UNIVERSITÉ DU
LUXEMBOURG

PhD-FSTM-2024-105
The Faculty of Sciences, Technology and Medicine

DISSERTATION

Defence held on 18 November 2024 in Luxembourg

to obtain the degree of

DOCTEUR DE L'UNIVERSITÉ DU LUXEMBOURG
EN INFORMATIQUE

by

Fatemeh Kavehmadavani

Born on 23 March 1992 in Darab (Iran)

Artificial Intelligence (AI)-enabled
Smart Radio Environments for 6G Wireless Networks

Dissertation defense committee

Dr. Symeon Chatzinotas, dissertation supervisor
Professor, Université du Luxembourg

Dr. Thang Xuan VU, Vice Chair
Research Scientist, Université du Luxembourg

Dr. Bhavani Shankar, Chair
Assistant Professor, Université du Luxembourg

Dr. Florian Kaltenberger
Associate Professor, Eurecom

Dr. Adrian Kliks
Professor, Poznan University of Technology

“Information is the resolution of uncertainty.”

Claude Shannon
Mathematician

To my love, Saman, and my family, those who have journeyed with me, offering unwavering support, love, and encouragement.
And to all who seek knowledge and strive for understanding, may this work contribute a small part to the collective quest for truth.

Abstract

The transition from fifth-generation (5G) to emerging sixth-generation (6G) wireless networks marks a significant evolution in communication technologies, aiming to support a wide range of services with enhanced performance requirements. Key use cases such as enhanced mobile broadband (eMBB) and ultra-reliable and low-latency communications (uRLLC) demand robust and flexible network architectures. The Open radio access network (O-RAN) Alliance introduces a transformative approach by advocating for disaggregated RAN functionality using open interface specifications. Within this framework, traffic steering (TS) and user association (UA) become crucial use cases, enabling the efficient management of network resources and the optimization of service delivery. Given the above context, this dissertation focuses on designing and developing intelligent TS and UA frameworks for multi-traffic scenarios, thanks to innovative 5G technologies and standardization, to enhance network performance and adaptability in dynamic environments via merging machine learning (ML) models. In particular, the study delves into four key aspects: 1) TS scheme to efficiently allocate heterogeneous network resources in the presence of known dynamic traffic demand and fixed numerology; 2) Long-short-term memory (LSTM)-based traffic prediction strategy for downlink (DL) eMBB and uRLLC coexistence within orthogonal frequency division multiple access (OFDMA)-based Open RAN architecture in the presence of unknown dynamic traffic demands and slice isolation; 3) Deep reinforcement learning (DRL)-based TS scheme in dynamic environments managing diverse services considering slice awareness technique and flexible numerologies; and 4) Hierarchical optimization-based intelligent UA, congestion control, and resource scheduling scheme align with the 7.2x functional split (FS) in Open RAN architecture assuming non-orthogonality between radio units (RUs).

Firstly, we examine the TS scheme for eMBB and uRLLC services, considering known traffic demand, slice isolation, and fixed numerology, to efficiently allocate heterogeneous network resources via both small-cell and macro-cell RUs in Open RAN, a promising paradigm. While we also formulate a rigorous analysis for the end-to-end (e2e) uRLLC latency, taking account of all factors of computation and communication. Herein, a successive convex approximation (SCA)-based iterative algorithm is proposed to solve the proposed problem. The results of our simulations demonstrate the effectiveness of our proposed centralized optimization method compared to other benchmark schemes.

Secondly, we shift our focus to an intelligent TS scheme within the proposed disaggregated Open RAN architecture, aiming to optimize resource utilization in the context of unknown dynamic traffic demands and flexible numerologies. In this setting, we develop an LSTM-based TS solution that integrates dynamic UA and radio resource management (RRM). Our findings demonstrate the effectiveness of the proposed algorithms compared to benchmark schemes, such as fixed numerology and uniform UA, while highlighting the significance of applying LSTM for predicting traffic demands with minimal measured mean square error (MSE).

Thirdly, we explore a novel centralized DRL-based intelligent TS algorithm within the Open RAN architecture, aimed at enhancing overall system performance while reducing complexity. Our focus is on deploying a multi-layer optimization and learning framework designed for different timescales within the non-real-time (non-RT) and near-real-time (near-RT) RAN intelligent controllers (RICs). Optimization on shorter timescales is performed at the RAN layer, which adapts to dynamic environments by incorporating inferences and policies from the RICs. The simulation results confirm the system's effectiveness in intelligently steering traffic through a slice-awareness scheme, significantly improving eMBB throughput.

Lastly, we address critical challenges in the dynamic landscape of *NextG* wireless networks by proposing an intelligent UA, congestion control, and resource scheduling scheme for a system model based on Open RAN architecture. This study focuses on mitigating issues such as frequent handovers and load balancing, which are exacerbated by varying traffic demands across different services. By aligning with the 7.2x FS option recommended by the O-RAN Alliance, we develop a hierarchical optimization framework incorporating heuristic methods, SCA, and distributed DRL across various Open RAN components. The simulation results convincingly demonstrate the superior performance of the proposed scheme compared to centralized approaches, highlighting its effectiveness in optimizing network operations in dynamic environments.

Acknowledgements

I would like to express my deepest gratitude to everyone who has supported and guided me throughout the development of this thesis. First and foremost, I am incredibly grateful to my supervisor, Prof. Symeon Chatzinotas, and my mentor, Dr. Thang Xuan VU, for their invaluable guidance, insightful feedback, and continuous encouragement. Their expertise and mentorship were pivotal in shaping my research and improving the quality of this work. I would also like to extend my sincere thanks to the members of my thesis committee, Prof. Florian Kaltenberger and Prof. Adrian Kliks, for their valuable suggestions and constructive criticism, which greatly enhanced the depth and clarity of my research.

I am especially thankful to my colleagues and friends at the Interdisciplinary Centre for Security, Reliability, and Trust (SnT), University of Luxembourg, for their support and collaboration that enriched my learning experience. A special thanks to Dr. Van-Dinh Nguyen for his assistance and technical support and insightful advice to refine ideas and revise the papers. I would like to gratefully acknowledge my former colleague at SnT, Dr. Shree Krishna Sharma, for dedicating his time to be my first mentor at the SIGCOM group and providing me with valuable comments helping me to find my Ph.D. road map. Furthermore, I would like to thank Prof. Björn Ottersten for giving me valuable advice and comments to enhance the quality of one of my research papers at the beginning steps of my doctoral journey.

I owe a profound debt of gratitude to my wonderful parents and family for their unwavering support, patience, and encouragement throughout this journey. And special thanks to my beloved husband, Saman, whose support and patience never let me down. His belief in me has been a constant source of motivation.

Lastly, I am grateful to the University of Luxembourg for providing the resources and facilities necessary for conducting this research. Additionally, I would like to gratefully acknowledge the generous funding of my Ph.D. by the ERC AGNOSTIC project and by the Luxembourg National Research Fund, via project RUTINE, for their financial support, which was crucial in enabling this research. Thank you all for your contributions and support in helping me achieve this milestone.

Preface

This Ph.D. thesis has been carried out from February 2021 to November 2024 at the Interdisciplinary Centre for Security, Reliability, and Trust (SnT), University of Luxembourg, Luxembourg, under the supervision of Prof. Symeon Chatzinotas at SnT, University of Luxembourg, Luxembourg. The time-to-time evaluation of the Ph.D. thesis was duly performed by the CET members constituting the supervisors at SnT, University of Luxembourg, Luxembourg, and Prof. Florian Kaltenberger from Eurecom, France.

Contents

This Ph.D. thesis entitled *"Artificial Intelligence (AI)-enabled Smart Radio Environments for 6G Wireless Networks"* is divided into seven chapters. In Chapter 1, the literature review, limitations of existing works, and contributions of this thesis are described. Chapter 2 provides and details common knowledge about the evolution of radio access network (RAN) over time from distributed RAN (D-RAN) to Open RAN, Open RAN architecture, and machine learning (ML) applications within Open RAN. Chapter 3 presents a joint RAN resource allocation scheme to realize enhanced mobile broadband (eMBB) and ultra-reliable and low-latency communications (uRLLC) coexisting in an orthogonal frequency division multiple access (OFDMA)-based Open RAN system in the presence of known traffic demands while assuming the fixed numerology. Chapter 4 includes the same scenario with unknown traffic demands in which a long-short-term memory (LSTM) method is proposed to predict the dynamic traffics to optimize the network performance. A novel intelligent traffic steering (TS) framework is proposed and analyzed in Chapter 5, where the benefits of adding deep reinforcement learning (DRL) to Open RAN's layers are illustrated to address challenges like varying channel conditions and unpredictable demand fluctuation. In Chapter 6, a novel intelligent user association (UA), congestion control, and resource scheduling scheme is proposed and studied to overcome frequent handovers and load balance faced with the varying traffic demands of different services in dynamic environments. Finally, Chapter 7 provides concluding remarks and future research directions.

Support of the Thesis

This Ph.D. thesis has been supported in whole, or in part, by the ERC AGNOSTIC project (ref. H2020/ERC2020POC/957570/DREAM) and by the Luxembourg National Research Fund via project RUTINE (ref. C22/IS/17220888). Additionally, the time-to-time support from SIGCOM is also gratefully acknowledged.

Publications

The original publications that have been produced during the period of Ph.D. candidacy are listed below. These publications are referred to in the text by J ≡ Journal and C ≡ Conference.

Journal Papers

[J1] **F. Kavehmadavani**, V.D. Nguyen, T.X. Vu, and S. Chatzinotas, **"Intelligent Traffic Steering in Beyond 5G Open RAN based on LSTM Traffic Prediction,"** in *IEEE Transactions on Wireless Communications,* March 2023, Doi: 10.1109/TWC.2023.3254903.

[J2] **F. Kavehmadavani**, V.D. Nguyen, T.X. Vu, and S. Chatzinotas, **"Empowering Traffic Steering in 6G Open RAN with Deep Reinforcement Learning,"** (Early Access) in *IEEE Transactions on Wireless Communications,* May 2024, Doi: 10.1109/TWC.2024.3396273.

[J3] **F. Kavehmadavani**, T.X. Vu, V.D. Nguyen, and S. Chatzinotas, **"Intelligent User Association and Scheduling in Open RAN: A Hierarchical Optimization Framework,"** in *IEEE Transactions on Communications,* - Major Review.

Conference Papers

[C1] **F. Kavehmadavani**, V.D. Nguyen, T.X. Vu, and S. Chatzinotas, **"Traffic Steering for eMBB and uRLLC Coexistence in Open Radio Access Networks,"** in *2022 IEEE International Conference on Communications Workshops (ICC Workshops),* Seoul, Korea, May 2022, Doi: 10.1109/ICCWorkshops53468.2022.9814611.

[C2] **F. Kavehmadavani**, V.D. Nguyen, T.X. Vu, and S. Chatzinotas, **"On Deep Reinforcement Learning for Traffic Steering Intelligent ORAN,"** in *2023 IEEE Globecom Workshops (GC Wkshps),* Kuala Lumpur, Malaysia, December 2023. Doi: 10.1109/GCWkshps58843.2023.10464606.

[C3] **F. Kavehmadavani**, T.X. Vu, and S. Chatzinotas, **"Intelligent User Association and Resource Scheduling in Open RAN with 7.2x Functional Split,"** in *2024 IEEE Globecom Conference,* Cape Town, South Africa, December 2024. (Accepted)

Publications not Included in the Thesis

[J4] **F. Kavehmadavani**, M. Soleimanpour-Moghadam, S. Talebi, S. Chatzinotas, and B. Ottersten, **"Joint Resource Allocation for Full-Duplex Ambient Backscatter Communication: A Difference Convex Algorithm,"** in *IEEE Transactions on Wireless Communications,* April 2022. Doi: 10.1109/TWC.2022.3163718.

List of Abbreviations

3GPP	third-generation partnership project
5G	fifth-generation
6G	sixth-generation
AI	artificial intelligence
AWGN	additive white Gaussian noise
BS	base station
COTS	commercial off-the-shelf
C-RAN	cloud radio access network
CSI	channel state information
CU	central unit
DQN	deep Q-network
D2QN	double deep Q-network
D3QN	dueling double deep Q-network
DL	down-link
DRL	deep reinforcement learning
D-RAN	distributed radio access network
DU	distributed unit
eMBB	enhanced mobile broadband
e2e	end to end

FH	fronthaul link
FS	functional split
IoT	Internet of Things
LSTM	long-short-term memory
MA	multi-agent
MADRL	multi-agent deep reinforcement learning
MC	multi-connectivity
MH	midhaul link
MINCP	mixed integer non-convex program
MINLP	mixed integer non-linear program
MISO	multiple-input single-output
ML	machine learning
MSE	mean square error
mMTC	massive machine type communications
near-RT	near-real-time
non-RT	non-real-time
NR	new radio
NS	network slicing
OFDMA	orthogonal frequency division multiple access
O-RAN	open radio access network
PHY	physical-layer
QoE	quality of experience
QoS	quality of service
RAN	radio access network
RAT	radio access technology
RB	resource block
RF	radio frequency

RIC	ran intelligent controller
RL	reinforcement learning
RRM	radio resource management
RNN	recurrent neural network
RU	radio unit
SCA	successive convex approximation
SINR	signal-to-interference-plus-noise ratio
SNR	signal-to-noise ratio
SMO	service management and orchestration
TS	traffic steering
TTI	transmission time interval
UA	user association
UAVs	unmanned aerial vehicles
UE	user equipment
uRLLC	ultra-reliable and low-latency
vRAN	virtualized radio access network

List of Tables

List of Figures

Contents

Chapter 1

Introduction

Fifth-generation (5G) and beyond, often referred to as *NextG* wireless networks, mark a significant departure from the radio-focused nature of previous generations of mobile communication systems. These networks represent a transformative shift towards more efficient architectures, aimed at enhancing the end-user experience and enabling new services, ecosystems, and revenue streams [1]. Designed with advanced capabilities, 5G offers significantly higher data rates, lower latency, increased capacity, and more efficient spectrum utilization, which collectively support a wide range of usage scenarios and applications.

The International Telecommunication Union-Radio Communication Sector (ITU-R) [2] has categorized the potential use cases for 5G networks into three main groups: enhanced mobile broadband (eMBB), ultra-reliable low-latency communications (uRLLC), and massive machine-type communications (mMTC), as illustrated in Fig. 1.1. eMBB extends traditional mobile broadband services by providing high data rates and supporting applications such as augmented reality, 4K streaming, and enterprise collaboration. These applications are enhanced through the high-speed connectivity that eMBB delivers. uRLLC is designed for services that require ultra-high reliability and low latency, such as industrial wireless control and intelligent transportation systems. These services demand precise control and safety, which uRLLC ensures through its robust communication protocols [3]. mMTC focuses on internet of things (IoT) services, characterized by large numbers of devices, such as those in smart cities and remote monitoring systems. This category prioritizes low device complexity, long battery life, and extensive coverage while managing a vast array of devices with relaxed latency and low data rate requirements [4, 5].

