcarriers per PRB is also fixed to 12 which follows the 3GPP Release 15 **[112]**. The details of each numerology is displayed in Table B.3.

In order to illustrate how frame structure in NR looks like, Figure B.12 shows the frame in time and frequency domain when numerology Num is configured for $num = 3$, and Bandwidth $400MHz$.

In FDD mode, 2 paired bandwidth parts are modelled where one is dedicated to transmit DL data and control, and the other for the transmission of UL data and control. Then, the configuration in both bandwidth parts are necessary to correctly routing the UL/DL control/data messages in appropriate channels. On the other hands, TDD model allows 3 different slot types: **(i)** DL-only slot, **(ii)** UL-only slot and **(iii)** Flexible slot (F). For the two first cases, only DL or UL data can be scheduled in these slots, correspondingly. For the latter, a certain number of DL symbols, a guard band and UL symbols can be configured according to a predefined pattern. In both TDD and FDD schemes, slot operation will follow *event scheduling feature*. Starting from time 0, events are inserted into event scheduler and the event processes will be executed to model time advancement.

- Time starts at the beginning of the slot.
- PHY layer of gNb get allocation information from gNb's MAC layer and extract them.
- gNb starts transmitting CTRL message to the corresponding users.
- Correspondingly, UEs begin receiving the CTRL messages (DCI information). Then, UEs can perform (i) receive data if they received DL DCI or (ii) send data if they received UL DCI.
- In UL or F slots, at the end of slot, there will be an opportunity for UEs to transmit their UL CTRL messages.
- gNb specifies this tume at the registration time and it is considered as the last operation of the slot.

Considering the processing of events in each time slot operation, there are 2 factors that we need

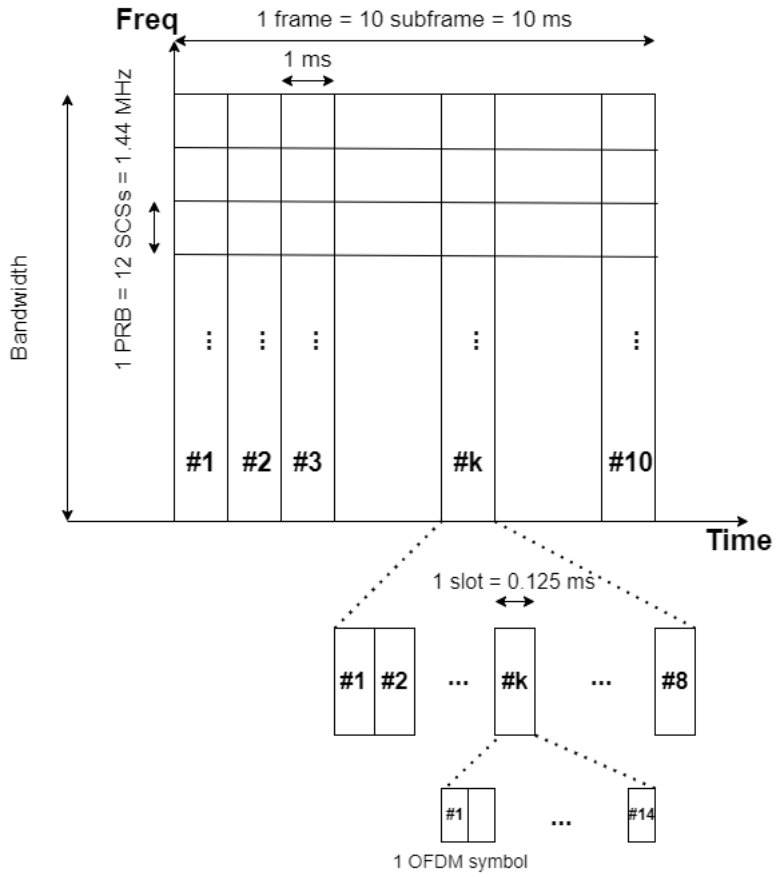

FIGURE B.12: *NR frame structure in time and frequency domain for* $Num = 3$.

to consider. The first is the availability of the channel. In TDD and FDD modes, the channel is always utilisable for the transmission, while in LAA this may not be true as the unlicensed spectrum is considered. In the case if unavailable channel, the event processes will be transferred to the next slot. The second is the MAC-to-PHY delay, i.e. $L1L2delay = L_{12}$ which indicates that the MAC is working ahead PHY to demonstrate the time needed for each component of the chain to perform its work.

B.5.2 DATA ERROR MODEL IN PHYSICAL LAYER

In this section, we are going to detail how to model error behaviours of PHY data transmission in system level simulation using PHY abstraction. Basically, the impacts of PHY layer to the wireless transmission are modelled by link-level simulation in which the processing blocks of modulation, channel coding, channel fading, channel estimation, demodulation are featured. However, in a system level simulation where the evaluations of average network performance are considered, the inclusion of these features into the PHY layer will be extremely complex and computationally heavy. In the level of system simulation, wide-band SINR and BLER of each active user in the wireless network are two most important parameters to insight into the system performance **[113]**. Then, the research on how to map the physical effects of link layer to the corresponding SINRs and BLER are studied. In OFDM and time-slotted based system, the TBs are transmitted over several sub-carriers whose SINRs are non-uniformed due to the time and/or frequency selectivity of the channel. Then, one of the traditional approaches to map several SINRs of sub-channels to an effective SNR is to average the multi post-processing SINRs which is known as ESM technique **[114]**. Basically, ESM technique is efficiently used for the purpose of PHY layer abstraction since it helps to reduce the computational complexity leveraging L2S method. In precedent

studies, EESM and mutual information based ESM are introduced for L2S method [115], [116], [117]. In [114], a DNN approach is used to accurately derive EESM for 5G New Radio (NR).

Given the SINR measurement on different sub-channel is carried out over the set of $\omega \in \Omega$ RBs (i.e., γ_ω), the effective SINR (i.e., γ_{eff}) can be described as follows [113]:

$$\gamma_{eff} = -\beta \times \Phi^{-1}\left(\frac{1}{|\Omega|}\sum_{\omega\in\Omega}\Phi\left(\frac{\gamma_\omega}{\beta}\right)\right) \tag{B.3}$$

where the function $\Phi(.)$ is derived from the Chernoff bound on the probability of error [118]. β is a parameter represented as a shift of $\Phi(.)$ to adapt the model to difference MCS. In case of MIMO scheme with linear processing (ZF or MMSE) is considered and a good estimation of MIMO channel matrix is assumed at the receiver, the post-processing SINR can be calculated based on well-known expression. In case of EESM, the effective SINR in the retransmission rth can be written as follows [113]:

$$\gamma_{eff}{}^{r}(\beta) = \beta \times \log\left(\frac{1}{|\Omega|}\sum_{\omega\in\Omega}\exp\left(\frac{\gamma_\omega{}^{r}}{\beta}\right)\right) \tag{B.4}$$

The value of β is obtainable by minimizing the mean square error between estimated BLER and simulated BLER using numerical methods as follows [113], [71]:

$$\hat{\beta} = \arg\min_{\beta}\{\sum_{r}\left[\log_{10}(BLER_i) - \log_{10}\left(BLER_R\left(\gamma_{eff}{}^{r}(\beta)\right)\right)\right]\} \tag{B.5}$$

where $BLER_R(.)$ is the reference BLER curve in the AWGN channel for the current MCS and CBS.

Inspired by the work on [71], the general PHY abstraction model that we used is shown in Figure B.13. The inputs for the model consist of: *(i)* a vector of SINRs per allocated RB, *(ii)* the MCS index which is the output of AMC model and its corresponding parameters, *(iii)* the calculated TBS and *(iv)* HARQ history which stores the computed SINR of each RB for the previous (re)transmission. The output of this model will be calculated Transport BLock Error Rte (TBLER).

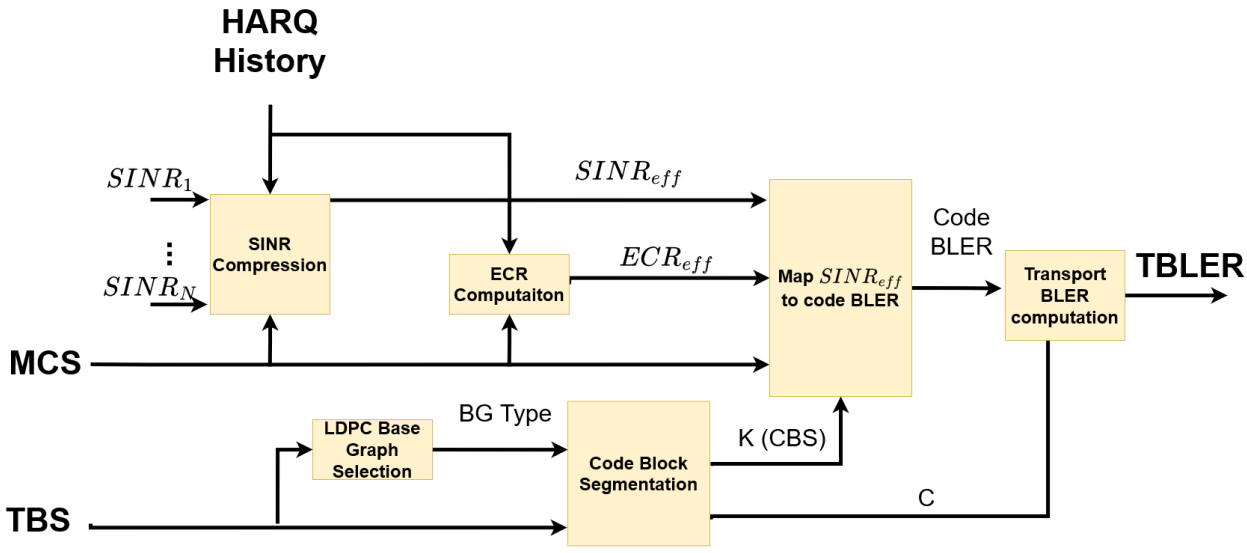

FIGURE B.13: *NR PHY abstraction model.*

First and formost, HARQ history stores SINR per allocated RB or the last computed effective SINR and number of coded bits for each of previous retransmission depending on HARQ Chase Combining (HARQ-CC) or HARQ-IR method. In HARQ-CC, the ECR remains the same as rth retransmissions are done. Thus, the the combined SINR values of corresponding resources ω over r retransmissions (i.e., $\gamma_{eff,i}$

$\forall i \in r$) will be summed across the retransmission. Then, the effective SINR using EESM is computed as [119]

$$\gamma_{eff}{}^{r} = -\beta \times \ln\left(\frac{1}{|\Omega|} \times \sum_{\omega \in \Omega} \exp\left(-\frac{1}{\beta}\sum_{i=1}^{r} \gamma_{eff,i}\right)\right) \tag{B.6}$$

On the other hand, in HARQ-IR, every retransmission contains different coded bits than the previous ones. In this case, both ECR and effective SINR require the recomputation after each retransmission. It is derived as following:

$$\gamma_{eff}{}^{r} = -\beta \times \ln\left(\frac{1}{|\Omega|} \times \sum_{\omega \in \Omega} \exp\left(-\frac{\gamma_{\omega}{}^{r} + \gamma_{eff}{}^{r-1}}{\beta}\right)\right) \tag{B.7}$$

Afterwards, the effective ECR and SINR will be used to calculate CBLER.

In order to calculate CBLER and TBLER, several supplementary information which corresponds to MCS, CBS and LDPC Base Graph are needed as the illustration in Figure B.13.