However, the existing radio access network (RAN) architectures struggle to meet the diverse requirements of these service categories due to their inflexible, "one-size-fits-all" design. This rigidity hampers efficient resource allocation and adaptation to varying service demands, leading to suboptimal performance and user experiences [6]. To overcome these challenges, the Open RAN architecture has emerged as a promising solution. It enables network operators to dynamically adjust to changing traffic patterns, latency needs, and throughput demands across different services [7]. Key use cases for Open RAN include traffic steering (TS) and user association (UA). These involve distributing traffic loads across

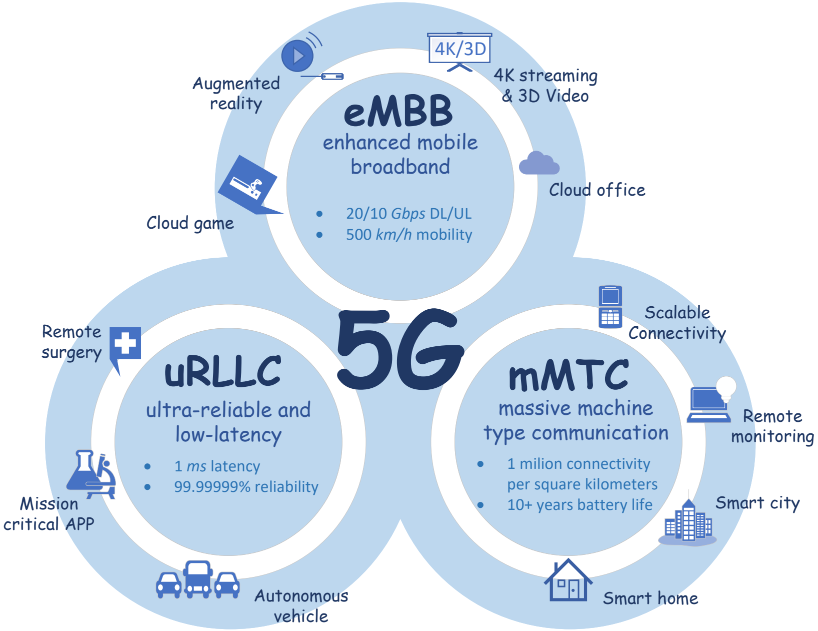


FIGURE 1.1: Overview of the three core service categories in 5G mobile network based on ITU-R.

various radio access technologies (RATs) within the RAN and managing the mobility of individual user equipment (UE) within the network [8]. Despite the flexibility and intelligence offered by Open RAN, it faces several challenges, particularly in IMT-2020 infrastructure. These challenges include unpredictable traffic demands, cell congestion, load balancing, and frequent handovers in UE-centric multi-traffic scenarios [9]. Addressing these issues requires the development of intelligent frameworks capable of optimizing RAN performance in dynamic environments.

1.1 Related Works

In this section, we provide an overview of the most relevant studies that relate to the primary contributions of this dissertation. Specifically, we discuss the related work in three key areas: radio resource management (RRM) in multi-traffic scenarios such as coexistence of eMBB, mMTC, and uRLLC; UA and TS in traditional RAN architectures; and the advancements in intelligent TS and UA within the context of Open RAN.

1.1.1 Radio Resource Management in Multi-traffic Scenarios

The coexistence of eMBB, mMTC, and uRLLC within a single network supports a range of applications with varying demands. However, their differing requirements classify them as heterogeneous services, creating a substantial challenge for RRM to ensure seamless integration. To tackle this challenge, extensive research has investigated how RRM can manage these heterogeneous services within traditional cellular RAN architectures, leveraging cutting-edge 5G technologies. Among these, network slicing (NS) offers a solution by creating customized network slices tailored to meet the diverse requirements of eMBB, mMTC, and uRLLC services, including data rates, connectivity, latency, and reliability. For instance, the authors in [10–14] have studied the resource scheduling for coexistence of three major services *i.e.*, eMBB, mMTC, and uRLLC in 5G networks. In [10], various NS schemes for uplink performance were analyzed, comparing orthogonal multiple access (OMA) and non-orthogonal approaches for eMBB, uRLLC, and mMTC services with a focus on the benefits of rate-splitting multiple access. The authors in [11] investigated the benefits of non-orthogonal sharing of RAN resources in uplink communications for eMBB, mMTC, and uRLLC devices using a technique called heterogeneous non-orthogonal multiple access (NOMA). A scheduling problem of allocating time-frequency resources with different numerologies to support heterogeneous services in 5G systems was proposed in [12] to maximize user scheduling while satisfying their service delay and data transmission requirements. The study in [13] proposed a fairness-aware uplink resource allocation scheme for 5G RAN slicing, focusing on uRLLC, eMBB, and mMTC, introducing a new optimization problem and a hierarchical framework for efficient resource and power allocation. A deep reinforcement learning (DRL) approach was proposed in [14] to address NS challenges in beyond 5G networks by handling uncertainties in demand and channel state information (CSI) to maximize utility for eMBB, uRLLC, and mMTC, proposing a non-convex model and applying a recurrent deterministic policy gradient algorithm.

Since uRLLC and eMBB are two vital services in 5G networks, recent research has primarily focused on their coexistence using advanced 5G technologies. The inherent trade-offs between latency, reliability, and spectral efficiency make radio resource allocation between eMBB and uRLLC a complex scheduling challenge. Numerous studies have recently explored the coexistence of eMBB and uRLLC under various scenarios such as NS [15–22]. For example, the studies in [15, 20] tackled the co-scheduling of eMBB and uRLLC traffic using a puncturing technique. For instance, the optimization problem in [15] aimed to maximize the minimum expected achieved rate for eMBB users while meeting uRLLC requirements. The approach included a penalty successive upper bound minimization-based algorithm for eMBB scheduling and a transportation model for uRLLC. While [16] examined the performance of orthogonal and non-orthogonal multiple access for multiplexing eMBB and uRLLC users in the uplink of a multi-cell cloud RAN (C-RAN). The authors in [17] studied a 5G network supporting both uRLLC and eMBB, proposing a dynamic resource allocation scheme using a two-dimensional bitmap for finer granularity and lower false cancellation rates. They also introduced a power control method to ensure uRLLC reliability with minimal impact on eMBB and a dynamic selection mechanism with load prediction for adaptive scenario handling. In [18], a puncturing scheme was proposed to address the coexistence of uRLLC and eMBB services in 5G networks by formulating an optimization problem that maximizes the minimum expected achieved rate of eMBB users while meeting uRLLC requirements. Besides, balancing both services with conflict requirements in the presence of limited resources is one of the crucial tasks in multi-traffic scenarios in 5G networks. To this end, to minimize the impact on eMBB users while ensuring uRLLC reliability, the authors in [19] proposed a risk-sensitive approach for allocating resources in 5G networks. Besides, to address the scheduling challenges arising from spectrum sharing between uRLLC and eMBB services, [20] proposed a co-scheduling approach using a puncturing technique within multiple-input multiple-output (MIMO) NOMA systems. The primary objective of [20] was to maximize the eMBB data rate while meeting uRLLC latency requirements through joint user selection and power allocation scheduling. Additionally, the stringent latency and reliability demands of uRLLC UEs require explicit prioritization. To address this, the authors in [21] developed a framework that employs preemptive priority service at new radio (NR) base stations (BS). This framework aimed to simultaneously support both services in industrial environments by leveraging stochastic geometry and queuing theory. Besides, the coexistence of uRLLC and eMBB quality of service (QoS) on the same radio spectrum creates a complex scheduling optimization challenge. This requires novel scheduling solutions that cross-optimize system performance based on a UE-centric approach rather than the traditional network-centric approach used in current cellular technologies. Accordingly, the authors in [22] proposed a null-space-based spatial preemptive scheduler for joint uRLLC and eMBB traffic in densely populated 5G networks to maximize eMBB ergodic capacity while ensuring uRLLC requirements are met. In [23], Mehdi *et al.* proposed a hierarchical deep learning framework to support eMBB and URLLC network slices in a shared RAN infrastructure. They applied numerology, mini-slot-based transmission, and punctured scheduling techniques to optimize resource allocation.

Inspired by industrial IoT, [24] proposed using NOMA to enhance the connectivity of uRLLC devices in the uplink to the same BS, applying this approach to both orthogonal and non-orthogonal NS scenarios with eMBB devices. In [25], a framework was proposed that incorporates multiple uRLLC and multicast eMBB slices within a C-RAN, considering the finite blocklength capacity to capture the uRLLC delay while utilizing the multicasting to improve the eMBB throughput. This framework aimed to maximize the C-RAN operator's revenue by efficiently managing the admission of RAN slice requests while adhering to limited physical resource constraints. To enhance efficiency and reduce computational complexity in radio resource management frameworks, [26, 27] also proposed the intelligent RRM scheme designed to meet the stringent latency, reliability, and high data rate requirements for both uRLLC and eMBB services, using DRL-based architecture in two phases, including eMBB resource allocation in and uRLLC scheduling, by leveraging the puncturing technique.

To further enhance network performance and ensure more reliable connections in beyond-5G wireless networks, multi-connectivity (MC) has been developed as an efficient alternative to the carrier aggregation technique. For example, the authors in [28] examined how this technique can be applied to meet the strict reliability requirements of downlink (DL) uRLLC service in 5G networks. Their approach involved utilizing multiple BSs to transmit to a uRLLC service by preempting time-frequency resources initially allocated to eMBB UEs. A dynamic MC-based joint scheduling framework with TS for eMBB and uRLLC was proposed in [29], assuming NS to avoid the queue of uRLLC, which improves the throughput of eMBB under the guarantee of uRLLC latency. The authors in [30] also introduced millimeter-wave (mmWave) communications and MC into 5G uRLLC systems to improve coexistence with eMBB, enhancing spectrum efficiency and reliability. They developed a resource management strategy that maximizes eMBB throughput while meeting uRLLC's strict requirements.

1.1.2 User Association and Traffic Steering in Traditional RAN

Extensive research has focused on optimizing user association to improve network performance. Key studies in this area have explored various strategies, including dynamic user association algorithms, resource allocation techniques, and their impacts on network efficiency and service quality [31–38]. For instance, [31] addressed the challenge of optimizing user association in small cell networks to improve capacity by actively directing users to lighter-loaded tiers via proposing a low-complexity distributed algorithm while achieving near-optimal performance with minimal loss. A joint user association and resource allocation problem for the downlink transmission focusing on the QoS demands of uRLLC and eMBB services was proposed in [32] to efficiently manage IoT applications with strict QoS requirements in fog networks with limited resources. For example, in [33], a context-aware user-cell association method was proposed to balance traffic and satisfy QoS requirements in small cell networks using a novel matching theory-based algorithm. In addition, a joint user association, admission control, and power allocation problem in heterogeneous C-RAN was formulated in [35] to maximize network throughput. The problem, categorized as a mixed-integer non-linear problem (MINLP), was addressed using an outer approximation approach based on linear programming to simplify the NP-Hard problem. The authors in [36] formulated a user association problem in 5G networks with dense small cell deployment and mmwave backhaul to maximize both energy and spectrum efficiency while maintaining user QoS. In [37], a framework was developed to jointly optimize energy efficiency, spectrum efficiency, and queue length in 5G long-term evolution-advanced (LTE-A) heterogeneous networks while addressing complex resource allocation, user association, and power control problems using mixed-integer programming and the drift-plus-penalty approach for stochastic optimization. In [38] a joint optimization functional split (FS), centralized unit-distributed unit (CU-DU) assignment, BS working mode, user association, and routing scheme were formulated to minimize total expenditure and complexity in virtualized RAN (vRAN).

To dynamically direct traffic across different cells and BSs to balance loads and improve overall efficiency, several studies in the literature have primarily focused on investigating TS in traditional RAN networks [39–44]. Among these efforts, the authors in [39] developed schemes for dynamic TS and energy-efficient RAN moderation in 5G networks by leveraging multiple radio links for integrating LTE and 5G networks. The main aims were to reduce packet delivery time by increasing capacity and reliability while also decreasing power consumption through efficient network operation. An energy-sustainable TS framework was proposed in [40], which dynamically adjusts the traffic load to align with energy distributions across both spatial and temporal domains through inter-tier and intra-tier steering, pushing, and caching. To efficiently address dynamic network conditions, [41] proposed a network-aided TS technique for 5G mobile systems. This technique leveraged a machine learning (ML) algorithm to identify dynamic network conditions and manage traffic across multiple RATs based on the ratio of measured throughput, thereby enhancing network efficiency. For instance, in [42], a multiservice-type based transmission strategy for optimizing traffic scheduling in multipath transport was developed to enhance both communication throughput and network resilience by effectively managing multiple network paths while meeting delay QoS constraints. Besides, in [43], a green TS framework was proposed to conserve energy in green mobile communications. This framework aimed to reduce overall network power consumption and maximize aggregated throughput by employing software-defined radio technology, spectrum-based cell zooming techniques, and the long-term evolution of the licensed assisted access concept. In addition, the authors in [44] took advantage of the MC technique to present a TS approach that leverages network and UE performance metrics to maximize the perceived quality of experience (QoE) of eMBB UEs, focusing on improving throughput and mean opinion score as key QoE metrics.

1.1.3 Intelligent Traffic Steering and User Association in Open RAN

Building on the discussion of traffic steering in traditional RAN, which focuses on optimizing resource allocation and managing service demands through established methods, we now turn our attention to the advancements in the Open RAN realm. In Open RAN, the approach to traffic steering is evolving with the integration of intelligent, software-defined techniques. These new methods leverage advanced data analytics, machine learning, and network programmability to enhance the flexibility and efficiency of traffic management. Intelligent TS uses advanced algorithms, often integrating artificial intelligence (AI) and ML, to dynamically allocate network resources in real time, focusing on improving network efficiency. However, it is important to note that research into intelligent traffic steering and user association specifically within the Open RAN context remains limited. For example, a joint user association-placement optimization problem under resource and QoS constraints within Open RAN architecture was formulated in [45] to enhance user admissibility, reduce deployment costs, and improve fairness. Authors in [46] studied a distributed TS mechanism between LTE and Wi-Fi networks based on a learning mechanism that enables each UE to select the proper network under dynamic network conditions. In [47], the problem of resource imbalance caused by overloaded connectivity, which results in excessive requests to the nearest radio remote head (RRH), was addressed using a dynamic RRH gateway steering method. This approach employed a lightweight K-Nearest Neighbor (KNN) supervised learning algorithm to direct UE requests to the most suitable RRHs, optimizing QoS in real-time IoT networks. The study in [48] also introduced an inter-slice resource block (RB) leasing and association adjustment scheme for Open RAN architecture, including a prediction module for forecasting RB demand, a leasing module for balancing resource isolation and costs using an iterative strategy, and an RB association adjustment module to ensure interference isolation.

However, these related works mainly focused on optimizing TS via ML algorithms in traditional RANs that are closed and inflexible in response to heterogeneous networks. To address this issue, the Open RAN paradigm offers an “open” architecture that enables intelligent optimization of the RAN at the UE level due to its nature of architecture, which empowers the application of AI/ML techniques in each layer. For example, a multi-layer intelligent TS framework utilizing reinforcement learning (RL), inner approximation, and bisection search methods was proposed in [49] to efficiently route traffic to appropriate radio units (RUs). While the authors in [50] proposed a novel UE-specific O-RAN intelligent TS framework that leverages convolutional neural networks (CNNs) to optimally assign BSs to UEs. They implemented this framework within the innovative non-standalone Open RAN (ns-O-RAN) software and evaluated its performance on a large-scale deployment. In [51], an intelligent TS scheme for uRLLC applications within Open RAN was presented to avoid network congestion while reducing the queuing delay based on a two-layered ML algorithm. However, existing TS methods suffer from performance degradation due to unnecessary handovers. To avoid the ping-pong phenomenon, [52] proposed an ML-aided intelligent TS scheme including UE classification, throughput prediction, and the TS technique guaranteeing an even load distribution among cells while maintaining an acceptable throughput level. Due to the disaggregated nature of the Open RAN architecture, it also enables emerging federated learning (FL) methods within its layers. To exemplify, [53] proposed a federated meta-learning-based algorithm for RAT allocation within the Open RAN paradigm. This algorithm rapidly adapts to dynamically changing environments to meet UE demands and deliver higher QoS values. In [54], a novel hierarchical RL-aided load-aware TS framework was proposed to meet the diverse QoS requirements for various traffic types. This framework consists of a meta-controller and a controller: the meta-controller sets thresholds for load balancing, while the controller handles traffic admission to the appropriate RAT at a lower level.

1.2 Motivation and Limitations of Existing Works

In this section, we point out some of the disadvantages, limitations, and challenges in the current literature that directly motivate this dissertation. In the following, the main limitations of the existing intelligent TS studies have been raised in three parts to motivate this thesis to address them.

- The rigidity of traditional RAN, as a monolithic architecture built on black-box hardware, offers limited flexibility in adapting to dynamic network conditions and heterogeneous service demands. While traditional RAN supports features like remote configuration and management to some extent, it lacks the modularity and programmability necessary for comprehensive, on-demand reconfiguration and optimization across the entire network. This rigidity makes it challenging to effectively address the conflicting demands of heterogeneous services in beyond-5G wireless networks, especially in scenarios requiring a holistic view of network conditions for real-time decision-making and resource allocation. Hence, one of the best approaches to managing this issue is to use TS, one of the primary use cases in 5G networks, which routes UE flows by the most appropriate radio resources. Although a few of the mentioned studies [14, 23] have employed ML methods to reduce complexity, the majority [10–13, 15–22, 24–30] have relied on heuristic methods or difference convex algorithms, which often result in high complexity and low accuracy due to the coupling between optimization variables and their combinatorial nature.

- Although TS has been extensively studied in various wireless system models prior to Open RAN, further research is needed to explore TS within the Open RAN architecture. Such research can provide new insights, enhance network efficiency, and contribute to the development of fully automated networks. Specifically, the studies mentioned in [39–44] do not examine TS schemes tailored for multi-traffic downlink (DL) orthogonal frequency division multiple access (OFDMA) systems in 5G Open RAN architecture. While these works address network-centric optimization, they fall short of considering automated control and optimization of network functionalities that are customized to the specific needs of individual UEs in heterogeneous scenarios. Moreover, they do not leverage state-of-the-art 5G technologies in a comprehensive manner. For example, integrating dynamic multi-connectivity (MC), network slicing (NS), flexible numerologies, mini-slots, and slice-awareness into an optimal TS framework to meet diverse QoS requirements remains largely unexplored in the context of Open RAN.

- While the features enabled by Open RAN, such as intelligence and closed-control loops among its nodes, could theoretically be implemented in traditional RAN systems, the modular and open nature of Open RAN significantly simplifies and enhances their deployment. Most existing works focus on step-by-step design and development of AI-based xApps for the near-RT RIC within the O-RAN architecture but often overlook the mathematical aspects of the models and the integration of cutting-edge 5G technologies. In contrast, studies such as [46, 47, 49–54] have explored intelligent TS in Open RAN, leveraging ML algorithms. Open RAN's centralized RICs provide a powerful platform for directing traffic effectively, particularly in UE-centric scenarios targeting individual QoS requirements. However, these studies often fail to address critical aspects, such as routing, congestion control, dynamic and unpredictable traffic demands, and UE-centric conditions, which are essential for achieving multi-layer QoS in Open RAN environments.

This dissertation addresses the challenges of intelligent TS in Open RAN scenarios, particularly in managing the conflicting demands of heterogeneous services such as eMBB and uRLLC, handling dynamic and unpredictable traffic conditions, and optimizing resource allocation while ensuring multi-layer QoS. These challenges are compounded by the need for scalable and automated solutions that leverage the modular and flexible nature of the Open RAN architecture. An outline of the contributions made in this thesis is provided in the following section.

1.3 Thesis Outline and Contributions

The contributions of this dissertation are divided into four main chapters, with each chapter detailing and highlighting its objectives to underscore the key achievements. First, in Chapter 3, we elaborate a general framework for TS within Open RAN architecture in the presence of known traffic demand. Then, we study a DL OFDMA-based coexistence eMBB and uRLLC scenario addressing the unknown dynamic traffic demand via long-short-term memory (LSTM) model in Chapter 4. This framework has been used within Chapter 5 exploring a computationally efficient design of intelligent TS by merging a centralized DRL model into an Open RAN architecture for multi-traffic scenarios. Finally, in Chapter 6, we investigate an intelligent UA and congestion control framework through a decentralized DRL model employed in the Open RAN architecture with 7.2x FS.

Chapter 3: Traffic Steering for eMBB and uRLLC Coexistence in Open Radio Access Networks – Part I: Known Traffic Demand

We investigate a general framework to formulate and address the TS scheme for multi-traffic scenarios within Open RAN architecture in Chapter 3.

- First, we provide a DL OFDMA TS framework based on MC and NS techniques to centrally allocate network resources within the Open RAN architecture to maximize eMBB throughput while minimizing worst user uRLLC latency under known traffic demands. This framework assumes fixed numerology and isolated slices for both eMBB and uRLLC services and benefits from both small and macro cells for RUs.
- Second, an efficient iterative algorithm that ensures at least a locally optimal solution is proposed to address the relaxed optimization problem. This approach is based on the successive convex approximation (SCA) method.
- Finally, we provide numerical results, proving the fast convergence behavior of the proposed algorithm and verifying its effectiveness in terms of eMBB throughput and end-to-end (e2e) uRLLC latency, compared with two schemes: uniform power allocation and random resource allocation.

The output of this chapter is published in:

[C1] **F. Kavehmadavani**, V.D. Nguyen, T.X. Vu, and S. Chatzinotas, "Traffic Steering for eMBB and uRLLC Coexistence in Open Radio Access Networks," in *2022 IEEE International Conference on Communications Workshops (ICC Workshops),* Seoul, Korea, May 2022, Doi: 10.1109/ICCWorkshops53468.2022.9814611.

Chapter 4: Traffic Steering for eMBB and uRLLC Coexistence in Open Radio Access Networks – Part II: LSTM Traffic Prediction

This chapter provides a joint intelligent traffic prediction, dynamic RAN slicing, flow-split distribution, and RRM scheme befitting the Open RAN architecture in the presence of unknown dynamic traffic demands.

- To achieve the maximum eMBB throughput while assuring the minimal uRLLC latency requirement and vice versa, we propose a joint intelligent traffic prediction, flow-split distribution, dynamic RAN slicing, and RRM framework under unknown dynamic traffic demands within the OFDMA-based Open RAN architecture. To this end, two optimization problems are developed, each with a separate objective function that is fitted to the service while meeting the following constraints: QoS requirements, slice isolation, power budget, and maximum fronthaul (FH) capacity and queuing buffer.

- Then, to adapt to dynamic environments on different time scales, we decompose each formulated optimization problem into two long-term and short-term subproblems. The long-term subproblem is mapped into three dependent rAPPs at non-real-time (non-RT) RIC, aiming to predict dynamic traffic demands, RAN slicing, and flow-split decisions. By using the SCA method, the resulting non-convex short-term subproblem deploying at the near-real-time (near-RT) RIC is transformed into a form that is more computationally tractable.
- Numerical results are presented to illustrate the rapid convergence of the proposed algorithm and to validate its effectiveness in comparison to the following benchmark schemes: 1) fixed numerology; 2) equal flow-split distribution; 3) single connectivity; and 4) the proposed framework under known traffic demands.

The achievement of this chapter is published in the following venue:

[J1] **F. Kavehmadavani**, V.D. Nguyen, T.X. Vu, and S. Chatzinotas, "Intelligent Traffic Steering in Beyond 5G Open RAN Based On LSTM Traffic Prediction," in *IEEE Transactions on Wireless Communications,* March 2023, Doi: 10.1109/TWC.2023.3254903.

Chapter 5: Empowering Traffic Steering in 6G Open RAN with Deep Reinforcement Learning

In Chapter 5, a multi-layer intelligent TS framework assuming slice-awareness technique within the OFDMA-based Open RAN is presented to centrally address challenges like high complexity, varying channel conditions, and dynamic traffic demands.

- We introduce a novel intelligent TS framework utilizing dynamic MC, slice-aware RAN slicing, and mixed numerology multiplexing in both frequency and time domains to minimize the long-term average queue length of eMBB UEs and the long-term average uRLLC latency, considering QoS requirements, slice awareness, power budget, and traffic flow-split decisions.
- This framework handles the lack of complete information such as time-varying CSI and queue length, reducing computational complexity by making decisions per frame instead of each time slot through a centralized ML method. Therefore, RICs at the upper Open RAN layers and DUs at the function Open RAN layer handle a two-stage optimization strategy on distinct time frames. Customized xAPP at near-RT RIC handles the long-term subproblem (frame structure), while DUs handle the short-term subproblem (time slot structure).
- Lastly, using a large-scale simulation set, we evaluate the performance of our method in comparison to schemes like slice isolation, SCA, *etc.*.

The outputs of this chapter are published in:

[J2] **F. Kavehmadavani**, V.D. Nguyen, T.X. Vu, and S. Chatzinotas, "Empowering Traffic Steering in 6G Open RAN with Deep Reinforcement Learning," in *IEEE Transactions on Wireless Communications,* (Early Access), May 2024, Doi: 10.1109/TWC.2024.3396273.

[C2] **F. Kavehmadavani**, V.D. Nguyen, T.X. Vu, and S. Chatzinotas, "On Deep Reinforcement Learning for Traffic Steering Intelligent ORAN," in *2023 IEEE Globecom Workshops (GC Wkshps),* Kuala Lumpur, Malaysia, December 2023. Doi: 10.1109/GCWkshps58843.2023.10464606.

Chapter 6: Intelligent User Association and Scheduling in Open RAN with 7.2x Functional Split: A Hierarchical Optimization Framework

In Chapter 6, we propose a joint intelligent UA, congestion control, and resource scheduling scheme to address two critical challenges: frequent handovers and load balancing within the Open RAN architecture, assuming a 7.2x FS. While a classical RAN architecture could support similar functionalities, the 7.2x FS offers key advantages in flexibility, modularity, and efficient distribution of network functions. By separating functions across RAN nodes, it enables more granular control over resource allocation, dynamic traffic management, and decentralized power control, which is critical for minimizing latency. This approach enhances traffic steering and congestion management, leveraging the capabilities of the near-RT RIC, DU, and RUs. In this scenario, UEs served by a single RU experience orthogonality in terms of resource allocation, while interference between different RUs can occur due to non-orthogonality, impacting the overall performance.

- Aligning with the 7.2x FS option recommended by the O-RAN Alliance, we proposed a novel intelligent UA, congestion control, and resource scheduling scheme, catering to both eMBB and uRLLC services. The formulated problem aims to minimize eMBB queue lengths and uRLLC latency while considering congestion control, power budgets, and other practical constraints.

- A hierarchical optimization approach combining heuristic, iterative SCA, and distributed DRL-based algorithms is proposed to solve the formulated problem. Different Open RAN's nodes serve as hosts for each of the aforementioned algorithms, according to 7.2x FS. To achieve this, the customized UA-xAPP located at near-RT RIC is in charge of updating UA variables via a heuristic algorithm, while physical resources are optimized in CUs through SCA method and a distributed DRL-based algorithm is deployed in RUs to optimize the power.

- Eventually, we provide comprehensive numerical results that demonstrate the superior performance of the proposed solution compared to the centralized scheme, OFDMA, and other schemes.

The outputs of these contributions are outlined below:

[J3] **F. Kavehmadavani**, T.X. Vu, V.D. Nguyen, and S. Chatzinotas, "Intelligent User Association and Scheduling in Open RAN: A Hierarchical Optimization Framework," in *IEEE Transactions on Wireless Communications.*

[C3] **F. Kavehmadavani**, T.X. Vu, and S. Chatzinotas, "Intelligent User Association and Resource Scheduling in Open RAN with 7.2x Functional Split," in *2024 IEEE Globecom Conference,* Cape Town, South Africa, December 2024.

Chapter 7: Conclusions and Future Research

Finally, Chapter 7 provides the conclusions of this dissertation and a discussion about the potential avenues for future works.

Chapter 2

Background

A mobile network architecture consists of three key components: the core network (CN), the radio access network (RAN), and the user equipment (UE), as shown in Fig. 2.1 (a) [55]. The CN serves as the backbone of the network, managing access, mobility, and essential services like interconnectivity to ensure seamless communication across the network. The RAN connects the UE to the CN through radio links, comprising base stations (BSs) that include a radio unit (RU) for signal transmission and reception, along with a baseband unit (BBU) that handles radio resource management. Finally, the UE, such as a smartphone, is the device that users employ to access the mobile network, typically authenticated via a subscriber identity module (SIM) card [56].

2.1 Evaluation of RAN before Open RAN

Before Open RAN, traditional RAN architectures—such as distributed RAN (D-RAN), cloud RAN (C-RAN), and virtualized RAN (vRAN)—were characterized by their proprietary and monolithic designs [57]. Created by a small group of vendors, these systems featured tightly integrated hardware and software, which limited their flexibility and stifled innovation. However, over the last three decades, BSs have transitioned from these rigid,

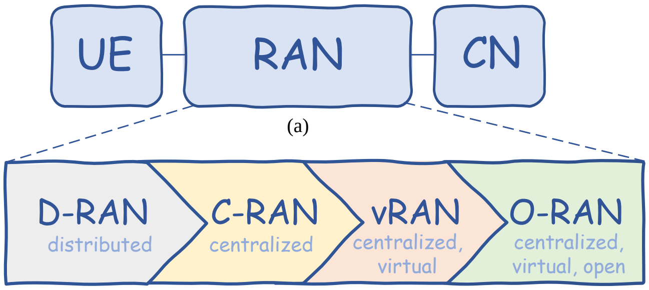


FIGURE 2.1: The mobile network architecture and the evolution of RAN

monolithic devices to more flexible configurations that use commercial off-the-shelf (COTS) hardware [58]. This shift underscores the limitations of traditional RAN and highlights the need for Open RAN, which aims to provide a more open, flexible, and interoperable architecture. This section explores the main aspects of traditional RAN and the various types of BSs to understand the challenges that prompted the development of Open RAN. The evolution of RAN over time is illustrated in Fig. 2.1 (b).

D-RAN represents one of the first traditional mobile network architecture where the baseband processing and radio functions are co-located at each cell site, in which they are directly connected via a common public radio interface (CPRI). In this setup, both the RU and BBU are housed together, with the BBU typically placed in an air-conditioned shelter and the RU, sometimes referred to as the RRU, located near the antenna or at the top of the tower. This proximity offers low-latency communication between the RU and BBU, but it also presents significant challenges. The independent operation of each BS makes resource management and coordination across multiple sites difficult, leading to inefficiencies, especially in densely populated areas. Moreover, the proprietary nature of D-RAN, where both hardware and software are tightly integrated, restricts flexibility and increases the cost of network expansions and upgrades [59]. As user demand grew, requiring more BSs to handle increased traffic, the costs associated with space, cooling, and specialized hardware became substantial. These limitations highlighted the need for more scalable and cost-effective solutions, driving the evolution toward centralized architectures like C-RAN.