Following the 3GPP Recommendation, several related parameters corresponding to each MCS index be classified into 3 Tables: Table 1 (up to 64-QAM), Table 2 (up to 256-QAM) and Table 3 (up to 64-QAM with low spectral efficiency) [110]. Given a particular MCS and selected table, a modulation order m, spectral efficiency and ECR can be found. These useful information will be subtle to map effective SINR to the related CBLER.

In the block of SINR Compression, an optimal value of β is found using calibration technique in [71] and [114]. The values of β are derived as the Table B.4.

In the NR PHY abstraction model, LDPC BG is essential to determine the maximum CBS possible in using LDPC coding [72]. The maximum derived CBS will be useful to see if the input TBS needs the segmentation into multiple CBSs or not. The selection of BG will be as follows:

- LDPC BG2 is selected if TBS $\leq$ 292 or ECR$\leq$ 0.25 or TBS $\leq$ 3824 with ECR $\leq$ 0.67
- Otherwise, LDPC BG1 will be selected.

Code Block Size (CBS) is a procedure which is necessary for LDPC coding when the number of total bits in TB comprising of CRC is larger than maximum CBS (K_{cb}). When BG1 is selected, $K_{cb} = 8448$ bits and $K_{cb} = 3840$ bits if BG2 is chosen. When CBS takes place, each TBS is divided into C CBS of K bits each. For each code block, an additional CRC sequence of $L = 24$ bits is put on the header to recover the original TB during decoding process. The whole process of CBS will be detailed in [72].

Once we finish the calculation of effective SINR, ECR and CBS, the last step is to transform those values into a representative BLER. In order to obtain SINR-BLER mapping, an extensive simulation using NR-compliant Link Level Simulation has been done according to reference [114], [71]. Then, the mapping process depends on the resource allocation, where the corresponding values of TBS, CBS, LDPC BG selection are derived, and it results in various values of BLER according to the allocation. In order to have a complete view of this curve for different CBS, BG type, MCS index, Patriciello et al. in [120] showed these results in their research with the note that Code Block Size (CBS) is not included in BLER calculation. Therefore, we need to convert these results, which are basically obtained from code BLER, into TBLER. Given the segmentation of TB into C code blocks, then the ultimate TBLER of this TB will be derived from each CBLER i as follows:

$$TBLER = 1 - \prod_{i=1}^{C}(1 - CBLER_i) \approx 1 - (1 - CBLER)^{C} \tag{B.8}$$

Where the last approximation assumes that the code blocks are generated with equal sizes.

TABLE B.4: *Optimal β values for each MCS*

MCS	Optimal β		**MCS**	Optimal β	
	Table 1	Table 2		Table 1	Table 2
0	1.60	1.60	15	6.16	19.33
1	1.61	1.63	16	6.50	21.85
2	1.63	1.67	17	9.95	24.51
3	1.65	1.73	18	10.97	27.14
4	1.67	1.79	19	12.92	29.94
5	1.70	4.27	20	14.96	56.48
6	1.73	4.71	21	17.06	65.00
7	1.76	5.16	22	19.33	78.58
8	1.79	5.66	23	21.85	92.48
9	1.82	6.16	24	24.51	106.27
10	3.97	6.50	25	27.14	118.74
11	4.27	10.97	26	29.94	126.36
12	4.71	12.92	27	32.05	132.54
13	5.16	14.96	28	34.28	-
14	5.66	17.06			

APPENDIX C

Spatial Channel Model for Indoor Factory (InF) Scenario

Along with the development of wireless technologies for 5G and beyond systems, a comprehensive channel modelling is of paramount importance to characterize a correct network behaviour. Currently, in order to evaluate novel algorithms in analytical studies, the simple propagation models, which statistically characterize Rayleigh, Nakagami-m **[121]** or Rayleigh fading, are usually assumed. These models are relatively simple, lightweight and do not capture the spatial dimensions of the channel. Hence, the multiple critical features related to the beam are merely omitted and do not precise the realistic behaviour of current systems. On the other hands, quasi deterministic channel model **[122]** is able to generate realistic radio channel impulse responses for system-level simulations of mobile radio networks at the cost of computation complexities and requiring a precise characterization of the environment.

In order to design a computationally efficient channel model which well represents the spatial characteristic of the beam, the Spatial Channel Model (SCM) which characterizes the channel behaviour between 0.5 and 100 GHz, is considered. The whole procedure of channel coefficient generation for SCM is illustrated in Figure C.1.

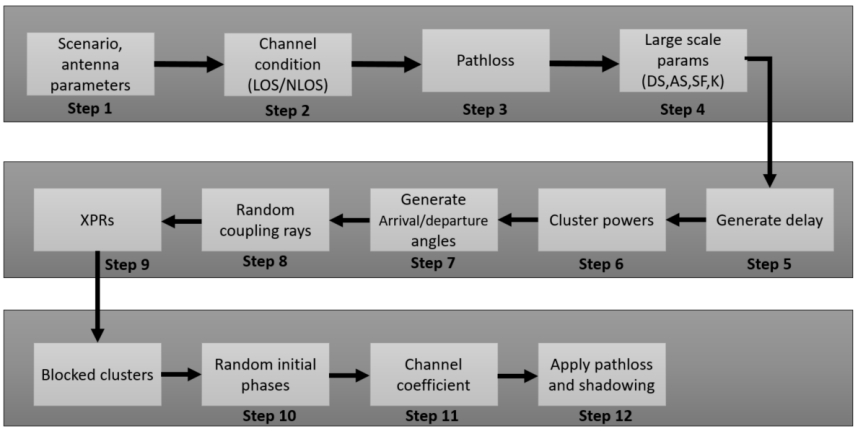

FIGURE C.1: *Channel coefficient generation procedure.*

In the first step , 3D distribution of gNbs and UEs in the InF scenario and their antenna parameters are set up. It enables us to determine the distances, working frequency band, AOD, ZOD, AOA, ZOA of each gNb and UE in the GCS.

In step 2 , the propagation conditions (LOS/NLOS) for different gNb-UE links are assigned.

In step 3 , the propagation pathloss (LOS/NLOS) for each gNb-UE is calculated.

In step 4 , large scale parameters, e.g. DS, angular spreads (ASA, ZSD, ZSA, ZSD), K and SF, are generated.

In step 5 and 6 , cluster delay and power are generated.

In step 7 , arrival angles and departure angles for both azimuth and elevation are generated.

In step 8 , coupling of rays within a cluster for both azimuth and elevation.

In step 9 , cross polarization power ratios are generated.

In step 10 , random initial phase for each ray of each cluster will be drawn.

In step 11 , channel coefficients for each cluster and each receiver and transmitter element pair (u, s) are generated.

In the last step , pathloss and shadowing for the channel coefficients will be applied.

C.1 Coordinate system

In order to characterise 3D channel model, a coordinate system, which is formed by x,y,z axes, the spherical angles and spherical unit vectors, will be described. In Cartesian coordinate system, there are zenith angle θ and azimuth angle ϕ as the illustration in Figure C.2. The orthonormal basis vector of the Cartesian coordinate system are $\hat{n}, \hat{\theta}$ and $\hat{\phi}$.

Basically, the propagation of signal between transmitter and receiver is scattered by MPC whose is characterised by its delay and direction of arrival/departure. By grouping MPCs which share the similar delays and directions into a group, it forms a cluster of MPCs. Thus, we assume that there are N clusters which is stochastically embedded in our scenario , and there are M rays in each cluster $n \in N$

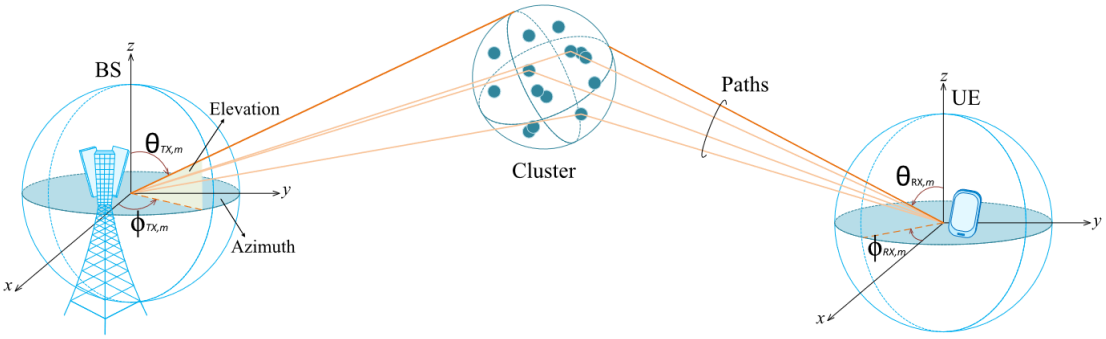

Figure C.2: *Scattering concept in 3D channel model [123].*

Antenna array for a gNb or an user is generally defined in a LCS. Such a system is used as reference to describe the far-field radiation power pattern and polarization of each element in the array. Besides, a network consists of multiple gNbs and UEs are normally sketched in GCS. Considering a network with a GCS defined with coordinates $(x, y, z, \theta, \gamma)$,$\hat{n}, \hat{\theta}$ and $\hat{\phi}$ and a LCS system with primed-coordinate $(x', y', z', \theta', \gamma')$,$\hat{n}', \hat{\theta}'$ and $\hat{\phi}'$, we can see that the antenna element radiation pattern of LCS and GCS are the same.

$$A(\theta, \phi) = A'(\theta', \phi') \tag{C.1}$$

We will define then the polarized field components in LCS and GCS as $F'_{\phi'}(\theta', \phi')$, $F'_{\theta'}(\theta', \phi')$ and $F_{\phi}(\theta, \phi)$ $F_{\theta}(\theta, \phi)$ which are related by:

$$\begin{pmatrix} F_{\theta}(\theta, \phi) \\ F_{\phi}(\theta, \phi) \end{pmatrix} = \begin{pmatrix} \hat{\theta}(\theta, \phi)^T \mathbf{R} \hat{\theta}'(\theta', \phi') & \hat{\theta}(\theta, \phi)^T \mathbf{R} \hat{\phi}'(\theta', \phi') \\ \hat{\phi}(\theta, \phi)^T \mathbf{R} \hat{\theta}'(\theta', \phi') & \hat{\phi}(\theta, \phi)^T \mathbf{R} \hat{\phi}'(\theta', \phi') \end{pmatrix} \times \begin{pmatrix} F'_{\theta'}(\theta', \phi') \\ F_{\phi'}(\theta', \phi') \end{pmatrix} \tag{C.2}$$

where $\hat{\theta}$ and $\hat{\phi}$ are the spherical unit vectors of the GCS and likewise $\hat{\theta}'$ and $\hat{\phi}'$ of the LCS, respectively. **R** is the rotation matrix transforming the LCS unit vectors to the GCS reference frame. The unit vector $\hat{\theta}'$ and $\hat{\phi}'$ are given by:

$$\hat{\theta} = \begin{pmatrix} cos\theta cos\phi \\ cos\theta sin\phi \\ -sin\phi \end{pmatrix} \tag{C.3}$$

and

$$\hat{\phi} = \begin{pmatrix} -sin\phi \\ cos\phi \\ 0 \end{pmatrix} \tag{C.4}$$

C.2 Scenarios

The first step of generating a 3D channel model is to identify the scenario of interest. In this thesis, we concentrate on the indoor factory (InF) scenarios which focuses on factory halls of varying sizes and clutter density (e.g. machinery, assembly lines, storage shelves, etc). The parameters for InF will be derived as the table C.1. Considering the effect of the antenna height between gNb and UE, as well as the clutter density, there are five sub-scenarios of InF in which four first cases are related to NLOS communication and the last one refers to the LOS transmission.

- Sparse clutter, low BS (InF-SL)
- Dense clutter, low BS (InF-DL)
- Sparse clutter, high BS (InF-SH)
- Dense clutter, high BS (InF-DH)
- High Tx, high Rx (InF-HH)

In InF scenarios, the objects in the environment are metallic machines, thus, their huge architectures cause signal blockage as in the NLOS cases. Because the machine heights change according to their types, so we defined "clutter-embedded" and "above clutter" to describe the relative heights of the gNbs and UEs. In order to make channel model for InF more practical, the clutter density is described which demonstrate the clutter-occupied ratio in the surface area. Its density are classified into 2 types, sparse clutter and dense ones which define the ratio threshold of 0.4.

C.3 Antenna modelling

Also in the first step, we need to deal with antenna configurations for gNb and user. The antenna is modelled as a Uniform Planar Array (UPA) which constitutes of rectangular panels. The conventional deployment of multi antenna arrays is the use of cross-antenna panels with $+/-45$ degree polarisation. Each rectangular panel is uniformly spaced in the horizontal direction and vertical direction with a spacing of d_H and d_V, respectively. Within each antenna panel, there are $M \times N$ antenna elements per polarisation, which are organised in M columns and N rows, to concentrate the transmission within a narrow beam-width in the vertical and horizontal direction, as illustrated in Figure C.3. For each antenna element, the dual polarisation is also taken into account, so each element can be either single polarised or dual polarised.

TABLE C.1: *Evaluation parameters for Indoor Factory scenario*

		Indoor Factory				
Parameters		InF-SL	InF-DL	InF-SH	InF-DH	InF-HH
Layout	Room size	Rectangular: 20 - 160000 m^2				
	Ceiling height	5-25 m	5-15 m	5-25 m	5-15 m	5-25 m
	Effective clutter height h_c	Below ceiling height, 0-10 m				
	External wall and ceiling type	Concrete or metal walls and ceiling with metal windows				
Clutter type		Big machines composed of regular metallic surface	Small to medium metalic machine with irregular structure	Big machine composed of regular metallic surface	Small to medium metalic machine with irregular structure	Any
Typical clutter size $d_{clutter}$		10 m	2 m	10 m	2 m	Any
Clutter density **r**		Low (< 0.4)	High ($\leq$ 0.4)	Low (< 0.4)	High ($\leq$ 0.4)	Any
gNb antenna height		Clutter-embedded		Above clutter		
UE location	LOS/NLOS	NLOS and LOS				LOS only
	UE height	Clutter embedded				Above clutter

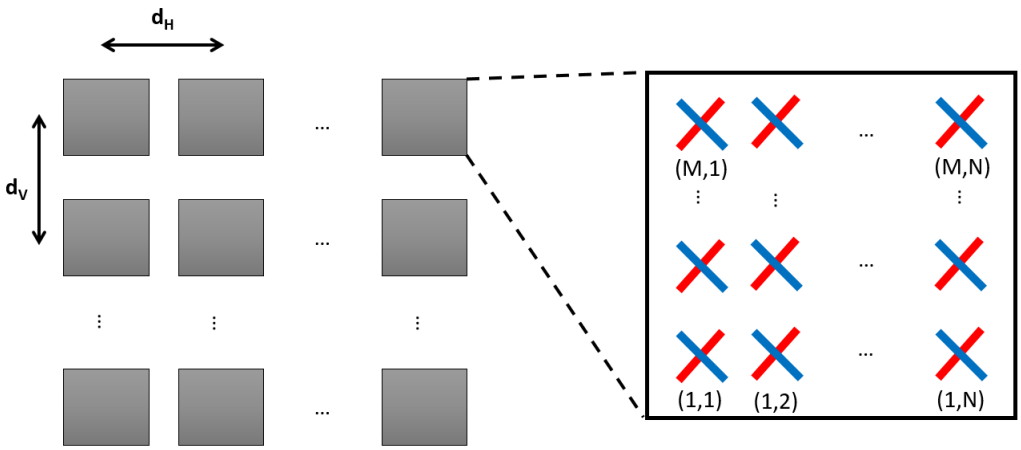

FIGURE C.3: *Cross-polarized panel array antenna model.*

The effect of antenna is modelled by its radiation field pattern **F** or radiation power pattern **P**. Depending on the direction (θ, ϕ) of the incoming or outgoing wave, the complex gain at certain direction (θ, ϕ), given the polarization will be expressed as following:

$$F(\theta, \phi) = \begin{cases} F_\theta(\theta, \phi) \\ F_\phi(\theta, \phi) \end{cases} \tag{C.5}$$

Then, the radiation power pattern $P(\theta, \phi)$ in linear scale will be defined as:

$$P(\theta, \phi) = |F_\theta(\theta, \phi)|^2 + |F_\phi(\theta, \phi)|^2 \tag{C.6}$$

C.4 Pathloss and LOS probability

In step 2, 3 and 4, we are going to get through the Pathloss model as well as the determination of LOS/ NLOS link state for the InF scenario. Concerning the pathloss model, the alpha-beta-gamma (ABG) freespace model is used in our system level simulation where the pathloss $PL(f_c, d_{3D})$ is calculated as following:

$$PL(f_c, d_{3D}) = \beta + 10\alpha \log_{10} \frac{d_{3D}}{d_0} + +10\gamma \log_{10} \frac{f_c}{f_0} + X_\sigma \tag{C.7}$$

where α is the distance-dependent exponent, β is the intercept, and γ is the frequency-dependent exponent. The variable X_σ is the shadow fading which follows a normal distribution with zero mean value. Its standard deviation is denoted by σ.

After collecting data for each sub-scenario, 3GPP TR 38.901 [44] specifies the pathloss as following:

- *For InF-HH (LOS) case:*

$$PL(f_c, d_{3D}) = 31.84 + 21.50 \log_{10} d_{3D} + 19 \log_{10} f_c + X_\sigma \tag{C.8}$$

$$\sigma = 4.3$$

- *For InF-SL case:*

$$PL(f_c, d_{3D}) = 33 + 25.50 \log_{10} d_{3D} + 20 \log_{10} f_c + X_\sigma \tag{C.9}$$

$$\sigma = 5.7$$

- *For InF-DL case:*

$$PL(f_c, d_{3D}) = 18.6 + 35.7 \log_{10} d_{3D} + 20 \log_{10} f_c + X_\sigma \tag{C.10}$$

$$\sigma = 7.2$$

- *For InF-SH case:*

$$PL(f_c, d_{3D}) = 32.4 + 23 \log_{10} d_{3D} + 20 \log_{10} f_c + X_\sigma \tag{C.11}$$

$$\sigma = 5.9$$

- *For InF-DH case:*

$$PL(f_c, d_{3D}) = 33.63 + 21.9 \log_{10} d_{3D} + 20 \log_{10} f_c + X_\sigma \tag{C.12}$$

$$\sigma = 4.0$$

In order to determine the visibility of connection between gNb and UEs, a probability model is used to determine whether a channel state is LOS or NLOS. Given the fact that the higher distance between gNb and UE is, the NLOS link states will be more likely.

The LOS probability, e.g $Pr_{LOS,subsce}$ under clutter densities and 2D distance d_{2D} is shown in the latest 3GPP 38.901 technical report as following:

$$Pr_{LOS,subsce}(d_{2D}) = \exp\left(-\frac{d_{2D}}{k_{subsce}}\right) \tag{C.13}$$

where

$$k_{subsce} = \begin{cases} -\frac{d_{clutter}}{ln(1-r)} & \text{for subsce} \in \{InF-SL, InF-DL\} \\ -\frac{d_{clutter}}{ln(1-r)}\frac{h_{gNb}-h_{UE}}{h_c-h_{UE}} & \text{for subsce} \in \{InF-SH, InF-DH\} \end{cases} \tag{C.14}$$

where the value range of clutter density r is from 0 to 1. The parameters $d_{clutter}, h_c, h_{UE}$, and h_{gNb} are the clutter size, clutter height, antenna height of UE, and antenna height of gNb, respectively.

B.5. Fast fading model

The fast fading model accounts for the changes in the phase and amplitude of the transmitted signal due to the effect of multipath propagation, i.e., the presence of multiple signal components that propagate over different paths.

In order to show the time-varying fluctuations of wireless channels which are caused by the statistical combination of multipath and user mobility, the fast fading channel coefficients will be applied in our system level simulation. The channel coefficients of a link between transmitter and receiver will be established by the channel impulse responses of many MPC. Each MPC is characterize by a path delay, a path power and random phase coupling, Angles Of Departure (AOD), Angles Of Arrival (AOA), Zenith Angles Of Departure (ZOD) as well as Zenith Angles Of Departure (ZOA).

In InF scenario, many metallic machines might be placed within the propagation path of the signal, thus the multiple path communication is seem more likely and lead to the increase of Root Mean Square (RMS) Delay Spread (DS). In NLOS case, when a large number of metallic materials are found between the signal propagation, the RMS delay spread may decrease from 664 ns to 10 ns **[124]**. It is assumed that the RMS delay spread is correlated to the factory volume V and surface area S.

Starting from step 5, we are going to generate delay for each cluster n. At first, under exponential distribution, path delays are drawn randomly as following:

$$\tau_n' = -r_\tau \sigma_\tau \ln(X_n) \tag{C.15}$$

where σ_τ the RMS delay spread, r_τ is the delay distribution proportionality factor, $X_n \sim (0,1)$, and cluster index $n = 1, ..., N$. With uniform delay distribution the delay values tau_n' are drawn from the corresponding range. Normalise the delays by subtracting the minimum delay and sort the normalised delays to descending order:

$$\tau_n = sort(\tau_n' - \min(\tau_i')_{i=1}^N) \tag{C.16}$$

In the case of LOS condition, additional scaling of delays is required to compensate for the effect of LOS peak addition to the delay spread. The heuristically determined Ricean K-factor dependent scaling constant C_τ is:

$$C_\tau = 0.775 - 0.0433K + 0.002K^2 + 0.000017K^3 \tag{C.17}$$

where K [dB] is the Ricean K-factor as given in C.4.

Then, the scaled delays in LOS is

$$\tau_n^{LOS} = \tau_n / C_\tau \tag{C.18}$$

TABLE C.2: *Scaling factors for AOA,AOD*

# clusters	4	5	8	10	11	12	14	15	16	19	20	25
C_ϕ^{NLOS}	0.77920	0.860	1.018	1.090	1.123	1.146	1.190	1.211	1.226	1.273	1.289	1.358

In step 6 we will initialize the second set of small scale parameters which is the cluster power, denoted by P_n. For each cluster n, we first compute:

$$P'_n = exp(-\tau_n \frac{r_\tau - 1}{r\tau.DS}) 10^{-\frac{Z_n}{10}} \tag{C.19}$$

where $Z_n \sim \mathcal{N}(0, \zeta^2)$, ζ represents the per-cluster shadowing and can be found in Table C.4. In order to have the sum of the cluster powers equal to one, we divide the previous power set P'_n by the total power and so we obtain the normalized set P_n

$$P_n = \frac{P'_n}{\sum_n P'_n} \tag{C.20}$$

In the case of LOS condition an additional specular component is added to the first cluster. Power of the single LOS ray is:

$$P_{1,LOS} = K_R/(1 + K_R) \tag{C.21}$$

where K_R is the Ricean K-factor converted to linear scale. Therefore, in the case of LOS, we need to change the normalization strategy as follows:

$$P_n = \frac{1}{1 + K_R} \frac{P'_n}{\sum_n P'_n} + \sigma(n-1) P_{1,LOS} \tag{C.22}$$

After the normalization, we remove the clusters with less than -25 dB power compared to the maximum cluster power. Once we have the cluster powers P_n, we assign the power to each ray by splitting P_n equally over the total number of rays, so $P_{n,m} = P_n/M_{ray}$, where $n = 1, ..., N$ is the cluster index and $m = 1, ..., M_{ray}$ is the ray index.