In 2009, C-RAN was introduced to overcome the limitations of D-RAN by providing a more cost-effective solution. In C-RAN, BBUs from multiple BSs are centralized, which significantly reduces site rental, operational, and maintenance costs. This centralization allows for the addition of new RRUs without requiring new BBUs at each cell site, thus meeting increasing traffic demands without a substantial rise in costs [60]. C-RAN also offers benefits over D-RAN, such as lower operational costs due to the reduced need for BBUs and the capability to use advanced techniques like coordinated multipoint (CoMP) transmission and reception. However, like D-RAN, C-RAN still has proprietary interfaces among BBUs, RRUs, and fronthaul (FH), leading to vendor lock-in and limiting flexibility. Despite these issues, C-RAN represents a significant advancement toward more scalable and flexible mobile network architectures, setting the stage for further developments like vRAN and Open RAN, which aim to further disaggregate network functions and utilize cloud-based technologies.

vRAN represents a significant advancement in mobile network architecture, evolving from D-RAN and C-RAN by introducing greater flexibility and scalability. In vRAN, the proprietary BBU hardware used in earlier architectures is replaced with COTS servers, and network functions are virtualized through network function virtualization (NFV) principles. This setup allows network functions to run on virtual machines or containers on general-purpose servers, enabling resource sharing across multiple sites and potentially cutting data processing needs by about 50% [61]. Although vRAN offers benefits such as centralized management and cost savings, it also brings challenges like increased network complexity and the need for high-capacity fronthaul (FH) connections. Nonetheless, vRAN provides a more adaptable and cost-effective approach, paving the way for future developments in

TABLE 2.1: Comparison of different types of RAN

Types	**BBU hardware**	**BBU software**	**Virtualization**	**RIC**	**Vendor lock-in**
D-RAN	Proprietary H/w	Proprietary S/w	-	-	✓
C-RAN	Proprietary H/w	Proprietary S/w	-	-	✓
vRAN	COTS	Proprietary S/w	✓	-	✓
O-RAN	COTS	Open	✓	✓	-

open and interoperable frameworks like Open RAN.

Despite advancements with D-RAN, C-RAN, and vRAN, moving to Open RAN is crucial for future network infrastructure. As illustrated in Table. 2.1, Open RAN provides a flexible, interoperable approach by using open interfaces and standardized components, which supports multi-vendor environments and reduces vendor lock-in. Unlike traditional, proprietary RAN systems, Open RAN's open architecture fosters innovation and customization [62]. This shift enhances resource management, cost-effectiveness, scalability, and adaptability to new technologies, making it essential for meeting growing demands and promoting a more competitive industry.

2.2 Open RAN Architecture

Open RAN surpasses vRAN by emphasizing *openness*, *intelligence*, and *interoperability*. It standardizes key interfaces, such as the open FH between the RRU and BBU, allowing operators to integrate components from various vendors. This flexibility enables operators to customize their networks more effectively, moving away from the proprietary, single-vendor constraints of traditional RAN systems like D-RAN and vRAN. Open RAN introduces a modular design by dividing the traditional BBU into central units (CUs), distributed units (DUs), and RUs; each component is designed to handle specific functions within the RAN, collectively enabling efficient and scalable network operations. These units can be deployed independently and managed centrally. Additionally, by utilizing COTS hardware and software-defined radio (SDR), Open RAN lowers deployment costs and improves the network's adaptability to evolving demands. Another crucial aspect of Open RAN is its capability to decouple the RAN control plane from the user plane. This decoupling facilitates more dynamic and efficient network management while integrating advanced, data-driven intelligence to automate RAN operations. By fostering a more open, flexible, and intelligent network environment, Open RAN stands out as a pivotal innovation, poised to meet the complex challenges of modern and future mobile communications. The O-RAN architecture is built on four foundational principles: **virtualization**, **disaggregation**, **open interfaces**, and **intelligence**, each contributing to a more flexible, cost-effective, and future-proof mobile network.

- **Virtualization** is a core tenet of O-RAN, where the RAN functions are decoupled from proprietary hardware and implemented as software running on generic, off-the-shelf hardware platforms. This approach not only reduces reliance on specialized equipment but also enables operators to scale their networks dynamically in response to fluctuating demands. Virtualization also simplifies the orchestration and management of RAN functions through abstraction layers, ultimately leading to significant reductions in operational costs and improving overall network efficiency.

- **Disaggregation** in O-RAN refers to the splitting of traditional BS functions into distinct, logical units: the RU, DU, and CU. The CU itself can be further divided into control plane (CU-CP) and user plane (CU-UP) components, enabling more granular control over where and how network functions are deployed. This separation allows operators to optimize their networks by distributing these units across different locations and hardware platforms, depending on the specific requirements of the deployment environment.
- **Open Interfaces** are another critical principle of O-RAN, enabling interoperability between equipment from various vendors. By adhering to standardized interfaces, O-RAN allows for a mix-and-match approach to network components, giving operators the flexibility to choose the best-in-class solutions for their specific needs. This vendor-agnostic approach not only enhances innovation and competition in the market but also helps operators avoid the pitfalls of vendor lock-in, ensuring that they can adapt their networks as new technologies and solutions emerge.
- **Intelligence** is embedded within the O-RAN framework through the introduction of the RAN intelligent controller (RIC), which plays a pivotal role in optimizing network performance. The RIC is available in two forms: the non-real-time (non-RT) RIC for non-real-time operations and optimization, and the near-real-time (near-RT) RIC for tasks requiring near-instantaneous response. These controllers provide a platform for integrating advanced algorithms and artificial intelligence (AI)/machine learning (ML)-driven applications, enabling more sophisticated management of RAN resources and the automation of complex tasks such as radio resource management (RRM) and network optimization.

The main elements in the Open RAN architecture based on O-RAN Alliance are briefly described below as shown in Fig. 2.2.

In the RAN part, the CU is the hub for higher-layer processing tasks for managing the control and user plane functions, such as radio resource control (RRC), packet data convergence protocol (PDCP), and service data adaptation protocol (SDAP). The duties of CU extend to critical processes like scheduling, mobility management, and encryption, which are essential for maintaining the integrity and performance of the network. By splitting the CU into two parts—CU-CP and CU-UP—the architecture allows for a more modular and scalable network design, enhancing the overall efficiency and flexibility of the RAN. To support inter-cell coordination, the X2/Xn interfaces connect CUs and DUs across different sites. The DU plays a crucial role in real-time processing, handling the lower layers of the RAN, including radio link control (RLC), medium access control (MAC), and aspects of the physical layer (PHY). Positioned closer to the network edge, the DU is vital for low-latency communications, making it indispensable for time-sensitive applications. It manages radio resource allocation, link adaptation, and re-transmissions, ensuring that the

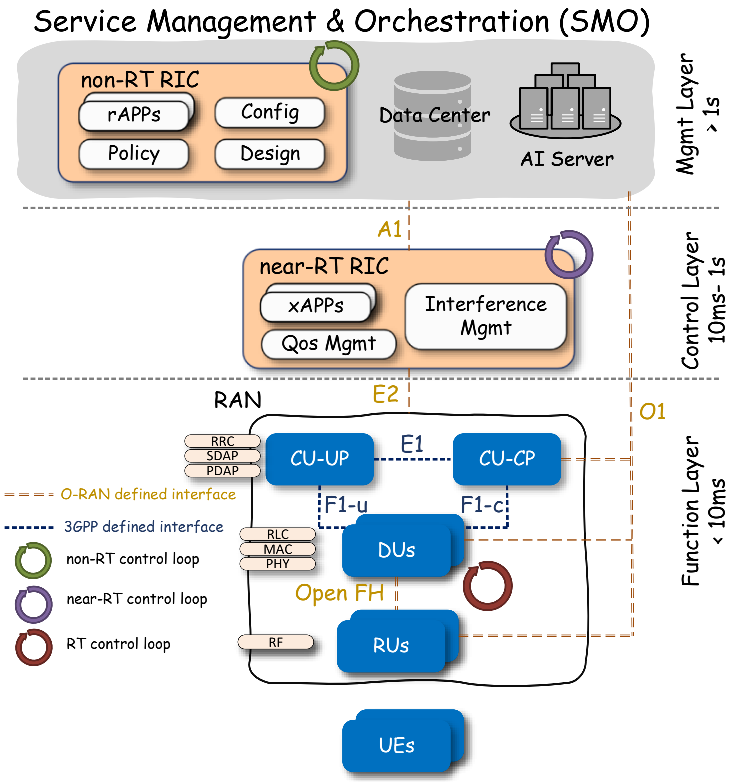


FIGURE 2.2: High-level architecture of O-RAN

data transmission between the RU and CU is both reliable and efficient. The RU is the front line of the RAN, directly responsible for the transmission and reception of radio signals. It includes both the antenna systems and radio frequency (RF) components necessary for wireless communication. The RU handles the digital front-end processes and parts of the PHY layer, such as modulation and coding. By converting digital data into radio signals and vice versa, the RU facilitates the seamless connection between the UE and the broader network infrastructure. These components form a disaggregated RAN architecture in the function layer (< 10 millisecond) that offers unprecedented flexibility.

On top of the RAN part, the service management and orchestration (SMO) framework oversees the orchestration and management of the RAN domain in the O-RAN architecture. Another hallmark of the O-RAN architecture is the introduction of RICs in both management and control layers, which are central to the intelligent management and optimization of the network. The architecture differentiates between two types of RICs: the non-RT RIC and near-RT RIC, each with distinct roles and responsibilities. The SMO coordinates the deployment and operation of RAN components, utilizing interfaces like O1, O2, and the Open FH M-Plane. It also interacts with the non-RT RIC, providing management functions across various RAN elements and ensuring the network operates efficiently and flexibly. The non-RT RIC, embedded within the SMO framework in the management layer, handles tasks that do not require immediate response, allowing for strategic, long-term optimizations. Operating on a timescale of over one second, it focuses on policy management, data analysis, and the training of AI/ML models that influence broader network behavior. The non-RT RIC runs rAPPs, which are modular applications designed to enhance network efficiency and provide services like resource management and trend analysis. These rAPPs interact with the non-RT RIC via the open R1 interface, ensuring interoperability and flexibility. Thanks to the A1 interface that connects the non-RT RIC to the near-RT RIC, facilitating AI-driven optimization, the near-RT RIC is designed for more immediate, time-sensitive operations, with control loops that range from 10 milliseconds to one second. This controller directly interacts with RAN elements like the DU and CU through the E2 interface, executing near-instantaneous decisions to optimize traffic flow, manage network load, and enhance user experience. xAPPs, which are micro-services running on the near-RT RIC, carry out these functions by analyzing real-time data and making swift adjustments to the network. These xAPPs, similar to rAPPs, are vendor-agnostic and can be developed by third parties, contributing to the overall agility and adaptability of the RAN.

To create a more efficient architecture, an optimal functional split (FS) strategy is essential. The various FSs among different components provide significant flexibility and customization, enabling the optimization of network performance within the Open RAN architecture, as depicted in Fig. 2.3. A key split in this architecture is the CU-DU split, where these units, connected via the F1 interface, allow the CU to manage higher-layer functions such as RRC and PDCP, while the DU oversees lower-layer functions like RLC and MAC. This division enhances scalability and operational efficiency by separating user plane and control plane functions, thus accommodating varying traffic demands and reducing latency. Further refinement is achieved through the CU-UP split, which separates the CU into CU-CP and CU-UP components, linked by the E1 interface. The CU-CP takes on RRC and control-plane PDCP functions, whereas the CU-UP manages the user-plane SDAP and PDCP. Additionally, the DU-RU split introduces another level of granularity. Although 3GPP has not standardized this split, options like split 7.2x and split 6 are supported by industry O-RAN Alliances. Split 7.2x involves the DU and RU handling different layers of the protocol stack, while Split 6 (lower layer split) divides the functionalities differently. These splits enable further customization of network deployment, tailored to specific operational and geographical needs.

2.3 Machine Learning in Open RAN

Traditionally, mobile networks relied on static, rule-based methods for tasks such as resource allocation, interference management, and performance monitoring. However, as networks become increasingly complex with the introduction of 5G and beyond, these static approaches are proving inadequate. ML algorithms provide a dynamic solution by analyzing vast amounts of network data to predict and proactively address issues like congestion, coverage gaps, and signal interference. Through techniques such as supervised learning for traffic forecasting and reinforcement learning for spectrum allocation, ML enables networks

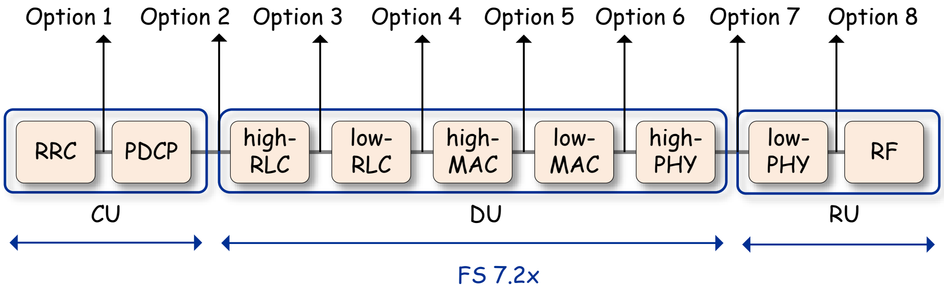


FIGURE 2.3: The functional split options

to self-optimize and adapt in real time to changing user behavior and environmental conditions. This shift towards automation and predictive management significantly enhances the efficiency and reliability of mobile communications, leading to smarter and more adaptable networks.

In the realm of Open RAN, ML is revolutionizing the way such networks are managed and optimized by injecting intelligence at every layer of the network. The Open RAN specifications provide a robust framework for leveraging ML models to enhance various network functionalities, from QoS optimization to traffic steering, predictive handovers, and radio fingerprinting. These advancements are enabled by a systematic ML workflow that integrates seamlessly into the Open RAN architecture. The details of these ML workflow phases will be discussed further in the following [63].

① **Data Collection and Preparation**: The process starts with extensive data collection, forming the backbone of any successful ML initiative. In the context of Open RAN, nodes like CUs, DUs, and RUs continuously produce extensive data—such as traffic patterns and signal quality—which is transmitted to the non-RT RIC via the O1 interface. This raw data is meticulously gathered and curated into large, structured datasets. The quality of these datasets is vital, as it directly impacts the effectiveness of the subsequent ML models.

② **Data Analysis**: Next, the collected data is explored to extract insights and detect patterns. This phase involves using data visualization, statistical analysis, and other exploration techniques to understand the data's underlying structure. This step is crucial for choosing the right ML algorithms and identifying relevant features for model training.

③ **Model Querying and Training**: With this data in hand, the ML model design phase begins. Here, engineers carefully select the critical input parameters, such as network throughput or latency, and determine the output controls, like resource allocation strategies, RAN slicing policies, or traffic steering. At this stage, the appropriate ML algorithm—whether it is a neural network, decision tree, or reinforcement learning model—is implemented based on the specific use case. After tuning model hyper-parameters and training, the model is evaluated and validated to ensure its accuracy and reliability. This phase is critical to ensuring that the model can accurately predict network conditions and make intelligent decisions. After training, the model is rigorously tested on unseen data to verify its ability to generalize and perform effectively even under unforeseen network conditions.

④ **Model Deployment**: Once validated, the ML model is packaged as an xAPP and deployed on the near-RT RIC, where it connects to the RAN via the E2 interface. It integrates with the network operations, accesses relevant data, and provides real-time predictions or recommendations. The xAPP then makes real-time adjustments to optimize network performance, such as steering traffic, adjusting slicing parameters, or managing handovers, while continuously learning from new data.

⑤ **Model Optimization**: After deployment, the model requires optimization to enhance its performance. This involves real-time performance monitoring, identifying areas for improvement, and updating the model as needed. This phase is essential for maintaining the accuracy and reliability of the ML model over time.

⑥ **Model Updating**: The final step involves ongoing maintenance of the ML model. This includes updating the model as needed and ensuring it remains aligned with the evolving requirements of the O-RAN network operations environment.