In order to accurately represent the spatial characteristics of the wireless channel, angular spread will also be described. According to several studies **[125] [126]**, the increases of angular spread is aligned with bigger factory (high V), higher density cluster and frequency-dependant. ***Step 7*** will detail the initialisation of arrival/departure angles for both azimuth ϕ and elevation θ.

For each cluster nn the azimuth angle of arrival (AOA), e.g. $\phi'_{n,AOA}$ is determined by its normalized cluster power P_n and RMS angle spread ASA:

$$\phi'_{n,AOA} = \frac{2(ASA/1.4)\sqrt{-\ln(P_n/\max(P_n)}}{C_\phi} \tag{C.23}$$

where C_ϕ defined as:

$$C_\phi = \begin{cases} C_\phi^{NLOS}(1.1035 - 0.028K - 0.002K^2 + 0.0001K^3) & \text{for LOS} \\ C_\phi^{NLOS} & \text{for NLOS} \end{cases} \tag{C.24}$$

where C_ϕ^{NLOS} is defined as a scaling factor related to the total number of clusters as given Table C.2. The additional scaling in the LOS case depending on the Ricean K factor and to compensate the effect of LOS peak addition to the angle spread.

Then, the transformation from $\phi'_{n,AOA}$ to $\phi_{n,AOA}$ is derived:

$$\phi'_{n,AOA} = C_n.\phi'_{n,AOA} + Y_n + \phi_{LOS,AOA} \tag{C.25}$$

TABLE C.3: *Ray offset angles within a cluster, given for rms angle spread normalised to 1*

Ray number m	Basis vector of offset angles α_m
1,2	±0.0447
3,4	±0.1413
5,6	±0.2492
7,8	±0.3715
9,10	±0.5129
11,12	±0.6797
13,14	±0.8844
15,16	±1.1481
17,18	± 1.5195
19,20	±2.1551

where random variable $X_n \sim \mathcal{U}(-1,1)$ and $Y_n \sim \mathcal{N}(0,(ASA/7)^2$. and $\phi_{LOS,AOA}$ is the direction of LOS which is assumably known. By default, there are $M = 20$ rays are initialized for each cluster n, so the cluster angles will be:

$$\phi_{n,m,AOA} = \phi_{n,AOA} + c_{ASA}\alpha_m \tag{C.26}$$

where c_{ASA} is the cluster wise RMS azimuth spread of arrival angles (ASA) which its value is given in Table C.4. The values of offset angle α_m for each ray m in a cluster is given in Table C.3 below.

Similarly, we could obtain the angular spread values for ZOA and ZOD as following:

$$\phi_{n,m,ZOA} = \phi_{n,ZOA} + c_{ZSA}\alpha_m \tag{C.27}$$

$$\phi_{n,m,ZOD} = \phi_{n,ZOD} + \frac{3}{8}(10^{\mu_{lgZSD}})\alpha_m \tag{C.28}$$

where μ_{lgZSD} is the mean of ZSD log-normal distribution, c_{ZSA} A is the cluster-wise rms spread of ZOA (cluster ZSA) in which their values are given in Table C.4.

In step 8, once we have all the per-ray angles, we need to random couple them within a cluster in the following way:

- Randomly couple AOD angles $\phi_{n,m,AOD}$ to AOA $\phi_{n,m,AOA}$ within a cluster n.
- Randomly couple ZOD angles $\phi_{n,m,ZOD}$ to ZOA $\phi_{n,m,ZOA}$ within a cluster n.
- Randomly couple AOD angles $\phi_{n,m,AOD}$ to ZOD $\phi_{n,m,ZOD}$ within a cluster n.

In step 9, the last step for small scale parameters, we are going to create the cross polarization power ratios (XPR) ,i.e $\kappa_{n,m}$ for each ray m of each cluster n.

$$\kappa_{n,m} = 10^{X_{n,m}/10} \tag{C.29}$$

where $X_{n,m} \sim \mathcal{N}(\mu_{XPR}, \sigma^2_{XPR})$ is Gaussian distributed and $(\mu_{XPR}, \sigma_{XPR}) = (12,6)$ if this link is LOS, otherwise, $(\mu_{XPR}, \sigma_{XPR}) = (11,6)$.

In step 10, we need to randomly draw the initial phases $\left\{\Phi^{\theta\theta}_{n,m}, \Phi^{\theta\phi}_{n,m}, \Phi^{\phi\theta}_{n,m}, \Phi^{\phi\phi}_{n,m}\right\}$ for each ray m of each cluster n and for four different combinations $(\theta\theta, \theta\phi, \phi\theta, \phi\phi)$.

TABLE C.4: *Fast fading parameters for InF*

Scenarios		**InF**	
		LOS	**NLOS**
Delay Spread	μ_{lgDS}	$\log_{10}(26\frac{V}{S}+14)-9.35$	$\log_{10}(30\frac{V}{S}+32)-9.44$
lgDS = $\log_{10}(DS/1s)$	σ_{lgDS}	0.15	0.19
AOD spread (ASD)	μ_{lgASD}	1.56	1.57
$lgASD = \log_{10}(ASD/1^o)$	σ_{lgASD}	0.25	0.2
AOA spread (ASA)	μ_{lgASA}	$-0.18\log_{10}(1+f_c)+1.78$	1.72
$lgASA = \log_{10}(ASA/1^o)$	σ_{lgASA}	$-0.12\log_{10}(1+f_c)+0.2$	0.3
ZOA spread (ZSA)	μ_{lgZSA}	$-0.2\log_{10}(1+f_c)+1.5$	$-0.13\log_{10}(1+f_c)+1.45$
$lgZSA = \log_{10}(ZSA/1^o)$	σ_{lgZSA}	0.35	0.45
ZOD spread (ZSD)	μ_{lgZSD}	1.35	1.2
$lgZSD = \log_{10}(ZSD/1^o)$	σ_{lgZSA}	0.35	0.55
Cluster Shadow std [dB]	ζ	4	3
K-factor (K)	μ_K	7	N/A
[dB]	σ_K	8	N/A
Cluster ASD (c_{ASD}) in [deg]		5	5
Cluster ASA (c_{ASA}) in [deg]		8	8
Cluster ZSA (c_{ZSA}) in [deg]		9	9

In the next step, we are going to generate channel coefficients for each cluster n and each receiver, transmitter pair u,s. According to 3GPP, channel coefficients for each u,s pair and cluster n is given by:

$$\begin{aligned} H_{u,s,n}(t,\tau) =& \sqrt{\frac{P_n}{M}} \cdot \sum_{m=1}^{M} \hat{F}_{rx}(\theta^A_{n,m},\phi^A_{n,m}) \\ &\times \begin{bmatrix} e^{j\Phi^{\theta,\theta}_{n,m}} & \sqrt{K^{-1}_{n,m}}e^{j\Phi^{\theta,\phi}_{n,m}} \\ \sqrt{K^{-1}_{n,m}}e^{j\Phi^{\phi,\theta}_{n,m}} & e^{j\Phi^{\phi,\phi}_{n,m}} \end{bmatrix} \\ &\times \hat{F}_{tx}(\theta^D_{n,m},\phi^D_{n,m}) \\ &\times e^{j\hat{k}^T_{rx,n,m}\hat{d}_{rx,u}} e^{j\hat{k}^T_{tx,n,m}\hat{d}_{tx,s}} . e^{j2\pi v_{n,m}t} . \delta(\tau-\tau_n) \end{aligned} \tag{C.30}$$

where

- Ray belonging to the same cluster have the same power P_n, τ_n , angles of arrival $\theta^D_{n,m}, \phi^D_{n,m}$, departures $\theta^A_{n,m}, \phi^A_{n,m}$ and uniformly distributed $\Phi_{n,m}$
- Each ray accounts for the antenna field pattern $F(\theta_{n,m}, \phi_{n,m})$
- Power distribution among the vertical and horizontal polarization through the term $K_{n,m}$
- the term $\exp(j\hat{k}^T\hat{d})$ represent the array responses of the transmitting and receiving antennas
- $v_{n,m}$ is a phase shift due to Doppler effect.

For the two strongest cluster, n=1 and n=2, rays are spread in delay to three sub-clusters (per cluster), with fixed delay offset. The delays of the sub-clusters are:

$$\begin{aligned}\tau_{n,1} &= \tau_n \\ \tau_{n,2} &= \tau_n + 1.28c_{DS} \\ \tau_{n,3} &= \tau_n + 2.56c_{DS}\end{aligned} \tag{C.31}$$

where c_{DS} is cluster delay spread specified in Table C.4 [44]. Normally, the N/A value of cluster delay specified in the table is default at $3.91ns$.

- **In the NLOS case, the channel impulse response for NLOS is given by:**

$$H^{NLOS}_{u,s}(\tau,t) = \sum_{n=1}^{2}\sum_{i=1}^{3}\sum_{m\in R_i} H^{NLOS}_{u,s,n,m}\delta(\tau-\tau_{n,i}) + \sum_{n=3}^{N} H^{NLOS}_{u,s,n}\delta(\tau-\tau_n) \tag{C.32}$$

where, $H^{NLOS}_{u,s,n}(t)$ for N-2 weekest links are given by:

$$\begin{aligned}H^{NLOS}_{u,s,n}(t) =& \sqrt{\frac{P_n}{M}} * \sum_{m=1}^{M} \hat{F}_{rx}(\theta^A_{n,m}, \phi^A_{n,m}) \begin{bmatrix} e^{j\Phi^{\theta,\theta}_{n,m}} & \sqrt{K^{-1}_{n,m}}e^{j\Phi^{\theta,\phi}_{n,m}} \\ \sqrt{K^{-1}_{n,m}}e^{j\Phi^{\phi,\theta}_{n,m}} & e^{j\Phi^{\phi,\phi}_{n,m}} \end{bmatrix} \\ & \hat{F}_{tx}(\theta^D_{n,m}, \phi^D_{n,m}) e^{j\hat{k}^T_{rx,n,m}\hat{d}_{rx,u}} e^{j\hat{k}^T_{tx,n,m}\hat{d}_{tx,s}} e^{j2\pi v_{n,m}t}\end{aligned} \tag{C.33}$$

where, $H^{NLOS}_{u,s,n,m}(t)$ for 2 strongest cluster are given by:

$$\begin{aligned}H^{NLOS}_{u,s,n,m}(t) =& \sqrt{\frac{P_n}{M}} \hat{F}_{rx}(\theta^A_{n,m}, \phi^A_{n,m}) \begin{bmatrix} e^{j\Phi^{\theta,\theta}_{n,m}} & \sqrt{K^{-1}_{n,m}}e^{j\Phi^{\theta,\phi}_{n,m}} \\ \sqrt{K^{-1}_{n,m}}e^{j\Phi^{\phi,\theta}_{n,m}} & e^{j\Phi^{\phi,\phi}_{n,m}} \end{bmatrix} \\ & \hat{F}_{tx}(\theta^D_{n,m}, \phi^D_{n,m}) e^{j\hat{k}^T_{rx,n,m}\hat{d}_{rx,u}} e^{j\hat{k}^T_{tx,n,m}\hat{d}_{tx,s}} e^{j2\pi v_{n,m}t}\end{aligned} \tag{C.34}$$