By harnessing the capabilities of ML algorithms, network operators can enhance network performance, lower energy usage, and deliver an improved user experience. Practical use cases of ML in Open RAN highlight its transformative potential [64]. For example, in vehicle-to-everything (V2X) communications, ML can manage dynamic handovers to maintain seamless connectivity. It also optimizes quality of experience (QoE) by adjusting network parameters in real time. Additionally, ML facilitates dynamic resource allocation for unmanned aerial vehicles (UAVs) by adapting to their flight paths, ensuring efficient use of network resources. These examples illustrate how integrating ML within Open RAN architecture promotes intelligent automation and adaptability, enhancing performance across diverse and complex network environments.

Various ML models are applied to address complex challenges across these environments, each offering unique benefits for enhancing performance, resource management, and user experience [65]. Supervised learning models, such as linear regression and decision trees, are used to predict traffic patterns, manage user mobility, and detect faults by learning from labeled data. Unsupervised learning models like k-means clustering and principal component analysis focus on finding hidden patterns in data, helping to cluster similar network behaviors and identify anomalies. These models are especially valuable for understanding large-scale network dynamics without the need for explicit labeling. Reinforcement learning takes dynamic decision-making to the next level by training agents to interact with their environment and learn from feedback. In mobile communications and Open RAN, RL is crucial for optimizing spectrum allocation, managing network resources, and enabling self-organizing networks (SONs). Algorithms like Q-learning and deep Q-networks (DQN) allow for real-time adaptability, helping to balance load, optimize handovers, and reduce interference.

Deep learning, with its powerful neural networks, excels at processing large-scale data and extracting complex patterns. In mobile networks, deep learning models such as conventional neural networks (CNNs) and recurrent neural networks (RNNs) enhance tasks like signal processing, network performance prediction, and QoE optimization. By leveraging massive datasets, deep learning can improve critical functions like beamforming and user satisfaction predictions. Among RNNs, long short-term memory (LSTM) networks stand out as particularly effective for tasks involving sequential data, such as traffic forecasting and network anomaly detection. LSTMs are designed to remember long-term dependencies, making them ideal for predicting time-series data like mobile traffic patterns, where past events heavily influence future behavior. Besides, federated learning offers a privacy-preserving approach by enabling distributed model training across decentralized devices, such as smartphones or BSs, without sharing raw data. This collaborative approach is particularly beneficial in Open RAN, where multiple vendors need to work together while ensuring data security. Transfer learning further complements this by allowing pretrained models to adapt to new but related tasks, which is useful for optimizing network operations across different environments or slices.

Chapter 3

Traffic Steering for eMBB and uRLLC Coexistence in Open Radio Access Networks – Part I: Known Traffic Demand

Existing radio access network (RAN) architectures lack sufficient openness, flexibility, and intelligence to meet the diverse demands of emerging services in beyond fifth-generation (5G) and sixth-generation (6G) wireless networks, including enhanced mobile broadband (eMBB) and ultra-reliable and low-latency (uRLLC). Open RAN is a promising paradigm that allows building a virtualized and intelligent architecture. In this chapter, we focus on traffic steering (TS) schemes based on multi-connectivity (MC) and network slicing (NS) techniques to efficiently allocate heterogeneous network resources in *NextG* cellular networks. We formulate the RAN resource allocation problem to simultaneously maximize the weighted sum eMBB data rate and minimize the worst-case uRLLC latency subject to quality of service (QoS) requirements, orthogonality, power, and limited fronthaul constraints. Since the formulated problem is categorized as a mixed integer nonlinear problem (MINLP), we first relax binary variables to continuous ones and develop an efficient iterative algorithm based on successive convex approximation techniques. System-level simulation results demonstrate the effectiveness of the proposed algorithm compared to several well-known benchmark schemes.

3.1 Introduction

Beyond fifth-generation (5G) and sixth-generation (6G), wireless networks are expected to meet three major services with different demands, *namely* ultra-reliability low-latency communication (uRLLC), enhanced mobile broadband (eMBB), and massive machine-type communications (mMTC). Different uRLLC, which supports traffics requiring extremely high reliability (*i.e.*, 99.999%) and very low latency (*i.e.*, less than 1 ms), eMBB requires high data rate connectivity [11]. Since existing 5G cellular networks are inflexible, closed, and aggregated, it is not possible to enable the coexistence of various services on the existing "one-size-fits-all" 5G architecture. Despite the cost-effectiveness of cloud radio access networks (C-RAN) and virtual RANs (vRAN), these architectures still lack open interfaces and non-proprietary hardware and software. Therefore, Open RAN has been recently proposed to address these issues, evolving towards flexible, virtualized, disaggregated, open, and intelligent *NextG* wireless networks [66].

The key objective of Open RAN is to improve the RAN performance by virtualizing RAN's elements, disaggregation of their components, software, and hardware, defining appropriate open interfaces for connecting them, as well as embedding machine learning (ML)/artificial intelligence (AI) techniques to construct adaptive and smarter RAN layers in its architecture [67]. Two novel modules, including near-real-time (near-RT) and non-real-time (non-RT) RAN intelligent controllers (RICs), are defined to enable a centralized network abstraction to further reduce cost, network complexity, and human-machine interaction [68]. Following a disaggregation approach, base station (BS) functionalities are virtualized as network functions based on third-generation partnership project (3GPP) functional split and are divided across various network nodes, *namely* central units (CUs), distributed units (DUs), and radio units (RUs).

Mobile networks have become increasingly complicated as network capacity and traffic have increased manifold. It is challenging to steer heterogeneous traffics to effectively improve network efficiency and user experience [69]. Network slicing (NS) has appeared as a promising solution to allocate resources to various wireless services with different requirements [70]. To meet the strict latency requirement of uRLLC services, 5G new radio (NR) adopts dynamic numerologies for a short transmission duration and also introduces the concept of mini-slots by diminishing the transmission time interval (TTI), which can be achievable with a larger subcarrier spacing [71]. Multi-connectivity (MC) refers to the ability for user equipment to simultaneously connect to multiple access points or base stations, providing improved reliability, increased throughput, and better network performance, particularly in heterogeneous networks like eMBB and uRLLC. In a 3GPP system, MC is implemented through techniques like dual connectivity (DC) and carrier aggregation (CA), where a user can connect to different cells or base stations using multiple bearers. This approach improves signal-to-interference-plus-noise ratio (SINR), enhances user reliability, and increases communication coverage, making it ideal for supporting both eMBB and uRLLC services in Open RAN.

Traffic steering (TS) is crucial to illustrate the practical applicability of Open RAN architecture. However, the current literature on TS is still sparse and isolated. In [72], a dynamic MC-based joint scheduling framework with TS for both eMBB and uRLLC traffics was proposed. Zhang *et al.* discussed TS in LTE networks with unlicensed bands in [73]. Due to the impact of the NS technique in multi-services networks, the authors in [39] analyzed schemes to enable dynamic TS and energy-efficient RAN moderation in 5G. To the best of our knowledge, there are few works in the literature that model TS within the Open RAN architecture, yet they lack a detailed mathematical investigation of the topic. For instance, [74] analyzed a study of the TS use case implemented in a modular way, following the open networking approach.

Unlike typical TS mechanisms that treat all users in the same way without considering user demands, in this chapter we propose a novel TS scheme using dynamic MC technique and RAN slicing scheme to steer traffic flows towards the most suitable cells based on user-centric conditions. Our main contributions are summarized as follows:

- We develop a joint RAN resource allocation framework as xAPP in near-RT RIC to dynamically optimize eMBB and uRLLC traffics over the same resources. The utility function of interest combines the sum eMBB data rate and worst-case uRLLC latency. In addition, a rigorous analysis for the uRLLC latency model will be provided that takes into account all factors of computation and communication. Existing RUs in the site will generate MC clusters, which vary according to the characteristics of the network. This, in turn, achieves a higher sum eMBB data rate while maintaining the uRLLC latency requirement.

- We propose an efficient iterative algorithm based on the successive convex approximation (SCA) method to solve the relaxed optimization problem, which guarantees at least a locally optimal solution.

- Numerical results are provided to show the fast convergence behavior of the proposed algorithm and verify its effectiveness, compared with benchmark schemes.

3.2 System Model

We consider a downlink (DL) orthogonal frequency-division multiple access (OFDMA) in the Open RAN architecture, consisting of one CU and a set $\mathcal{N} \triangleq \{1, 2, \ldots, N\}$ of N DUs. These two types of processing nodes run on the general-purpose data centers as virtual machines (VMs) by virtual network function (VNF) technology, which can process incoming user packets in parallel, as illustrated in Fig. 3.1. Towards cost-efficient deployment, we assume that DUs serve non-overlapped geographical areas, *i.e.*, each DU serves a cluster of RUs.

3.2.1 Network Model

Let $\mathcal{M}_n \triangleq \{0, 1, \ldots, m, \ldots, M_n\}$ indicates the set of RUs served by the n-th DU, which consists of one macro RU (MRU), *i.e.*, RU 0, and M_n small-cell RUs (SRUs), *i.e.*, RU $m \in \{1, \cdots, M_n\}$. SRUs' unique ability to handle high-density data makes them the ideal choice to satisfy demands of services in 5G networks, especially uRLLC. In contrast, the MRU provides a high data rate and extended coverage to eMBB users. SRUs are also used to support the control plane and provide a quick response to small packet data sizes [29].

In addition, we denote by $\mathcal{U}_n = \{1, \ldots, U_n\}$ the set of users served by DU n, which can be further divided into two disjoint sets $\mathcal{U}_n^{\mathsf{ur}}$ of U_n^{ur} uRLLC users and $\mathcal{U}_n^{\mathsf{em}}$ of U_n^{em} eMBB users. The m-th RU is equipped with K_{tx} antennas, while users are equipped with a single antenna and are randomly distributed across the network area.

Under MC configuration, the MRU's operating frequency is orthogonal to that of SRUs. Denote by $\mathcal{F}_0$ and $\mathcal{F}_1$ the sets of sub-band frequencies operated by MRU and SRUs, respectively. The numbers of sub-bands operated by MRU and SRUs are $F_0 = |\mathcal{F}_0|$ and $F_1 = |\mathcal{F}_1|$,

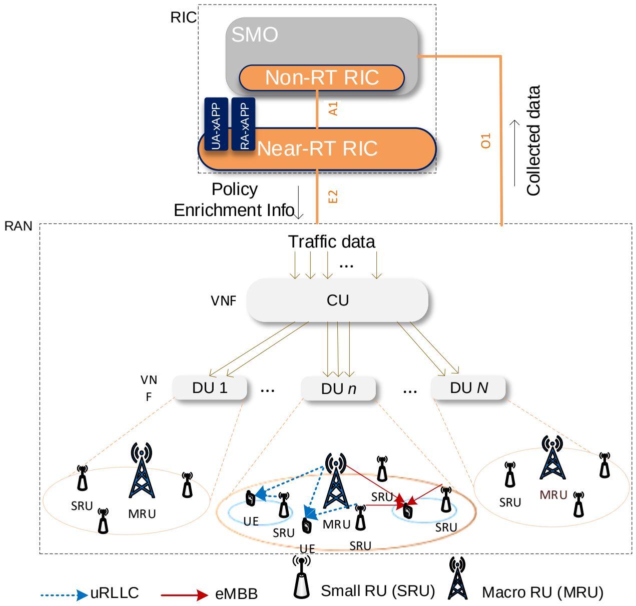


FIGURE 3.1: Open RAN-based system model.

respectively. Each transmission frame is divided into T time slots whose duration is equal to one TTI. The bandwidth of each sub-band defined in 3GPP 5G NR is equal to 360 KHz. Therefore, one resource block (RB) with the time duration of mini-slot ($\Delta_t = 0.25$ ms) corresponds to 7 OFDM symbols.

Achievable Rate: Our goal is to optimize the RAN performance in three-dimensional (3D), encompassing time, frequency, and power domains. The problem casts into assigning the total $(F_0 + F_1) \times T$ RBs of all RUs covered by the considered DU to its users. Because of the non-overlapped DUs' coverage, the resource optimization design at one DU is similar to that of other DUs. Thus, for ease of presentation, we drop the subscript index of DUs hereafter.

Let $\boldsymbol{h}_{t,f,m,u} \in \mathbb{C}^{K_m \times 1}$ be the channel vector from the m-th RU to the u-th user equipment (UE) at the sub-band f and the time slot t, including the path loss. Within each frame, assume that the channel remains temporally invariant, while it may be different across the sub-bands. In this work, we employ maximal ratio transmission (MRT) to maximize the received signal-to-noise ratio (SNR), which is also the optimal beamformer under the considered OFDMA scheme [75]. As a result, the effective channel gain for UE u served by RU m at sub-band f and time slot t is $g_{t,f,m,u} \triangleq \|\boldsymbol{h}_{t,f,m,u}\|_2^2$.

Given the orthogonality constraint, this work considers that each RB of a RU is assigned to only one single user during one time slot. To introduce this assignment, we define the decision variables $\pi^{\mathsf{em}}_{t,f,m,u} \in \{0,1\}$ and $\pi^{\mathsf{ur}}_{t,f,m,u} \in \{0,1\}$ for eMBB and uRLLC traffics, respectively. Here, $\pi^{\mathsf{em}}_{t,f,m,u} = 1$ if the RB associated with time-slot t and sub-band f of RU m assigned to the u-th eMBB user, and $\pi^{\mathsf{em}}_{t,f,m,u} = 0$ otherwise; similarly definition for uRLLC users. The achievable rate in bits/s for a given set of channel realizations at the u-th eMBB user is given by

$$R_u^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}) = \sum_{t,f,m} \beta \log_2 \Big(1 + \frac{p_{t,f,m,u}^{\mathsf{em}} g_{t,f,m,u}}{N_0}\Big) \tag{3.1}$$

where β, N_0, and $p_{t,f,m,u}^{\mathsf{em}}$ are the bandwidth of each RB, the power of the additive white Gaussian noise (AWGN), and transmit power from RU m to UE u for eMBB traffic at subband f at the TTI t, respectively. Let us define $\boldsymbol{p}^{\mathsf{em}} \triangleq [p_{t,f,m,u}^{\mathsf{em}}]_{\forall t,f,m,u}$. We note that there is no inter-user interference in (3.1) since it is cancelled via OFDMA constraints introduced latter in this section. Furthermore, the transmit power must satisfy $p_{t,f,m,u}^{\mathsf{em}} \leq \pi_{t,f,m,u}^{\mathsf{em}} P_m^{\max}$ with $P_m^{\max}$ being the power budget at RU m, which guarantees that RU m allocates power to user u on RB (t, f) only if $\pi_{t,f,m,u}^{\mathsf{em}} = 1$; otherwise $\pi_{t,f,m,u}^{\mathsf{em}} = 0$ and $p_{t,f,m,u}^{\mathsf{em}} = 0$. As a result, the sum data rate of eMBB users is given as $\mathcal{R}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}) = \sum_{u\in\mathcal{U}^{\mathsf{em}}} R_u^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}})$. The minimum QoS requirement for eMBB users is guaranteed by the constraint $R_u^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}) \geq R^{\mathsf{th}}$, where R^{th} is a given QoS threshold.