- **In the LOS case, the LOS channel coefficient is given by:**

$$\begin{aligned}H^{LOS}_{u,s,l}(t) =& \begin{bmatrix} F_{rx,u,\theta}(\theta_{LOS,ZOA}, \phi_{LOS,AOA}) \\ F_{rx,u,\phi}(\theta_{LOS,ZOA}, \phi_{LOS,AOA}) \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} F_{tx,s,\theta}(\theta_{LOS,ZOD}, \phi_{LOS,AOD}) \\ F_{tx,s,\phi}(\theta_{LOS,ZOD}, \phi_{LOS,AOD}) \end{bmatrix} \\ & \exp\left(-j2\pi\frac{d_{3D}}{\lambda_0}\right).\exp\left(-j2\pi\frac{\hat{r}^T_{rx,LOS}.\overline{d}_{rx,u}}{\lambda_0}\right).\exp\left(-j2\pi\frac{\hat{r}^T_{tx,LOS}.\overline{d}_{tx,s}}{\lambda_0}\right).\exp\left(-j2\pi\frac{\hat{r}^T_{rx,LOS}.\overline{v}}{\lambda_0}t\right)\end{aligned} \tag{C.35}$$

Then, the channel impulse response is given by adding the LOS channel coefficient to the NLOS channel impulse response and scaling both terms according to the desired K-factor K_R as:

$$H^{LOS}_{u,s}(\tau,t) = \sqrt{\frac{1}{K_R+1}}.H^{NLOS}_{u,s}(\tau,t) + \sqrt{\frac{K_R}{K_R+1}}.H^{LOS}_{u,s,l}(t)\delta(\tau-\tau_1) \tag{C.36}$$

where $H^{NLOS}_{u,s}(\tau,t)$ is given in Equation (C.32).

C.5 Additional component: spatial consistency , blockage effects

In the scenario where the mobility of UE is considered, i.e, UEs move in an area in which channel is uncorrelated and the fading parameters completely change, the spatial consistency will be added according to Procedure A of [44] for both LOS and NLOS communications. The spatial consistency procedures involve the channel update method, in which channel parameters (large scale and small scales) change every period of t_{PER}. In particular, the realisation of the channel matrix C.30 will be deleted, and the new channel coefficient for each antenna pair (u, s) is forced to recompute by repeating from **Step 2** to **Step 11**. It will update the LOS/NLOS probability, propagation path loss according to the current position, cluster delays, powers and departure as well as arrival angles. The updates will take into account velocity and position with respect to these generated at time $t - t_{PER}$

In terms of blockage features that are modelled in our simulation, we support modelling the attenuation of signals that get through the human body or vehicles according to stochastic or geometric methods of Procedure A or B [44]. For more realistic blocking modelling purpose, we adopted Procedure B by first placing K_{bloc} blockers in the scenario. Each blocker $k \in K_{bloc}$ physically has the dimension $(h_k \times w_k)$ at coordinate (x_k, y_k, h_k). In InF scenario, the blockers, which can be automated guided vehicles (AVG) or other industrial robots, cause blocking each sub-path of the propagation according to a simple knife edge diffraction (KED) model [127]. Moreover, the blocking of certain cluster is correlated in space and time according to the UE mobility. If both spatial consistency and blockage are used, the update of channel realization with these featured will be synchronized, i.e the cluster blockage is updated before the channel coefficients are recomputed with the spatial consistency procedure.

C.6 Beamforming (BF) scheme

In this part, we will describe the analog MIMO BF scheme implementation that is used in parallel with the channel model. In this work, we consider an ideal beamforming method which is based on the perfect knowledge acquirement (e.g: cell-scan method) and does not consume any time/frequency overhead for BF vectors. Since analog beamforming is considered, each antenna element applies a phase shift to the transmit signal so that it is concentrated in one direction [128]. Thus, the gain of beamforming is obtained by multiplexing transmitter-receiver beamforming vector with the channel coefficient matrix $H(t, \tau)$. In this work, the cell scanning method is applied to determine the best beam direction as the illustration in Figure C.4. Basically, the coverage area is divided into several sectors in which the directional beam points towards using predefined codebook. It means that, if the transmitter wants to transmit in a sector s, it will select from a list (i.e codebook) the beamforming vector corresponding to that sector and the signal will be pointed in that sector.

For a given transmitter and receiver, in order to determine which is the best pair of sectors (or BF vectors) to direct the beam toward, we relied on the highest SNR assuming the perfect knowledge of the channel.

After calculating the best beamforming vector for each transmitter-receiver pair, the beamforming gain, $G(t, f_s)$ at time t and subcarrier frequency f_s is computed as [129]:

$$G(t, f_s) = \sum_{n=1}^{N} \mathbf{w}_{rx}^{\dagger} H_n \mathbf{w}_{tx} e^{j2\pi \frac{r_{rx,n}^T vt}{\lambda_0}} e^{j2\pi\tau_n f_s} \tag{C.37}$$

where w_{rx}, w_{tx} are the receiver and transmitter side beamforming vector, H_n is the matrix of size $U \times S$ representing the long-term MIMO channel for cluster n (Equation C.30). The first exponential term is the small scale fading effect caused by the central angle of cluster, $r_{rx,n}^T$ is the transpose of the

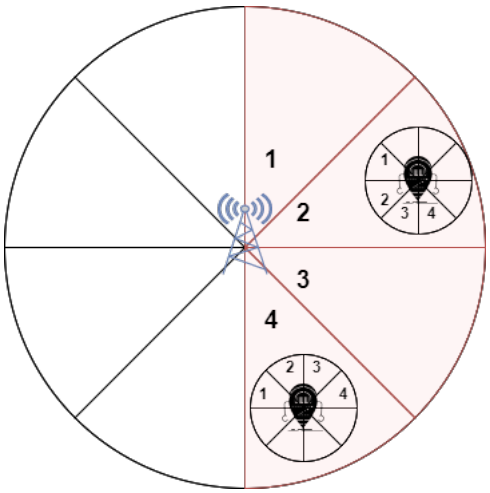

FIGURE C.4: *Sector selection with cell scanning method.*

receiver spherical unit vector, with azimuth angle $\theta_{n,AOA}$ and elevation arrival angle $\theta_{n,ZOA}$, given by:

$$r_{rx,n} = \begin{pmatrix} sin\theta_{n,ZOA}.cos\phi_{n,AOA} \\ sin\theta_{n,ZOA}.sin\phi_{n,AOA} \\ cos\theta_{n,ZOA} \end{pmatrix} \tag{C.38}$$

v is the vector of the relative speed in a 3D space between UE and gNb and λ_0 is the wavelength. The second exponential term represents the frequency selectivity effect caused by the delay τ_n of the n-th cluster.

APPENDIX D

Lyapunov's Optimizations

In this Appendix, we will provide full derivations of the Lyapunov's optimization used for Chapter 3.

D.1 From (in)equality constraints to the virtual queue stabilisation

Suppose that we want to design a control policy that solves the following optimization problems:

$$\begin{aligned} &\text{minimize } \overline{f_0(t)} && && (\mathcal{L}_1)\\ &\text{s.t.} \quad \overline{i_l} \leq 0, && \forall l \in \{1,..,L\} && (1)\\ &\qquad \overline{e_j} = 0, && \forall j \in \{1,..,J\} && (2)\\ &\qquad \text{Queues } Q_k(t) \text{ are mean rate stable,} && \forall k \in \{1,..,K\} && (3) \end{aligned}$$

where: $\overline{f_0(t)}$ is the time average function to minimize, $\overline{i_l}$ and $\overline{e_j}$ are the limiting values of $\overline{i_l(t)}$ and $\overline{e_j}(t)$ which correspond to the inequality and equality constraints, respectively.

The update of actual queues will be done according to the following dynamic equation:

$$Q(t+1) = \max\left[Q(t) - b(t), 0\right] + a(t) \tag{D.1}$$

where the value of a(t),b(t) represents the amount of new packets that arrives, departs on slot t, respectively, and they are assumed to be non-negative.

We can equivalently express the dynamics in Equation D.1 without the appearance of non-linear function $max[.,0]$ as follows:

$$Q(t+1) = Q(t) - \tilde{b}(t) + a(t) \tag{D.2}$$

where $\tilde{b}(t) = \min[b(t), Q(t)] \leq b(t) \forall t$ represents the actual packets proceeded on slot t.

In order to solve the problem $\mathcal{L}_1$, we first transform all inequality and equality constrains into queue stability problem. Particularly, we define the following *virtual queues* $Z_l(t)$ and $H_j(t)$ for each $l \in \{1,..,L\}$ and $k \in \{1,..,K\}$:

$$Z_l(t+1) = \max\left[Z_l(t) + i_l(t), 0\right] \tag{D.3}$$

$$H_j(t+1) = H_j(t) + e_j(t) \tag{D.4}$$

These designed virtual queues are necessary condition to turn the time average (in)equality constraints into a pure queue stability problem.

Proof. Indeed, if $Z_l(t)$ satisfies Equation D.3, we can easily see that

$$\frac{Z_l(t)}{t} - \frac{Z_l(0)}{t} \geq \frac{1}{t}\sum_{\tau=0}^{t-1} i_l(t)$$

By taking the expectations of both sides of above inequality equation and taking $t \to \infty$, we obtain:

$$\limsup_{t\to\infty} \frac{\mathbb{E}\{Z_l(t)\}}{t} \geq \limsup_{t\to\infty} \overline{i_l(t)}$$

Recalling that $\overline{i_l(t)}$ is the time average expectation of $i_l(\tau)$ over $\tau \in \{0, ..., t-1\}$. Therefore, if $Z_l(t)$ is mean rate stable, the left-hand-side of the above inequality is 0, thus:

$$\limsup_{t\to\infty} \overline{i_l(t)} \leq 0$$

This means that the inequality constraints for $i_l(t)$ is already satisfied.

In terms of the virtual queue $H_j(t)$ concerning equality constraints, we can easily obtain the following equation for any $t > 0$:

$$H_j(t) - H_j(0) = \sum_{\tau=0}^{t-1} e_j(\tau)$$

Taking expectations and dividing by t will yield follow equation:

$$\frac{\mathbb{E}\{H_j(t)\} - \mathbb{E}\{H_j(0)\}}{t} = \overline{e_j(t)}$$

Therefore, if $H_j(t)$ is mean rate stable, then the following equation holds and the desired equality constraint for $e_j(t)$ is satisfied.

$$\lim_{t\to\infty} \overline{e_j(t)} = 0$$

D.2 Upper bound of Lyapunov drift-plus-penalty function

Let $\boldsymbol{\Theta(t)} = \{\boldsymbol{Q(t)}, \boldsymbol{Z(t)}, \boldsymbol{H(t)}\}$ be concatenated vector of all actual queues and virtual queues. Then, the Lyapunov function is defined as:

$$L(\boldsymbol{\Theta(t)}) = \frac{1}{2}\sum_{k=1}^{K} Q_k(t)^2 + \frac{1}{2}\sum_{l=1}^{L} Z_l(t)^2 + \frac{1}{2}\sum_{j=1}^{L} H_j(t)^2 \tag{D.5}$$

According to Neely et al. [70], under any control algorithm, the upper bound of the drift-plus-penalty function can be expressed as following, for any possible values of $\boldsymbol{\Theta(t)}$ and all parameter $v \geq 0$:

$$\begin{aligned}\Delta(\boldsymbol{\Theta(t)}) + v.\mathbb{E}\{f_0(t)|\boldsymbol{\Theta(t)}\} \leq B + v.\mathbb{E}\{f_0(t)|\boldsymbol{\Theta(t)}\} + \sum_{k=1}^{K} Q_k(t)\mathbb{E}\{a_k(t) - b_k(t)|\boldsymbol{\Theta(t)}\} \\ \sum_{l=1}^{L} Z_l(t)\mathbb{E}\{i_l(t)|\boldsymbol{\Theta(t)}\} + \sum_{j=1}^{L} H_j(t)\mathbb{E}\{e_j(t)|\boldsymbol{\Theta(t)}\}\end{aligned} \tag{D.6}$$

where B is a positive constant that satisfies the following for all t:

$$\begin{aligned} B \geq & \frac{1}{2}\sum_{k=1}^{K}\mathbb{E}\{a_k(t)^2 - b_k(t)^2|\boldsymbol{\Theta(t)}\} + \frac{1}{2}\sum_{l=1}^{L}\mathbb{E}\{i_l(t)^2|\boldsymbol{\Theta(t)}\} \\ & + \frac{1}{2}\sum_{j=1}^{L}\mathbb{E}\{e_j(t)^2|\boldsymbol{\Theta(t)}\} - \sum_{k=1}^{K}\mathbb{E}\{a_k(t).\tilde{b}_k(t)|\boldsymbol{\Theta(t)}\} \end{aligned} \tag{D.7}$$

Proof. By squaring Equation D.1 stressing the fact that $max[p,0]^2 \leq p^2$, we will get the following expression:

$$Q(t+1)^2 \leq (Q_k(t) - b_k(t))^2 + a_k(t)^2 + 2\max[Q_k(t) - b_k(t), 0]a_k(t) \tag{D.8}$$

$$= (Q_k(t) - b_k(t))^2 + a_k(t)^2 + 2\left(Q_k(t) - \tilde{b}_k(t)\right)a_k(t) \tag{D.9}$$

Therefore:

$$\frac{Q_k(t+1)^2 - Q_k(t)^2}{2} \leq \frac{a_k(t)^2 - b_k(t)^2}{2} - \tilde{b}_k(t)a_k(t) + Q_k(t)[a_k(t) - b_k(t)] \tag{D.10}$$

Similarly, we will apply the same proofs for $Z_l(t)$ and $H_j(t)$ and get the following expression:

$$\begin{aligned} \frac{Z_l(t+1)^2 - Z_l(t)^2}{2} &\leq \frac{i_l(t)^2}{2} + Z_l(t)i_l(t) \\ \frac{H_j(t+1)^2 - H_j(t)^2}{2} &\leq \frac{e_j(t)^2}{2} + H_j(t)e_j(t) \end{aligned} \tag{D.11}$$

Then, we will take the conditional expectations of the above three equations and summing over $k \in \{1,...,K\}$, $l \in \{1,...,L\}$ and $j \in \{1,...,J\}$, we will get the following results:

$$\begin{aligned} & \frac{1}{2}\sum_{k=1}^{K}\mathbb{E}\left\{\left(Q_k^2(t+1) - Q_k^2(t)\right)|\boldsymbol{\Theta(t)}\right\} + \frac{1}{2}\sum_{l=1}^{L}\mathbb{E}\left\{\left(Z_l^2(t+1) - Z_l^2(t)\right)|\boldsymbol{\Theta(t)}\right\} \\ & + \frac{1}{2}\sum_{j=1}^{J}\mathbb{E}\left\{\left(H_j^2(t+1) - H_j^2(t)\right)|\boldsymbol{\Theta(t)}\right\} \\ & \leq \sum_{k=1}^{K}\mathbb{E}\left\{\left(a_k^2(t) - b_k^2(t)\right)|\boldsymbol{\Theta(t)}\right\} - \sum_{k=1}^{K}\mathbb{E}\{\tilde{b}_k(t)a_k(t)|\boldsymbol{\Theta(t)}\} + \sum_{k=1}^{K}\mathbb{E}\{Q_k(t)[a_k(t) - b_k(t)]|\boldsymbol{\Theta(t)}\} \\ & + \frac{1}{2}\sum_{l=1}^{L}\mathbb{E}\{i_l(t)^2|\boldsymbol{\Theta(t)}\} + \sum_{l=1}^{L}\mathbb{E}\{Z_l(t)i_l(t)|\boldsymbol{\Theta(t)}\} \\ & + \frac{1}{2}\sum_{j=1}^{J}\mathbb{E}\{e_j(t)^2|\boldsymbol{\Theta(t)}\} + \sum_{j=1}^{J}\mathbb{E}\{H_j(t)e_j(t)|\boldsymbol{\Theta(t)}\} \end{aligned} \tag{D.12}$$

We see that the right hand side of Equation D.12 follows the definition of *one-slot conditional Lyapunov drift function* $\Delta(\boldsymbol{\Theta(t)})$ By choosing a positive constant B value that satisfies Equation D.7, we will transform Equation D.12 as follows:

$$\Delta(\boldsymbol{\Theta(t)}) \leq B + \sum_{k=1}^{K}Q_k(t)\mathbb{E}\{a_k(t) - b_k(t)|\boldsymbol{\Theta(t)}\} + \sum_{l=1}^{L}Z_l(t)\mathbb{E}\{i_l(t)|\boldsymbol{\Theta(t)}\} + \sum_{j=1}^{J}H_j(t)\mathbb{E}\{e_j(t)|\boldsymbol{\Theta(t)}\} \tag{D.13}$$

Afterwards, by adding $v.\mathbb{E}\{f_0(t)|\boldsymbol{\Theta(t)}\}$ in both sides of equation above, we will get the upper bound of the Lyapunov drift-plus-penalty-function as Equation D.6

Remarks D.2.1

Thus, instead of directly minimising the Lyapunov-drift-plus-penalty function $\Delta(\boldsymbol{\Theta(t)}) + v.\mathbb{E}\{f_0(t)|\boldsymbol{\Theta(t)}\}$ *on every slot t, we are now trying to minimize the bound given in the right-hand-size of Equation D.6.*

The outcomes of upper bound minimization have been realised in Chapter 3, Section 3.3. In adaptive resource allocation algorithm, the equality constraint is not considered, thus J=0. In adaptive, reliability-aware resource allocation algorithm, both equality constraint and inequality constraint are not used, so virtual queues are not included in the optimization problem ($J = 0$ and $L = 0$).

Bibliography

[1] G. J. Sutton et al. “Enabling Technologies for Ultra-Reliable and Low Latency Communications: From PHY and MAC Layer Perspectives”. In: *IEEE Communications Surveys Tutorials* 21.3 (2019), pp. 2488–2524.

[2] M. Bennis, M. Debbah, and H. V. Poor. “Ultrareliable and Low-Latency Wireless Communication: Tail, Risk, and Scale”. In: *Proceedings of the IEEE* 106.10 (2018), pp. 1834–1853.

[3] N. Patriciello, S. Lagen, B. Bojovic, and L. Giupponi. “An E2E simulator for 5G NR networks”. en. In: *Simulation Modelling Practice and Theory* 96 (Nov. 2019), p. 101933. URL: https://www.sciencedirect.com/science/article/pii/S1569190X19300589 (visited on 05/16/2021).

[4] H. Ji et al. “Ultra-Reliable and Low-Latency Communications in 5G Downlink: Physical Layer Aspects”. In: *IEEE Wireless Communications* 25.3 (June 2018), pp. 124–130. eprint: 1704.05565 (cs, math).

[5] L. Chettri and R. Bera. “A Comprehensive Survey on Internet of Things (IoT) Toward 5G Wireless Systems”. In: *IEEE Internet of Things Journal* 7.1 (Jan. 2020), pp. 16–32.

[6] S. Painuly, P. Kohli, P. Matta, and S. Sharma. “Advance Applications and Future Challenges of 5G IoT”. In: *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*. Dec. 2020, pp. 1381–1384.

[7] *M.2083 : IMT Vision - Framework and Overall Objectives of the Future Development of IMT for 2020 and Beyond*. https://www.itu.int/rec/R-REC-M.2083-0-201509-I.

[8] 3GPP. *Study on Scenarios and Requirements for Next Generation Access Technologies, Release 17 (v17.0.0)*. 2022.

[9] O. O. Erunkulu et al. “5G Mobile Communication Applications: A Survey and Comparison of Use Cases”. In: *IEEE Access* 9 (2021), pp. 97251–97295.

[10] G. Hampel, C. Li, and J. Li. “5G Ultra-Reliable Low-Latency Communications in Factory Automation Leveraging Licensed and Unlicensed Bands”. In: *IEEE Communications Magazine* 57.5 (May 2019), pp. 117–123.

[11] M. Kim, S. Kim, and Y. Lim. “An Implementation of Downlink Asynchronous HARQ for LTE TDD System”. In: *2012 IEEE Radio and Wireless Symposium*. 2012 IEEE Radio and Wireless Symposium. 2012, pp. 271–274.

[12] M. Torres Vega et al. “Immersive Interconnected Virtual and Augmented Reality: A 5G and IoT Perspective”. In: *Journal of Network and Systems Management* 28.4 (Oct. 2020), pp. 796–826.

[13] P. Hu and J. Zhang. “5G-Enabled Fault Detection and Diagnostics: How Do We Achieve Efficiency?” In: *IEEE Internet of Things Journal* 7.4 (Apr. 2020), pp. 3267–3281.

[14] M. Maksimović et al. “On the Role of 5G Ultra-Reliable Low-Latency Communications (URLLC) in Applications Extending Smart Grid (SG) Capabilities”. In: *2022 11th Mediterranean Conference on Embedded Computing (MECO)*. June 2022, pp. 1–4.

[15] 3GPP. *Service requirements for the 5G system, Release 16 (v16.14.0)*. 2021.

[16] P. Popovski et al. "Wireless Access in Ultra-Reliable Low-Latency Communication (URLLC)". In: *IEEE Transactions on Communications* 67.8 (Aug. 2019), pp. 5783–5801.

[17] M. Shirvanimoghaddam et al. "Short Block-Length Codes for Ultra-Reliable Low Latency Communications". In: *IEEE Communications Magazine* 57.2 (Feb. 2019), pp. 130–137.

[18] C. Sun, C. She, and C. Yang. "Energy-Efficient Resource Allocation for Ultra-Reliable and Low-Latency Communications". In: *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*. Dec. 2017, pp. 1–6.

[19] J. Park and P. Popovski. "Coverage and Rate of Downlink Sequence Transmissions With Reliability Guarantees". In: *IEEE Wireless Communications Letters* 6.6 (Dec. 2017), pp. 722–725.

[20] G. Durisi, T. Koch, and P. Popovski. "Toward Massive, Ultrareliable, and Low-Latency Wireless Communication With Short Packets". In: *Proceedings of the IEEE* 104.9 (Sept. 2016), pp. 1711–1726.

[21] P. Guan et al. *Ultra-Low Latency for 5G - A Lab Trial*. Oct. 2016. arXiv: 1610.04362 [cs].

[22] A. Anand, G. De Veciana, and S. Shakkottai. "Joint Scheduling of URLLC and eMBB Traffic in 5G Wireless Networks". In: *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*. Apr. 2018, pp. 1970–1978.