In contrast, owing to the finite block length in uRLLC traffics, the Shannon capacity is no longer used to obtain the data rate for uRLLC users. The achievable rate of u-th user for uRLLC traffic using the short block-length can be expressed as [76]

$$R_u^{\mathsf{ur}}(\boldsymbol{p}^{\mathsf{ur}}, \boldsymbol{\pi}^{\mathsf{ur}}) = \textstyle\sum_{t,f,m} \beta \Big[\log_2 \Big(1 + \frac{p_{t,f,m,u}^{\mathsf{ur}} g_{t,f,m,u}}{N_0}\Big) - \log_2(e) \frac{\pi_{t,f,m,u}^{\mathsf{ur}} \sqrt{V} Q^{-1}(P_e)}{\sqrt{\Delta_t \beta}}\Big] \tag{3.2}$$

where V, P_e, and Q^{-1}: $\{0, 1\} \to \mathbb{R}$ denote the channel dispersion, error probability, and inverse of the Gaussian Q-function, respectively. It is observed that $V = 1 - \frac{1}{(\mathrm{SNR})^2} \approx 1$ when the received SNR=$\frac{p_{t,f,m,u}^{\mathsf{ur}} g_{t,f,m,u}}{N_0} \geq \Gamma_0$ with $\Gamma_0 \geq 5$ dB. This can be easily achieved in cellular networks by arranging the uRLLC decoding vector into one possible null space of the reference subspace [77]. Hence, we consider the constraint $\frac{N_0\beta}{g_{t,f,m,u}} \Gamma_0 \pi_{t,f,m,u}^{\mathsf{ur}} \leq p_{t,f,m,u}^{\mathsf{ur}} \leq \pi_{t,f,m,u}^{\mathsf{ur}} P_m^{\mathsf{max}}$ to guarantee the approximation $V \approx 1$ as well as the big-M formulation theory to avoid non-convexity of (3.2).

uRLLC Latency Model: Unlike eMBB users, the main aim for uRLLC users is to minimize their end-to-end (e2e) latency from CU to end users. To meet stringent latency requirements, uRLLC traffics should be immediately served, resulting in no queuing delay. As a result, the e2e latency of uRLLC users consists of the processing and transmission time.

As depicted in Fig. 3.1, after processing, the uRLLC users's arrival packets at the CU layer are subsequently routed to VNFs in the DU layer for parallel processing. We adopt the $M/M/1$ processing queue model on a first-come, first-serve basis to serve each user's packets. The packet process of the u-th arrival packet for uRLLC traffics generated by the FTP3 model standardized in 3GPP with Z bytes length follows the Poisson process with the mean arrival rate λ_u (packets/frame). It is assumed that each packet has an identical length, and packet segmentation is not allowed. Suppose that each assigned RB to the uRLLC user should transmit at least one complete data packet. In contrast, eMBB users generate continuous traffic with infinite packet size, avoiding packet loss due to buffer overflow.

Denote by f_{cu} and f_{du} the computation capacities of CU and DU (cycles/sec), respectively. Considering the identical packet size, the required computation resource to process one packet of size Z is C (number of cycles). As a result, $\mu_{cu} = f_{cu}/C$ and $\mu_{du} = f_{du}/C$ are the task rates (1/sec) at CU and DU, respectively. As a result, $1/\mu_{cu}$ and $1/\mu_{du}$ represent the mean service times of the CU and DU layers, respectively. The processing latency of the uRLLC arrival packet at the CU layer (τ_{cu}^{pro}) and DU layer (τ_{du}^{pro}) is computed as

$$\tau_{cu}^{\mathsf{pro}} = 1/(\mu_{cu} - \Lambda) \text{ and } \tau_{du}^{\mathsf{pro}} = 1/(\mu_{du} - \Lambda) \tag{3.3}$$

where $\Lambda = \sum_{u\in\mathcal{U}^{\mathsf{ur}}} \lambda_u$ is the total packet arrival rate for all uRLLC users at the CU layer. We assume that $\mu_{cu} > \Lambda$ and $\mu_{du} > \Lambda$ to guarantee the queue stability.

Next, the arrival packets λ_u for the u-th uRLLC user are transported to DU via the midhauld (MH) link with the maximum capacity C^{MH} (bits/sec). It should be mentioned that the mean arrival data rate of the DU layer is approximately equal to the mean arrival data rate of the first layer. By Burke's theorem, the mean arrival data rate of the second layer, which is processed in the first layer, is still Poisson with the rate Λ [78]. Hence, the data transmission latency of the uRLLC traffic for UE u under the MH limited capacity is

$$\tau_{cu,du}^{\mathsf{tx}} = \frac{\Lambda Z}{C^{\mathsf{MH}}}. \tag{3.4}$$

Using the MC technique, the generated traffic per frame for the u-th uRLLC user is split into several partitions, which are transmitted in separate links, and then aggregated at this user. The maximum number of paths from DU n to each user is M_n. We denote by $\boldsymbol{b}_u \triangleq [b_{0,u}, b_{1,u}, \cdots, b_{M_n,u}]$ the flow-split indicator vector for the u-th uRLLC user. In particular, if $b_{m,u} = 1$, RU $m \in \mathcal{M}_n$ is selected to transmit the data of the u-th uRLLC user; otherwise $b_{m,u} = 0$. In addition, let us denote by $\boldsymbol{\varphi}_u \triangleq [\varphi_{0,u}, \varphi_{1,u}, \varphi_{2,u}, \cdots, \varphi_{M_n,u}]$ the flow-split portion vector of user u with $\sum_{m\in\mathcal{M}_n} \varphi_{m,u} = 1$, where $\varphi_{m,u}$ represents a portion of traffic routed to user u via RU m. Since the packets for the u-th user can be transmitted by multiple RUs, the effective response time $\tau_{du,ru}^{\mathsf{tx}}$ to transport all packets the DU layer should be computed by the worst average response time among its connected fronthaul (FH) links with maximum capacity C_m^{FH} (bits/sec), *i.e.*,

$$\tau_{du,ru}^{\mathsf{tx}} = \max_m \Big\{ \frac{\sum_{u\in\mathcal{U}_n^{\mathsf{ur}}} \varphi_{m,u}\lambda_u Z}{C_m^{\mathsf{FH}}} \Big\}, \ \forall m \in \mathcal{M}_n. \tag{3.5}$$

We denote by $r_{m,u}^{\mathsf{ur}}$ the data rate from RU m to user u, which can be directly extracted from (3.2). The transmission latency from RU m to user u is then calculated as

$$\tau_{ru,u}^{\mathsf{tx}} = \max_m \Big\{ \frac{\varphi_{m,u}\lambda_u Z}{r_{m,u}^{\mathsf{ur}}} \Big\}, \quad \forall u \in \mathcal{U}^{\mathsf{ur}}. \tag{3.6}$$

Simply put, the e2e latency of uRLLC user $u \in \mathcal{U}^{\mathsf{ur}}$ per frame is computed as

$$\tau_u^{\mathsf{ur}} = \tau_{cu}^{\mathsf{pro}} + \tau_{cu,du}^{\mathsf{tx}} + \tau_{du}^{\mathsf{pro}} + \tau_{du,ru}^{\mathsf{tx}} + \tau_{ru,u}^{\mathsf{tx}} + \tau_{ru}^{\mathsf{pro}} + \tau_u^{\mathsf{pro}} + \tau^{\mathsf{align}} \tag{3.7}$$

where $\tau_{ru}^{\mathsf{pro}}, \tau_{u}^{\mathsf{pro}}$ and τ^{align} are the process latency at RU m, uRLLC user u, and frame alignment time, respectively. τ_{ru}^{pro} and τ_{u}^{pro} are bounded by three OFDM symbol durations that are typically very small, while τ^{align} is upper-bounded by one mini-slot TTI interval and can be considered negligible. To ensure a minimum latency requirement for uRLLC user u, the e2e latency is bounded by a predetermined threshold D^{ur}, *i.e.*, $\tau_u^{\mathsf{ur}} \leq D^{\mathsf{ur}}$.

3.3 Joint Traffic Steering and Resource Allocation Optimization

3.3.1 Problem Formulation

Utility function: The main goal is to jointly optimize the RAN slicing and TS to serve eMBB and uRLLC users, subject to various resource constraints and diverse QoS requirements. To do so, the utility function should capture both the sum eMBB data rate and worst-case e2e uRLLC latency, such as: $\alpha \frac{\mathcal{R}^{\mathsf{em}}}{R_0} - (1-\alpha)\max_u \left\{\frac{\tau_u^{\mathsf{ur}}}{\tau_0}\right\}$, where $R_0 > 0$ and $\tau_0 > 0$ are the reference data rate of eMBB and latency of uRLLC, respectively, which are used to balance two different dimensions of the two quantities; and $\alpha \in [0,1]$ denotes the priority parameter. Based on the above definitions and discussions, the problem of joint TS and resource allocation is mathematically formulated as

$$\max_{\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p}} \quad \alpha \frac{\mathcal{R}^{\mathsf{em}}}{R_0} - (1-\alpha)\max_u \left\{\frac{\tau_u^{\mathsf{ur}}}{\tau_0}\right\} \tag{3.8}$$

$$\text{s.t.} \quad \text{C1:} \quad \pi_{t,f,m,u}^{\mathsf{em}},\ \pi_{t,f,m,u}^{\mathsf{ur}} \in \{0,1\},\ \forall t,f,m,u \tag{3.9}$$

$$\text{C2:} \sum_{t,f,m,u} [\pi_{t,f,m,u}^{\mathsf{em}} + \pi_{t,f,m,u}^{\mathsf{ur}}] \leq 1, \forall m = 0, f \in \mathscr{F}_0$$

$$\sum_{t,f,m,u} [\pi_{t,f,m,u}^{\mathsf{em}} + \pi_{t,f,m,u}^{\mathsf{ur}}] \leq 1, \forall m \neq 0, f \in \mathscr{F}_1 \tag{3.10}$$

$$\text{C3:} \sum\nolimits_{m,t,f} \pi_{t,f,m,u}^{\mathsf{ur}} \geq e_u^{\mathsf{ur}}, \quad \forall u \in \mathscr{U}^{\mathsf{ur}} \tag{3.11}$$

$$\text{C4:} \sum\nolimits_{f,u} (p_{t,f,m,u}^{\mathsf{em}} + p_{t,f,m,u}^{\mathsf{ur}}) \leq P_m^{\mathsf{max}},\ \forall t,m \tag{3.12}$$

$$\text{C5:}\ 0 \leq p_{t,f,m,u}^{\mathsf{em}} \leq \pi_{t,f,m,u}^{\mathsf{em}} P_m^{\mathsf{max}},\ \forall t,f,m,u \tag{3.13}$$

$$\text{C6:}\ \frac{N_0 \Gamma_0 \pi_{t,f,m,u}^{\mathsf{ur}}}{g_{t,f,m,u}} \leq p_{t,f,m,u}^{\mathsf{ur}} \leq \pi_{t,f,m,u}^{\mathsf{ur}} P_m^{\mathsf{max}}, \forall t,f,m,u \tag{3.14}$$

$$\text{C7:}\ r_u^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}) \geq R^{\mathsf{th}}, \quad \forall u \in \mathscr{U}^{\mathsf{em}} \tag{3.15}$$

$$\text{C8:} \sum\nolimits_u \left[R_{m,u}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}) + R_{m,u}^{\mathsf{ur}}(\boldsymbol{\pi}^{\mathsf{ur}}, \boldsymbol{p}^{\mathsf{ur}})\right] \leq C_m^{\mathsf{FH}}, \forall m \tag{3.16}$$

$$\text{C9:} \quad R_{m,u}^{\mathsf{ur}}(\boldsymbol{\pi}^{\mathsf{ur}}, \boldsymbol{p}^{\mathsf{ur}}) \geq \varphi_{m,u} \lambda_u Z,\ \forall m, u \in \mathscr{U}^{\mathsf{ur}} \tag{3.17}$$

$$\text{C10:} \sum\nolimits_m \varphi_{m,u} = 1,\ 0 \leq \varphi_{m,u} \leq 1,\ \forall u \in \mathscr{U}^{\mathsf{ur}} \tag{3.18}$$

$$\text{C11:} \quad \tau_u^{\mathsf{ur}}(\boldsymbol{\varphi}_u, \boldsymbol{\pi}^{\mathsf{ur}}, \boldsymbol{p}^{\mathsf{ur}}) \leq D^{\mathsf{ur}},\ \forall u \in \mathscr{U}^{\mathsf{ur}} \tag{3.19}$$

where $\boldsymbol{\varphi}, \boldsymbol{\pi}$ and $\boldsymbol{p}$ are the vectors, encompassing the flow-split portions, sub-band assignments, and power allocation variables, respectively. Here, C2 is the orthogonality constraint to assure that each RB is allocated to only one user (either eMBB or uRLLC). Constraint C3 refers to QoS requirements for uRLLC users, which expresses that every scheduled uRLLC user should be assigned at least $e_u^{\mathsf{ur}} = \lceil \frac{\lambda Z}{B} \rceil$ number of RBs from the dedicated uRLLC slice to empty the available packets in the queues of uRLLC users; and B is the number of bits that each RB carries. Constraint C4 ensures that the total transmission power is no larger than the power budget at RU m, denoted by P_m^{max}. The limited capacity of the FH link between DU and RU m is expressed by constraint C8. Finally, constraint C9 ensures that each RB assigned to the u-th uRLLC user should transmit a complete data packet with size Z.

3.3.2 Proposed SCA-based Iterative Algorithm for Solving (3.8)

Challenges of solving problem (3.8): The main challenges in solving problem (3.8) lie in the non-convexity of τ_u^{ur} (appears in the objective function and constraint C11) and constraint C8 with respect to TS and transmit power variables. Furthermore, the binary nature of the sub-band allocation variables makes the problem more difficult to solve directly. Once may employ the mixed integer linear program (MILP) solver, *e.g.*, Gurobi or MOSEK, to directly solve binary $\boldsymbol{\pi}$. However, we argue that the exponential computation complexity of such MILP formulation limits its practical feasibility, especially when the number of variables exceeds a few hundred in Open RAN scenarios. To tackle these difficulties, we first relax binary variables to continuous ones (*i.e.*, the box constraints between 0 and 1) and transform constraint τ_u^{ur} and C8 into a more traceable form, which can be efficiently solved by an SCA-based iterative algorithm.

Penalty function: In order to speed up the convergence of the proposed iterative algorithm presented shortly, we introduce the following penalty function

$$\mathcal{P}(\boldsymbol{\pi}) = \sum_{t,f,m,u} \left[(\pi_{t,f,m,u}^{\mathsf{em}})^2 + (\pi_{t,f,m,u}^{\mathsf{ur}})^2 - \pi_{t,f,m,u}^{\mathsf{em}} - \pi_{t,f,m,u}^{\mathsf{ur}} \right] \tag{3.20}$$

which is convex in $\boldsymbol{\pi}$. It is clear that $\mathcal{P}(\boldsymbol{\pi}) \leq 0$ for any $\pi_{t,f,m,u} \in [0,1]$, which is useful to penalize the relaxed variables to obtain near-exact binary solutions at optimum (*i.e.*, satisfying C1). By incorporating $\mathcal{P}(\boldsymbol{\pi})$ into the objective function of (3.8), the parameterized relaxed problem is expressed as

$$\max_{\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p}} \quad \alpha \frac{\mathcal{R}^{\mathsf{em}}}{R_0} - (1-\alpha) \max_u \left\{ \frac{\tau_u^{\mathsf{ur}}}{\tau_0} \right\} + \gamma \mathcal{P}(\boldsymbol{\pi}) \tag{3.21a}$$

$$\text{s.t. } \widetilde{\text{C}}1: \ \pi_{t,f,m,u}^{\mathsf{em}}, \ \pi_{t,f,m,u}^{\mathsf{ur}} \in [0,1], \ \forall t,f,m,u \tag{3.21b}$$

$$\text{C2 - C11} \tag{3.21c}$$

where $\gamma > 0$ is a given penalty parameter.

Proposition 1. *With an appropriate positive value of γ, problems* (3.8) *and* (3.21) *share the same optimal solution, i.e.,* $(\boldsymbol{\varphi}^*, \boldsymbol{\pi}^*, \boldsymbol{p}^*)$.