[23] N. H. Mahmood et al. "Uplink Grant-Free Access Solutions for URLLC Services in 5G New Radio". In: *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*. Aug. 2019, pp. 607–612.

[24] M. Amjad and L. Musavian. "Performance Analysis of NOMA for Ultra-Reliable and Low-Latency Communications". In: *2018 IEEE Globecom Workshops (GC Wkshps)*. Dec. 2018, pp. 1–5.

[25] J. Khan and L. Jacob. "Multi-Connectivity for URLLC: Performance Comparison of Different Architectures". In: *2020 IEEE Region 10 Symposium (TENSYMP)*. June 2020, pp. 892–895.

[26] T. K. Vu et al. "Ultra-Reliable and Low Latency Communication in mmWave-Enabled Massive MIMO Networks". In: *IEEE Communications Letters* 21.9 (Sept. 2017), pp. 2041–2044.

[27] H. Shariatmadari et al. "Optimized Transmission and Resource Allocation Strategies for Ultra-Reliable Communications". In: *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. Sept. 2016, pp. 1–6.

[28] P. Marsch et al. "Emerging Network Architecture and Functional Design Considerations for 5G Radio Access". In: *Transactions on Emerging Telecommunications Technologies* 27.9 (2016), pp. 1168–1177.

[29] A. E. Kalør et al. *Network Slicing for Ultra-Reliable Low Latency Communication in Industry 4.0 Scenarios*. Aug. 2017. eprint: 1708.09132 (cs, math).

[30] E. Bastug, M. Bennis, M. Medard, and M. Debbah. "Toward Interconnected Virtual Reality: Opportunities, Challenges, and Enablers". In: *IEEE Communications Magazine* 55.6 (June 2017), pp. 110–117.

[31] R. Ford et al. *Provisioning Low Latency, Resilient Mobile Edge Clouds for 5G*. Mar. 2017.

[32] F. Boccardi et al. "Five Disruptive Technology Directions for 5G". In: *IEEE Communications Magazine* 52.2 (Feb. 2014), pp. 74–80.

[33] M. Chen et al. "Mobility-Aware Caching and Computation Offloading in 5G Ultra-Dense Cellular Networks". In: *Sensors* 16.7 (July 2016), p. 974.

[34] M. Gregori, J. Gómez-Vilardebó, J. Matamoros, and D. Gündüz. "Wireless Content Caching for Small Cell and D2D Networks". In: *IEEE Journal on Selected Areas in Communications* 34.5 (May 2016), pp. 1222–1234.

[35] A. Sengupta, R. Tandon, and O. Simeone. "Cache Aided Wireless Networks: Tradeoffs between Storage and Latency". In: *2016 Annual Conference on Information Science and Systems (CISS)*. Mar. 2016, pp. 320–325.

[36] F. Xu, K. Liu, and M. Tao. "Cooperative Tx/Rx Caching in Interference Channels: A Storage-Latency Tradeoff Study". In: *2016 IEEE International Symposium on Information Theory (ISIT)*. July 2016, pp. 2034–2038.

[37] S. M. Azimi, O. Simeone, and R. Tandon. "Fundamental Limits on Latency in Small-Cell Caching Systems: An Information-Theoretic Analysis". In: *2016 IEEE Global Communications Conference (GLOBECOM)*. Dec. 2016, pp. 1–6.

[38] J. Liu and Q. Zhang. "Offloading Schemes in Mobile Edge Computing for Ultra-Reliable Low Latency Communications". In: *IEEE Access* 6 (2018), pp. 12825–12837.

[39] P. K. Agyapong et al. "Design Considerations for a 5G Network Architecture". In: *IEEE Communications Magazine* 52.11 (Nov. 2014), pp. 65–75.

[40] I. Parvez et al. "A Survey on Low Latency Towards 5G: RAN, Core Network and Caching Solutions". In: *IEEE Communications Surveys & Tutorials* 20.4 (2018), pp. 3098–3130.

[41] C. Liang, F. R. Yu, and X. Zhang. "Information-Centric Network Function Virtualization over 5g Mobile Wireless Networks". In: *IEEE Network* 29.3 (May 2015), pp. 68–74.

[42] C. C. Marquezan et al. "Understanding Processing Latency of SDN Based Mobility Management in Mobile Core Networks". In: *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. Sept. 2016, pp. 1–7.

[43] R. Casellas, R. Muñoz, R. Vilalta, and R. Martínez. "Orchestration of IT/Cloud and Networks: From Inter-DC Interconnection to SDN/NFV 5G Services". In: *2016 International Conference on Optical Network Design and Modeling (ONDM)*. May 2016, pp. 1–6.

[44] 3GPP. *Study on channel model for frequencies from 0.5 to 100 GHz, Release 16 (v16.1.0)*. 2019.

[45] J. Baranda et al. "Orchestration of End-to-End Network Services in the 5G-Crosshaul Multi-Domain Multi-Technology Transport Network". In: *IEEE Communications Magazine* 56.7 (July 2018), pp. 184–191.

[46] K. Katsalis, N. Nikaein, and A. Edmonds. "Multi-Domain Orchestration for NFV: Challenges and Research Directions". In: *2016 15th International Conference on Ubiquitous Computing and Communications and 2016 International Symposium on Cyberspace and Security (IUCC-CSS)*. IEEE, Dec. 2016.

[47] P. Rost et al. "Mobile network architecture evolution toward 5G". In: *IEEE Communications Magazine* 54.5 (May 2016), pp. 84–91.

[48] I. F. Akyildiz, S. Nie, S.-C. Lin, and M. Chandrasekaran. "5G roadmap: 10 key enabling technologies". In: *Computer Networks* (Sept. 2016).

[49] G. Liva, L. Gaudio, T. Ninacs, and T. Jerkovits. "Code Design for Short Blocks: A Survey". In: *11th International ITG Conference on Systems, Communications and Coding* (2016). arXiv: 1610.00873v1 [cs.IT].

[50] H. Chen et al. "Ultra-Reliable Low Latency Cellular Networks: Use Cases, Challenges and Approaches". In: *IEEE Communications Magazine* 56.12 (Dec. 2018), pp. 119–125.

[51] G. Pocovi et al. "Signal Quality Outage Analysis for Ultra-Reliable Communications in Cellular Networks". In: *2015 IEEE Globecom Workshops (GC Wkshps)*. IEEE, Dec. 2015.

[52] C.-P. Li et al. "5G ultra-reliable and low-latency systems design". In: *2017 European Conference on Networks and Communications (EuCNC)*. IEEE, June 2017.

[53] J. Qadir et al. “Exploiting the Power of Multiplicity: A Holistic Survey of Network-Layer Multipath”. In: *IEEE Communications Surveys & Tutorials* 17.4 (2015), pp. 2176–2213.

[54] J. J. Nielsen, R. Liu, and P. Popovski. “Ultra-Reliable Low Latency Communication Using Interface Diversity”. In: *IEEE Transactions on Communications* 66.3 (Mar. 2018), pp. 1322–1334.

[55] K. I. Pedersen, G. Pocovi, J. Steiner, and S. R. Khosravirad. “Punctured Scheduling for Critical Low Latency Data on a Shared Channel with Mobile Broadband”. In: *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*. IEEE, Sept. 2017.

[56] 3GPP. *Study on latency reduction techniques for LTE (release 14), v14.0.0*. 2016.

[57] D. Sabella. *Multi-Access Edge Computing: Software Development at the Network Edge*. Textbooks in Telecommunication Engineering. Cham: Springer International Publishing, 2021. (Visited on 03/23/2023).

[58] N. Patriciello, S. Lagen, L. Giupponi, and B. Bojovic. “An Improved MAC Layer for the 5G NR Ns-3 Module”. In: *Proceedings of the 2019 Workshop on Ns-3*. Florence Italy: ACM, June 2019, pp. 41–48.

[59] 5G PPP Architecture Working Group. *View on 5G Architecture, Version 3.0*. Tech. rep. 5GPP, 2019.

[60] W. Jiang, S. D. Anton, and H. D. Schotten. “Intelligence Slicing: A Unified Framework to Integrate Artificial Intelligence into 5G Networks”. In: *2019 12th IFIP Wireless and Mobile Networking Conference (WMNC)*. IEEE, Sept. 2019.

[61] M. Maman, L. N. Dinh, and E. Calvanese Strinati. *Procede et dispositif d’orchestration de l’execution de mecanismes dans un reseau sans fil, FR Patent 2103542*. U.S. pat. 2021.

[62] T.-K. Le, U. Salim, and F. Kaltenberger. “An Overview of Physical Layer Design for Ultra-Reliable Low-Latency Communications in 3GPP Releases 15, 16, and 17”. In: *IEEE Access* 9 (2021), pp. 433–444.

[63] Y. Liu et al. “Analyzing Grant-Free Access for URLLC Service”. In: *IEEE Journal on Selected Areas in Communications* 39.3 (2021), pp. 741–755.

[64] T. Jacobsen et al. “System Level Analysis of Uplink Grant-Free Transmission for URLLC”. In: *2017 IEEE Globecom Workshops (GC Wkshps)*. 2017 IEEE Globecom Workshops (GC Wkshps). 2017, pp. 1–6.

[65] T.-K. Le, U. Salim, and F. Kaltenberger. “Optimal Reserved Resources to Ensure the Repetitions in Ultra-Reliable Low-Latency Communication Uplink Grant-free Transmission”. In: *2019 European Conference on Networks and Communications (EuCNC)*. 2019 European Conference on Networks and Communications (EuCNC). 2019, pp. 554–558.

[66] S. Pfletschinger, D. Declercq, and M. Navarro. “Adaptive HARQ With Non-Binary Repetition Coding”. In: *IEEE Transactions on Wireless Communications* 13.8 (2014), pp. 4193–4204.

[67] M. Jabi, M. Benjillali, L. Szczecinski, and F. Labeau. “Energy Efficiency of Adaptive HARQ”. In: *IEEE Transactions on Communications* 64.2 (2016), pp. 818–831.

[68] H. Mukhtar, A. Al-Dweik, and M. Al-Mualla. “Content-Aware and Occupancy-Based Hybrid ARQ for Video Transmission”. In: *2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*. 2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS). 2016, pp. 1–4.

[69] B. Han et al. “CLARQ: A Dynamic ARQ Solution for Ultra-High Closed-Loop Reliability”. In: *IEEE Transactions on Wireless Communications* 21.1 (Jan. 2022), pp. 280–294.

[70] M. J. Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Vol. 3. Jan. 2010.

[71] S. Lagen et al. “New Radio Physical Layer Abstraction for System-Level Simulations of 5G Networks”. In: *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*. June 2020, pp. 1–7.

[72] 3GPP. *Multiplexing and channel coding, Release 16 (v16.2.0)*. 2020.

[73] P. Popovski. “Ultra-Reliable Communication in 5G Wireless Systems”. In: *1st International Conference on 5G for Ubiquitous Connectivity*. Nov. 2014, pp. 146–151.

[74] 3GPP. *Medium Access Control (MAC) Protocol Specification, Release 16 (v16.2.0)*. 2019.

[75] ITU. *The prediction of the time and the spatial profile for broadband and mobile services using UHF and SHF bands*. en. Aug. 2019.

[76] M. Marvi, A. Aijaz, and M. Khurram. “On the Use of ON/OFF Traffic Models for Spatio-Temporal Analysis of Wireless Networks”. In: *IEEE Communications Letters* 23.7 (July 2019), pp. 1219–1222.

[77] L. N. Dinh, I. Labriji, M. Maman, and E. Calvanese Strinati. “Toward URLLC with Proactive HARQ Adaptation”. In: *To appear in EUCNC2022* ().

[78] N. Nikaein et al. “OpenAirInterface: A flexible platform for 5G research”. In: *ACM SIGCOMM Computer Communication Review* (2014), pp. 33–38.

[79] L. N. Dinh, R. Bertolini, and M. Maman. “Dynamic Resource Scheduling Optimization for Ultra-Reliable Low Latency Communications: From Simulation to Experimentation”. In: *2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. Sept. 2022, pp. 1026–1031.

[80] F. Kaltenberger, G. de Souza, R. Knopp, and H. Wang. “The OpenAirInterface 5G New Radio Implementation: Current Status and Roadmap”. In: *WSA 2019; 23rd International ITG Workshop on Smart Antennas*. Apr. 2019, pp. 1–5.

[81] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation (Release 15), v15.2.0*. 2018.

[82] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding (Release 15), v15.2.1*. 2018.

[83] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures (Release 17), v17.2.0*. 2022.

[84] 3GPP. *Packet Data Convergence Protocol (PDCP) specification, Release 15 (v15.0.0)*. 2018.

[85] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Link Control (RLC) protocol specification (Release 15), v15.1.0*. 2018.

[86] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification, Release 15 (v15.3.0)*. 2018.

[87] Z. Zhou, R. Ratasuk, N. Mangalvedhe, and A. Ghosh. “Resource Allocation for Uplink Grant-Free Ultra-Reliable and Low Latency Communications”. In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. June 2018, pp. 1–5.

[88] M. W. Nomeir, Y. Gadallah, and K. G. Seddik. “Uplink Scheduling for Mixed Grant-Based eMBB and Grant-Free URLLC Traffic in 5G Networks”. In: *2021 17th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. Oct. 2021, pp. 187–192.

[89] Q. Huang, X. Xie, and M. Cheriet. “Reinforcement Learning-Based Hybrid Spectrum Resource Allocation Scheme for the High Load of URLLC Services”. In: *EURASIP Journal on Wireless Communications and Networking* 2020.1 (Dec. 2020), p. 250.

[90] L. Liang, H. Ye, and G. Y. Li. "Spectrum Sharing in Vehicular Networks Based on Multi-Agent Reinforcement Learning". In: *IEEE Journal on Selected Areas in Communications* 37.10 (Oct. 2019), pp. 2282–2292.

[91] R. Mennes, F. A. P. De Figueiredo, and S. Latré. "Multi-Agent Deep Learning for Multi-Channel Access in Slotted Wireless Networks". In: *IEEE Access* 8 (2020), pp. 95032–95045.

[92] O. Naparstek and K. Cohen. "Deep Multi-User Reinforcement Learning for Distributed Dynamic Spectrum Access". In: *IEEE Transactions on Wireless Communications* 18.1 (Jan. 2019), pp. 310–323.

[93] H. Yang et al. "Deep Reinforcement Learning Based Massive Access Management for Ultra-Reliable Low-Latency Communications". In: *arXiv:2002.08743* (2020). eprint: 2002.08743 (eess).

[94] A. Azari, M. Ozger, and C. Cavdar. *Risk-Aware Resource Allocation for URLLC: Challenges and Strategies with Machine Learning*. 2018. eprint: 1901.04292 (cs, stat).

[95] E. A. Feinberg, P. O. Kasyanov, and N. V. Zadoianchuk. "Average-Cost Markov Decision Processes with Weakly Continuous Transition Probabilities". In: (Feb. 2012).

[96] V. Mnih et al. *Playing Atari with Deep Reinforcement Learning*. Dec. 2013. eprint: 1312.5602 (cs).

[97] V. Mnih et al. "Human-Level Control through Deep Reinforcement Learning". In: *Nature* 518.7540 (Feb. 2015), pp. 529–533.

[98] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. Second edition. Adaptive Computation and Machine Learning Series. Cambridge, Massachusetts: The MIT Press, 2018.

[99] D. Silver et al. "Deterministic Policy Gradient Algorithms". In: *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*. ICML'14. Beijing, China: JMLR.org, 2014, I–387–I–395.

[100] M. L. Littman. "Markov Games as a Framework for Multi-Agent Reinforcement Learning". In: *Proceedings of the Eleventh International Conference on International Conference on Machine Learning*. ICML'94. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., July 1994, pp. 157–163.

[101] R. Lowe et al. "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., Dec. 2017, pp. 6382–6393.

[102] A. Wong, T. Bäck, A. V. Kononova, and A. Plaat. *Deep Multiagent Reinforcement Learning: Challenges and Directions*. Oct. 2022. eprint: 2106.15691 (cs).

[103] D. S. Bernstein, S. Zilberstein, and N. Immerman. *The Complexity of Decentralized Control of Markov Decision Processes*. Jan. 2013. eprint: 1301.3836 (cs).

[104] J. K. Gupta, M. Egorov, and M. Kochenderfer. "Cooperative Multi-agent Control Using Deep Reinforcement Learning". In: *Autonomous Agents and Multiagent Systems*. Ed. by G. Sukthankar and J. A. Rodriguez-Aguilar. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, pp. 66–83.

[105] N. Yang, H. Zhang, and R. Berry. "Partially Observable Multi-Agent Deep Reinforcement Learning for Cognitive Resource Management". In: *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*. Dec. 2020, pp. 1–6.

[106] 3GPP. *NR and NG-RAN Overall description, Release 16 (v16.4.0)*. 2021.

[107] 3GPP. *Universal Mobile Telecommunications System (UMTS), Radio Resource Control (RRC), Protocol specification, Release 14 (v14.2.0)*. 2017.

[108] R. Shreevastav and R. S. Carbajo. "Dynamic RLC Mode Based upon Link Adaptation to Reduce Latency and Improve Throughput in Cellular Networks". In: *2016 IEEE 7th Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)*. 2016 IEEE 7th Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON). Oct. 2016, pp. 1–6.

[109] M. Mezzavilla et al. "End-to-End Simulation of 5G mmWave Networks". In: *IEEE Communications Surveys & Tutorials* 20.3 (2018), pp. 2237–2263. URL: https://ieeexplore.ieee.org/document/8344116/ (visited on 2021).

[110] 3GPP. *Physical layer procedures for data, Release 16 (v16.2.0)*. 2020.

[111] 3GPP. *Radio Resource Control (RRC); Protocol specification, Release 16 (v16.1.0)*. 2020.

[112] 3GPP. *Physical channels and modulation, Release 15 (v15.2.0)*. 2018.

[113] S. Ahmadi. *LTE-advanced: A Practical Systems Approach to Understanding the 3GPP LTE Releases 10 and 11 Radio Access Technologies*. Amsterdam: Elsevier, 2014.

[114] E. Chu, J. Yoon, and B. C. Jung. "A Novel Link-to-System Mapping Technique Based on Machine Learning for 5G/IoT Wireless Networks". In: *Sensors* 19.5 (Jan. 2019), p. 1196.

[115] J. Olmos, S. Ruiz, M. García-Lozano, and D. Martín-Sacristán. "Link Abstraction Models Based on Mutual Information for LTE Downlink". In: (), p. 18.

[116] E. Tuomaala and H. Wang. "Effective SINR Approach of Link to System Mapping in OFDM/Multi-Carrier Mobile Network". In: *2005 2nd Asia Pacific Conference on Mobile Technology, Applications and Systems*. Nov. 2005, 5 pp.–5.

[117] M. Ben Hcine and R. Bouallegue. "Analysis of Uplink Effective SINR in LTE Networks". In: *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*. Aug. 2015, pp. 425–430.

[118] H. Chernoff. "A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on the Sum of Observations". In: *Ann. Math. Statist.* 23.4 (1952), pp. 493–507.

[119] IEEE802.16m-08/004r2. "IEEE 802.16 Broadband Wireless Access Working Group". In: *IEEE 802.16m Evaluation Methodology Document (EMD)* (July 2018), p. 162.

[120] N. Patriciello, S. Lagen, B. Bojovic, and L. Giupponi. "An E2E Simulator for 5G NR Networks". In: *Simulation Modelling Practice and Theory* 96 (Nov. 2019), p. 101933.

[121] J. G. Andrews et al. "Modeling and Analyzing Millimeter Wave Cellular Systems". In: *IEEE Transactions on Communications* 65.1 (Jan. 2017), pp. 403–430.

[122] K. Börner, J. Dommel, S. Jaeckel, and L. Thiele. "On the Requirements for Quasi-Deterministic Radio Channel Models for Heterogeneous Networks". In: *2012 International Symposium on Signals, Systems, and Electronics (ISSSE)*. 2012 International Symposium on Signals, Systems, and Electronics (ISSSE). Oct. 2012, pp. 1–5.

[123] F. Ademaj, M. Taranetz, and M. Rupp. "3GPP 3D MIMO Channel Model: A Holistic Implementation Guideline for Open Source Simulation Tools". In: *EURASIP Journal on Wireless Communications and Networking* 2016.1 (Feb. 19, 2016), p. 55. URL: https://doi.org/10.1186/s13638-016-0549-9 (visited on 05/05/2022).

[124] P. Stenumgaard, J. Chilo, J. Ferrer-Coll, and P. Angskog. "Challenges and Conditions for Wireless Machine-to-Machine Communications in Industrial Environments". In: *IEEE Communications Magazine* 51.6 (June 2013), pp. 187–192.

[125] D. Hampicke et al. “Characterization of the Directional Mobile Radio Channel in Industrial Scenarios, Based on Wideband Propagation Measurements”. In: *Gateway to 21st Century Communications Village. VTC 1999-Fall. IEEE VTS 50th Vehicular Technology Conference (Cat. No.99CH36324)*. Gateway to 21st Century Communications Village. VTC 1999-Fall. IEEE VTS 50th Vehicular Technology Conference (Cat. No.99CH36324). Vol. 4. Sept. 1999, 2258–2262 vol.4.

[126] D. Solomitckii et al. “Characterization of mmWave Channel Properties at 28 and 60 GHz in Factory Automation Deployments”. In: *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. 2018 IEEE Wireless Communications and Networking Conference (WCNC). Apr. 2018, pp. 1–6.

[127] S. Deng, G. R. Maccartney, and T. S. Rappaport. “Indoor and Outdoor 5G Diffraction Measurements and Models at 10, 20, and 26 GHz: 59th IEEE Global Communications Conference, GLOBECOM 2016”. In: *2016 IEEE Global Communications Conference, GLOBECOM 2016 - Proceedings*. 2016 IEEE Global Communications Conference, GLOBECOM 2016 - Proceedings (2016). URL: http://www.scopus.com/inward/record.url?scp=85015442988&partnerID=8YFLogxK (visited on 05/13/2022).

[128] Y. Zou, Q. Li, G. Yang, and X. Cheng. “Analog Beamforming for Millimeter-Wave MIMO Systems via Stochastic Optimization”. In: *2016 8th International Conference on Wireless Communications Signal Processing (WCSP)*. 2016 8th International Conference on Wireless Communications Signal Processing (WCSP). Oct. 2016, pp. 1–5.

[129] M. Zhang et al. “Ns-3 Implementation of the 3GPP MIMO Channel Model for Frequency Spectrum above 6 GHz”. In: *ACM International Conference Proceeding Series* Part F128360 (Feb. 2017), pp. 71–78. URL: http://arxiv.org/abs/1702.04822.