The proof is directly followed [79] by showing that $\mathcal{P}(\boldsymbol{\pi}) = 0$ at optimum in maximizing (3.21a). It implies that there always exists a constant γ to ensure that $\boldsymbol{\pi}$ are binary at optimum, so that the relaxation is tight. In practice, it is acceptable if $\mathcal{P}(\boldsymbol{\pi}) \leq \varepsilon$ for a very small ε, which leads to a near-exact optimal solution.

To handle the non-convexity of τ^{ur}_u, we introduce new variables $\boldsymbol{t} \triangleq [t_1, t_2]$ to rewrite (3.21) equivalently as

$$
\begin{aligned}
\max_{\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p},\boldsymbol{t}} \quad & \alpha \frac{\mathcal{R}^{\mathsf{em}}}{R_0} - (1-\alpha)\max_u \left\{ \frac{\widetilde{\tau}^{\mathsf{ur}}_u}{\tau_0} \right\} + \gamma \mathcal{P}(\boldsymbol{\pi}) && (3.22a)\\
\mathsf{s.t.:} \quad & \widetilde{\mathrm{C}}1,\ \mathrm{C}2 \text{ - } \mathrm{C}10 && (3.22b)\\
\widetilde{\mathrm{C}}11: \quad & \widetilde{\tau}^{\mathsf{ur}}_u \leq D^{\mathsf{ur}},\ \forall u && (3.22c)\\
\mathrm{C}12: \quad & r^{\mathsf{ur}}_{m,u} \geq 1/t_1,\ \forall m, u && (3.22d)\\
\mathrm{C}13: \quad & \varphi_{m,u} t_1 \leq \frac{t_2}{\lambda_u Z}, \forall m, u && (3.22e)
\end{aligned}
$$

where $\widetilde{\tau}^{\mathsf{ur}}_u = \tau^{\mathsf{pro}}_{cu} + \tau^{\mathsf{tx}}_{cu,du} + \tau^{\mathsf{pro}}_{du} + \tau^{\mathsf{tx}}_{du,ru} + t_2 + \tau^{\mathsf{pro}}_{ru} + \tau^{\mathsf{pro}}_{u} + \tau^{\mathsf{align}}$. The equivalence between (3.21) and (3.22) is easily verified by showing equality of C12 and C13 at optimum. In problem (3.22), the objective function is non-concave due to $\mathcal{P}(\boldsymbol{\pi})$, while constraints C8 and C13 are non-convex.

Under SCA method, the function $\mathcal{P}(\boldsymbol{\pi})$ is linearized at the κ-th iteration by the first-order Taylor approximation as

$$
\mathcal{P}^{(\kappa)}(\boldsymbol{\pi}) \triangleq \sum_{t,f,m,u} \left[\pi^{\mathsf{em}}_{t,f,m,u}(2\pi^{\mathsf{em},(\kappa)}_{t,f,m,u} - 1) - (\pi^{\mathsf{em},(\kappa)}_{t,f,m,u})^2 + \pi^{\mathsf{ur}}_{t,f,m,u}(2\pi^{\mathsf{ur},(\kappa)}_{t,f,m,u} - 1) - (\pi^{\mathsf{ur},(\kappa)}_{t,f,m,u})^2 \right] \tag{3.23}
$$

where $\mathcal{P}(\boldsymbol{\pi}) \geq \mathcal{P}^{(\kappa)}(\boldsymbol{\pi})$ and $\mathcal{P}(\boldsymbol{\pi}^{(\kappa)}) \geq \mathcal{P}^{(\kappa)}(\boldsymbol{\pi}^{(\kappa)})$. For C8, we denote its left hand sight (LHS) as $R_m(\boldsymbol{p}) \triangleq \sum_u \left[R^{\mathsf{em}}_{m,u}(\boldsymbol{p}^{\mathsf{em}}) + R^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}) \right]$, which is concave in $\boldsymbol{p}$ and can be approximated at $\boldsymbol{p}^{(\kappa)}$ as

$$
\begin{aligned}
R^{(\kappa)}_m(\boldsymbol{p}) \triangleq R_m(\boldsymbol{p}^{(\kappa)}) - \sum_{t,f,u} \beta \log_2(e) \frac{\pi^{\mathsf{ur}}_{t,f,m,u} Q^{-1}(P_e)}{\sqrt{\Delta_t \beta}} + \frac{\beta}{\ln 2} \sum_{t,f,u,\mathsf{x}} (p^{\mathsf{x}}_{t,f,m,u} - p^{\mathsf{x},(\kappa)}_{t,f,m,u}) \times \\
\left[\frac{g_{t,f,m,u}}{N_0 + p^{\mathsf{x},(\kappa)}_{t,f,m,u} g_{t,f,m,u}} \right]
\end{aligned} \tag{3.24}
$$

where $\mathsf{x} \in \{\mathsf{em}, \mathsf{ur}\}$. For C13, it is rewritten equivalently as $(\varphi_{m,u} + t_1)^2 \leq 2\frac{t_2}{\lambda_u Z} + \varphi^2_{m,u} + t_1^2$, where both sides are convex. By the first-order Taylor approximation, we convexify C13 as

$$
\widetilde{\mathrm{C}}13: (\varphi_{m,u} + t_1)^2 \leq \frac{2t_2}{\lambda_u Z} + \Phi^{(\kappa)}(\varphi_{m,u}, t_1) \tag{3.25}
$$

where $\Phi^{(\kappa)}(\varphi_{m,u}, t_1) \triangleq 2\varphi^{(\kappa)}_{m,u}\varphi_{m,u} + 2t^{(\kappa)}_1 t_1 - (\varphi^{(\kappa)}_{m,u})^2 - (t^{(\kappa)}_1)^2$.

Algorithm 1 SCA-based Iterative Algorithm to Solve (3.8)

```
1: Initialization: Set $\kappa := 1$ and generate initial feasible points for $(\boldsymbol{\varphi}^{(0)}, \boldsymbol{\pi}^{(0)}, \boldsymbol{p}^{(0)}, \boldsymbol{t}^{(0)})$ to constraints in (3.26)
2: repeat
3: Solve (3.26) to obtain $(\boldsymbol{\varphi}^*, \boldsymbol{\pi}^*, \boldsymbol{p}^*, \boldsymbol{t}^*)$ and $\psi^*$;
4: Update $(\boldsymbol{\varphi}^{(\kappa)}, \boldsymbol{\pi}^{(\kappa)}, \boldsymbol{p}^{(\kappa)}, \boldsymbol{t}^{(\kappa)}) := (\boldsymbol{\varphi}^*, \boldsymbol{\pi}^*, \boldsymbol{p}^*, \boldsymbol{t}^*)$ and $\psi^{(\kappa)} := \psi^*$;
5: Set $\kappa := \kappa + 1$;
6: until Convergence or $|\psi^{(\kappa)} - \psi^{(\kappa-1)}| \leq \delta$ {/*Satisfying a given accuracy level*/}
7: Recover an exact binary by computing $\boldsymbol{\pi}^* = \lfloor \boldsymbol{\pi}^{(\kappa)} + 0.5 \rfloor$ and repeat step 1 to 5 for given $\boldsymbol{\pi}^*$;
8: Output:$(\boldsymbol{\varphi}^*, \boldsymbol{\pi}^*, \boldsymbol{p}^*, \boldsymbol{t}^*)$.
```

Bearing all the above approximations in mind, the convex approximate program of (3.22) solved at iteration κ is given as

$$\max_{\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p},\boldsymbol{t}} \psi^{(\kappa)} \triangleq \alpha \frac{\mathcal{R}^{\mathsf{em}}}{R_0} - (1-\alpha) \max_u \left\{ \frac{\widetilde{\tau}^{\mathsf{ur}}_u}{\tau_0} \right\} + \gamma \mathcal{P}^{(\kappa)}(\boldsymbol{\pi}) \tag{3.26a}$$

$$\mathsf{s.t.}: \widetilde{\text{C}}1, \text{ C2 - C7, C9, C10}, \widetilde{\text{C}}11, \text{C12}, \widetilde{\text{C}}13 \tag{3.26b}$$

$$\widetilde{\text{C}}8: R^{(\kappa)}_m(\boldsymbol{p}) \leq C^{\mathsf{FH}}_m, \ \forall m. \tag{3.26c}$$

The SCA-based iterative algorithm is summarized in Algorithm 1. To guarantee a feasible solution to problem (3.8), Step 6 is performed to recover an exact binary solution and re-run Steps 1-5 to refine the final solution.

Convergence and complexity analysis: Algorithm 1 produces a sequence of improved solutions $\{\boldsymbol{\varphi}^{(\kappa)}, \boldsymbol{\pi}^{(\kappa)}, \boldsymbol{p}^{(\kappa)}, \boldsymbol{t}^{(\kappa)}\}$ (see [80] for more details) as well as a non-decreasing sequence of the objective values $\{\psi^{(\kappa)}\}$, *i.e.*, $\psi^{(\kappa)} \geq \psi^{(\kappa-1)}$. The sequence $\{\psi^{(\kappa)}\}_{\kappa\to\infty}$ is bounded above due to the limited bandwidth and power budgets. By the interior-point method, the per-iteration complexity of Algorithm 1 is $\mathcal{O}\big(\sqrt{c}(v)^3\big)$, where $c = (M_n+1)\big(3U_n(F_0+F_1)T + T + U^{\mathsf{ur}}_n + 1\big) + 3U^{\mathsf{ur}}_n + U^{\mathsf{em}}_n$ and $v = (M_n+1)\big(2U_n(F_0+F_1)T + U^{\mathsf{ur}}_n\big) + 2$ are the numbers of constraints and variables, respectively.

3.4 Numerical Results

In this section, we numerically evaluate the performance of the proposed algorithm. All users are uniformly and randomly positioned in a circular area with a radius of 500 m. MRU is located in the center, serving three sectors, each of which includes two SRUs. The coverage of SRUs is 100 m. The channels are generated as Rayleigh fading with the path-loss $\mathsf{PL}_{\mathrm{MRU-UE}} = 128.1 + 37.6\log_{10}(d/1000)$ for the MRU-UE channels and $\mathsf{PL}_{\mathrm{SRU-UE}} = 38 + 30\log_{10}(d)$ for the SRU-UE channels. The number of frequency sub-bands for MRU and SRUs are $F_0 = 6$ and $F1 = 10$, respectively. Unless stated otherwise, other simulation parameters are given in Table 3.1. For performance comparison, we consider the following two well-known benchmark schemes:

Table 3.1: Simulation Parameters

Parameter	**Value**	**Parameter**	**Value**
No. eMBB users	8	Noise power (N_0)	-110 dBm
No. uRLLC users	4	uRLLC packet size (Z)	32 bytes
Bandwidth of MRU	20 MHz	Length of time-frame	10 ms
Bandwidth of SRU	100 MHz	Time slot (Δ_t)	0.25 ms
Bandwidth of subcarriers (β)	360 KHz	Predetermined latency (D^{ur})	0.5 ms
Power of MRU	46 dBm	Error probability (P_e)	10^{-3}
Power of SRU	30 dBm	MH capacity	50 Gbps
MRU's FH capacity	1000 Mbps	SRUs' FH capacity	500 Mbps

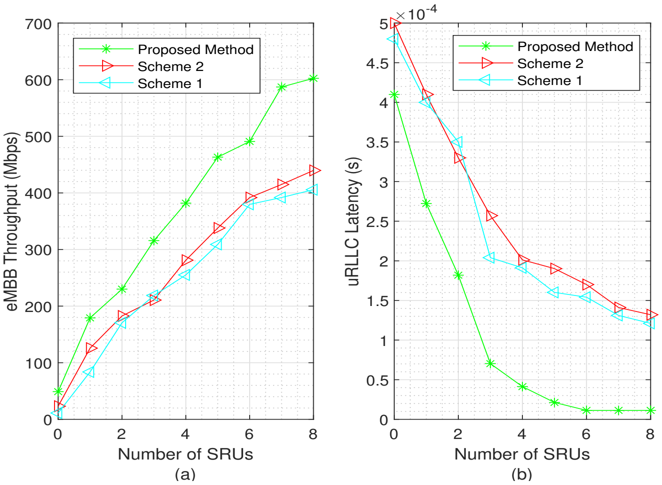


Figure 3.2: The sum eMBB data rate and worst-case uRLLC latency versus the number of SRUs.

- **Scheme 1**: This scheme optimizes only the traffic steering $\boldsymbol{\varphi}$ and RBs allocation $\boldsymbol{\pi}$ and allocates equal power to all users.

- **Scheme 2**: The second scheme randomly allocates RBs to users and optimizes only the traffic steering $\boldsymbol{\varphi}$ and power allocation $\boldsymbol{p}$ variables.

Fig. 3.2 shows the impact of the number of SRUs on the sum eMBB data rate and worst-case uRLLC latency. As expected, the sum eMBB data rate increases and the uRLLC latency decreases as the number of SRUs increases. The sum eMBB data rate achieved by the proposed method is significantly higher than that of benchmark schemes. In addition, the proposed method clearly provides the lowest latency of uRLLC users comparing two other schemes, regardless of the number of SRUs. For instance, the proposed method improves about 50% of the sum eMBB data rate and reduces the uRLLC latency by 90% at $M = 4$, compared to Scheme 1 and Scheme 2. These observations confirm the effectiveness of the proposed joint RAN resource allocation framework.

As mentioned previously, the unique characteristics of MRU and SRUs are to meet

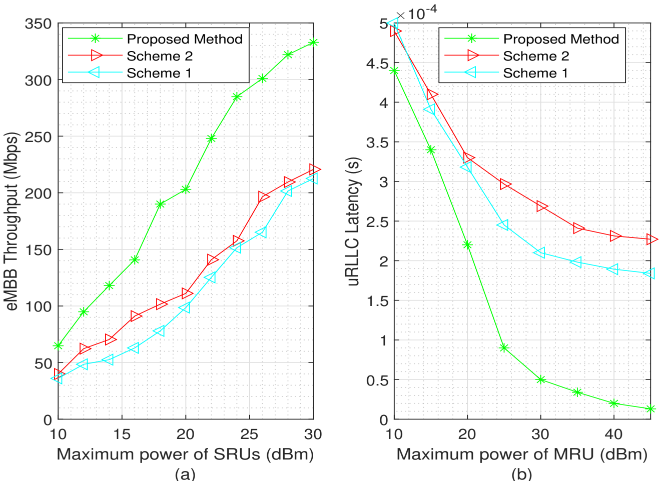


FIGURE 3.3: The sum eMBB data rate and worst-case uRLLC latency versus $P^{\max}$.

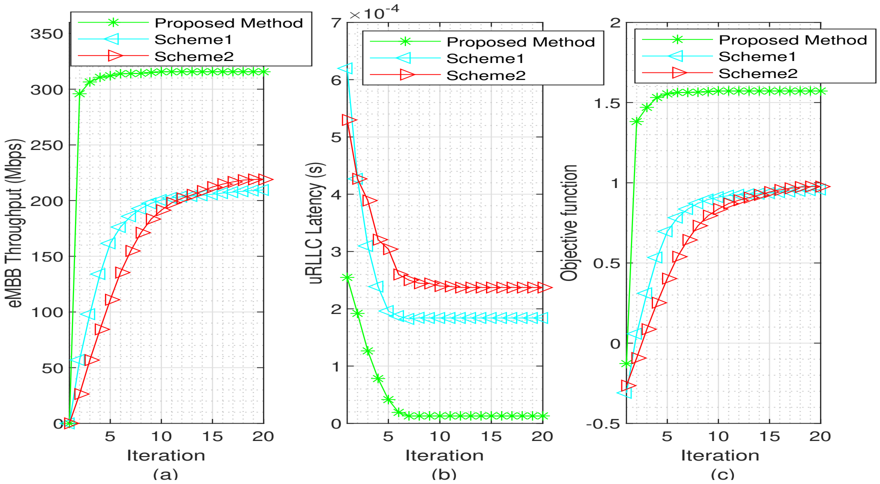


FIGURE 3.4: Convergence behavior.

uRLLC and eMBB demands, respectively. Therefore, we evaluate the impact of the maximum power budgets of these RUs on two services in Fig. 3.3. Fig. 3.3(a) plots the sum eMBB data rate as a function of the SRUs' transmit power, while Fig. 3.3(b) shows the uRLLC latency under the different maximum power of MRU. We can see in both scenarios that increasing the maximum power of RUs results in an improved sum eMBB data rate and reduced uRLLC latency, which reveal the impact of $P_m^{\max}$ on both objective functions. In addition, according to the performance gain given in Fig. 3.3, the proposed method offers the highest performance compared to two considered benchmark schemes. At $P_m^{\max} = 20$ dBm of SRUs ($i.e.$, $\forall m = 1, \cdots, M$), the proposed method achieves 98% and 67% performance gains in terms of the sum eMBB data rate, compared to Scheme 1 and Scheme 2, respectively. Similarly, for $P_0^{\max} = 30$ dBm, the uRLLC latency of the proposed method provides about three times less than Scheme 1 and Scheme 2, respectively.

Finally, we examine the convergence performance of the three considered schemes using Algorithm 1 in Fig. 3.4. It is shown that the proposed algorithm not only converges very fast, reaching the optimal value in less than 10 iterations, but also greatly outperforms two benchmark schemes.

3.5 Summary

We have presented a joint RAN resource allocation framework to realize eMBB and uRLLC coexisting in an OFDMA-based Open RAN system. We have proposed a comprehensive optimization problem under some practical constraints to maximize the sum eMBB data rate while minimizing the uRLLC latency. We have conducted an in-depth analytical e2e uRLLC latency. A new SCA-iterative algorithm has been developed to solve the formulated problem effectively. We have shown that the proposed method based on MC greatly improves resource utilization compared to benchmark schemes.

Chapter 4

Traffic Steering for eMBB and uRLLC Coexistence in Open Radio Access Networks – Part II: LSTM Traffic Prediction

Open radio access network (O-RAN) Alliance offers disaggregated radio access network (RAN) functionality built using open interface specifications between blocks. To efficiently support various competing services, *namely* enhanced mobile broadband (eMBB) and ultra-reliable and low-latency (uRLLC), the O-RAN Alliance has introduced a standard approach toward more virtualized, open, and intelligent networks. To realize the benefits of Open RAN in optimizing resource utilization, this chapter studies an intelligent traffic steering (TS) scheme within the proposed disaggregated Open RAN architecture. For this purpose, we propose a joint intelligent traffic prediction, flow-split distribution, dynamic user association, and radio resource management (JIFDR) framework in the presence of unknown dynamic traffic demands. To adapt to dynamic environments on different time scales, we decompose the formulated optimization problem into two long-term and short-term subproblems, where the optimality of the latter is strongly dependent on the optimal dynamic traffic demand. We then apply a long-short-term memory (LSTM) model to effectively solve the long-term subproblem, aiming to predict dynamic traffic demands, RAN slicing, and flow-split decisions. The resulting non-convex short-term subproblem is converted to a more computationally tractable form by exploiting successive convex approximations. Finally, simulation results are provided to demonstrate the effectiveness of the proposed algorithms compared to several well-known benchmark schemes.

4.1 Introduction

Next-generation (*NextG*) mobile communication networks (*e.g.*, beyond fifth-generation (5G) and sixth-generation (6G)) are designed to accommodate a wide range of service types with their own specific demands, such as data rate, reliability, and delay. The mentioned services are basically categorized into three principal cases: enhanced mobile broadband (eMBB), massive machine-type communications (mMTC), and ultra-reliability low-latency communication (uRLLC) [11]. Efficiently supporting the coexistence of these heterogeneous services is challenging in the *NextG* wireless networks due to their competing demands. The existing "one-size-fits-all" 5G architecture makes it very difficult, if not impossible, to enable the coexistence of heterogeneous services since the present 5G wireless networks are aggregated, closed, and inflexible. Despite the cost-effectiveness of centralized/cloud radio access networks (C-RAN) and virtual radio access networks (vRAN), open interfaces, non-proprietary hardware, and software are still lacking in these systems. Open radio access network (Open RAN) is an emerging solution to enable flexible, virtualized, disaggregated, intelligent, and open *NextG* wireless networks to support the heterogeneity of wireless services [66]. The openness of RAN components not only increases the interoperability between vendors but also speeds up the delivery of new services, which can be dynamically nominated to users. Due to the increasing complexity of *NextG* wireless networks, a self-organizing network's optimization, deployment, and operation are increasingly becoming impossible without intelligence [81], [82].

Accommodating heterogeneous services (uRLLC, eMBB, and mMTC) with competing demands on the identical RAN infrastructure is exceedingly challenging, such that building numerous physical networks to accommodate distinct services is not practical. Hence, it is difficult to efficiently route heterogeneous traffic to enhance user experience and network efficiency [69]. To this end, the concept of RAN slicing has been suggested as a potential remedy to constantly assign accessible storage, compute, and communication resources across multiple services whilst guaranteeing their isolation [70]. In this chapter, we concentrate on the RAN slicing mechanism's optimization, which entails the effective allocation of the physical radio resources such as transmit power and the time-frequency unit. Meeting the multi-traffic coexistence to handle nonuniform requirements is not possible only by allocating the transmit power and time-frequency unit. Traffic steering (TS), one of the most efficient approaches, enables network software to steer the traffic in the most proper paths by routing user traffic through the most suitable radio resources. Nevertheless, the available research on TS in 5G is still limited and uncompleted. While most existing works of literature have studied typical TS, which treats all users similarly, regardless of users' demands and network conditions, meaning that a network operator may even waste its resources if a simple strategy is implemented. To address this issue, this chapter proposes a novel TS based on the traffic demands to achieve multi-traffic coexistence. To enhance data rate and reliability in wireless networks with limited bandwidth, the multi-connectivity (MC) technique can be used to aggregate multiple links and allow a user to connect to more than two nodes. In practice, MC has the potential to dramatically reduce interference and latency of mobility methods, especially at the cell edge [83]. The multi-link capability makes MC the most practical method for achieving uRLLC and eMBB coexistence, whereas the recent proposals for the 5G air interface in 3GPP Release 15 utilize flexible mixed numerologies [84].

Another great challenge of 5G is achieving low latency in latency-critical applications. To meet this, 5G new radio (NR) defines a new concept of mini-slot, which consists of at most 4 orthogonal frequency division multiplexing (OFDM) symbols to support small packet transmission sizes. Significantly, this short slot duration reduces the transmission time. Furthermore, the single numerology that is used in fourth-generation (4G) long-term evolution (LTE) is not suitable for expected multiservices in 5G wireless networks. Hence, flexible mixed numerologies have been recently proposed for such wireless networks in 3GPP Release 15, which enhances flexibility. To this end, this chapter considers mixed numerologies in a frequency domain such that the assigned services to each slice can select a proper numerology to allocate its data transmission while guaranteeing each service's requirements. It should be mentioned that this new concept introduces new challenges related to RAN slicing that need to be studied. For instance, the dynamic allocation of the mixed numerology-based time-frequency units and transmit power is a vital challenge.

Inspired by [85] and [86], this chapter introduces a joint intelligent traffic steering and slice-isolation radio resource allocation framework for allocating the RAN resources with mixed numerologies, taking into account the Open RAN architectural requirements, various service requirements, and queue status. To present the role of intelligence in Open RAN architecture, this chapter benefits from the long short-term memory (LSTM) recurrent neural network (RNN) to learn the network traffic pattern and predict the unknown incoming traffic packets of the network. LSTM has been introduced as an undeniable state-of-the-art method within the deep neural networks to overcome the exploding/vanishing gradient problem, especially in learning long-term dependencies [87]. We outline the compliance of the overall scheme with the Open RAN requirements later.

4.1.1 Related Works

To improve services for network providers, the work in [88] focused on providing an efficient scheduling scheme to dynamically allocate radio resources in LTE networks. In [89], the authors proposed a joint resource allocation and dynamic link adaptation scheme for multiplexing eMBB and uRLLC on a shared channel, which dynamically tunes the block error probability of uRLLC small payload transmissions in each cell. A control channel and packet size-aware resource allocation approach was introduced in [90] to enable the packet scheduling and resource allocation for uRLLC and eMBB traffic coexistence in 5G NR networks. Although the heuristic algorithm proposed in [90] meets the uRLLC's requirements by preserving a large number of resources to uRLLC, this method has failed to isolate the slice, resulting in a reduction of the eMBB data rate compared to high uRLLC traffic. Wu *et al.* [91] developed the puncturing method to eliminate the uRLLC queuing delay for multiplexing of uRLLC and eMBB services. The authors in [92] studied a joint scheduling scheme to maximize the eMBB data rate while minimizing the utility of uRLLC to meet the quality of service (QoS) requirements. Since uRLLC services are prioritized in the puncturing-based schemes and scheduled on the assigned eMBB's resources, the eMBB performance (data rate and reliability) significantly decreases when the uRLLC traffic increases. Moreover, the fixed numerology over frequency-time resources for the scheduling scheme is often considered.

There is significant attention from academia and industry to TS in the literature. In [73], a TS framework was studied in unlicensed bands on the LTE network in order to distribute traffic among radio access technologies, heterogeneous cells, and spectrum bands. To overcome the puncturing difficulties in multiple services, Praveenkumar *et al.* in [93] proposed a slice-isolated RAN slicing scheme with orthogonal frequency-division multiple access (OFDMA) for the coexistence of uRLLC and eMBB. A joint scheduling and TS scheme based on dynamic MC and RAN slicing in 5G networks was analyzed in [29], in which an effective capacity model to evaluate the frameworks' performance is proposed. To integrate the LTE into 5G networks, Prasad*et al.* [39] investigated an energy-efficient RAN moderation and dynamic TS based on the connectivity by multiple radio links.

The RAN slicing framework over multiple services networks has been recently developed under frequency-time resources thanks to the flexibility of mixed-numerologies. The authors in [94] studied a resource allocation optimization problem by considering the flexible numerology in both frequency and time domains. The work in [12] analyzed the wireless scheduling optimization problem over the mixed-numerologies to support the heterogeneous services with different quality of service (QoS) requirements, assuming that mapping the radio resources (time and frequency) is decoupled from service scheduling. A joint optimization of RAN slicing, RB, and power allocation problem for eMBB, mMTC, and uRLLC in 5G wireless networks was considered in [95] under imperfect channel state information (CSI).

However, the aforementioned works have investigated TS with flexible numerology in the "one-size-fits-all" network architecture, which is not adaptable enough to support heterogeneous services. Despite the huge benefit of the intelligence of Open RAN, there are only a few attempts on the TS in the literature. Solmaz *et al.* in [68] proposed an intelligent traffic prediction and radio resource management framework to control the congested cell based on cell splitting in Open RAN architecture for multiplexing uRLLC and eMBB services. In [67], a systematic analysis for implementing the intelligence in each layer of Open RAN architecture for data-driven *NextG* wireless networks was provided by considering the closed-control loops between Open RAN components. Furthermore, in our previous study [85], we have proposed a TS scheme based on MC and RAN slicing technologies to effectively allocate diverse network resources in Open RAN architecture by assuming fixed-numerology (*i.e.*, 0.25ms mini-slots) tailored with 5G NR. However, the works mentioned earlier have not designed a traffic steering and RAN resource slicing scheme for heterogeneous traffic applicable to the beyond 5G wireless networks on Open RAN architecture. The majority of works have investigated the resource allocation scheme for various services with fixed numerology in the monolithic and inflexible architecture of 4G LTE networks. Hence, this is the first attempt to investigate the performance efficiency of mixed numerologies considering the RAN slicing, MC, and mini-slot to achieve multi-traffic coexistence in Open RAN architecture while guaranteeing the users' QoS requirements.

4.1.2 Contributions

In this chapter, we develop an intelligent TS framework in the presence of unknown dynamic traffic demand to meet the requirements of both uRLLC and eMBB services in beyond 5G networks based on dynamic MC. Learning an optimal traffic steering policy in dynamic environments is challenging because fluctuations in traffic demand over time are non-stationary and unknown, hindering the computation of cost-efficient associations. This chapter proposes an intelligent framework by locating rAPPs and xAPP at the non-real-time (non-RT) RAN intelligent controller (RIC) and near-real-time (near-RT) RIC of the Open RAN architecture. The existing rAPPs at non-RT RIC include the traffic prediction, dynamic RAN slicing decision, and flow-split distribution, while the xAPP at near-RT RIC is radio resource management to schedule the joint resource block (RB) and transmission power with mixed numerologies based on standardization in 5G NR. To the best of our knowledge, this is the first work to model intelligent TS in Open RAN architecture and study TS in-depth detail in Open RAN layers considering the mixed-numerology in the presence of unknown traffic demands.

To achieve the maximum data rate for eMBB traffic while guaranteeing the minimum uRLLC latency requirement and vice versa, we propose a joint intelligent traffic prediction, flow-split distribution, dynamic user association, and radio resource management scheme befitting the Open RAN architecture. Then, we identify the location of the ML training, AI server, and inference modules to provide a high-level architecture of deployment scenarios and end-to-end (e2e) workflow to prove compatibility with Open RAN standards. Our main contributions are summarized as follows:

- We develop a general optimization framework to jointly optimize intelligent traffic prediction, flow-split distribution, dynamic user association, and radio resource management, called “JIFDR”. To maximize the eMBB's data rate while guaranteeing the uRLLC latency requirement, or vice versa, we formulate two optimization problems with different objective designs while satisfying QoS requirements, slice isolation, power budget, and maximum fronthaul (FH) capacity.

- To effectively solve the formulated problems, we divide each problem into long-term and short-term subproblems, which are executed on different time scales. The long-term subproblem is mapped into three dependent rAPPs: traffic prediction, dynamic RAN slicing decision, and flow-split distribution at the non-RT RIC. In contrast, the short-term sub-problem is deployed as the radio resource management xAPP at the near-RT RIC, which is linked to the upper layer through the A1 interface.

- The long-term subproblem benefits from the LSTM RNN to learn and predict traffic patterns and demands. This model is trained offline at the non-RT RIC in the service management and orchestration (SMO) through the long-term collected data from the RAN layer via the O1 interface. RNN is utilized to learn the temporal pattern of the traffic demand from current values in order to forecast future values. Upon the inference result, two heuristic methods are proposed to optimize the RAN slicing and flow-split distribution.

- Next, given rAPPs' outcomes sent from the non-RT RIC via the A1 interface, we propose a successive convex approximation (SCA)-based iterative algorithm to solve the short-term subproblem, which belongs to a class of mixed-integer non-convex programming (MINCP) problems.

- Finally, numerical results are presented to demonstrate the proposed algorithm's quick convergence behavior and to confirm its efficacy in comparison to benchmark schemes.

Furthermore, by using a mathematical analysis, convergence and complexity analysis are studied. The average mean square error (MSE) of the prediction is relatively low at 0.0033.

The rest of this chapter is organized as follows: Section 4.2 introduces the Open RAN architecture and system model. In Section 4.3, we present the problem formulation and overall intelligent TS deployment architecture and algorithm. Section 4.4 first proposes the LSTM model and heuristic methods to solve the long-term subproblem and then develops an SCA-based iterative algorithm to solve the short-term subproblem. Simulation results and discussions are provided in Section 4.5, while Section 4.6 concludes the paper.

4.2 Open RAN Architecture and System Model

4.2.1 Open RAN Architecture

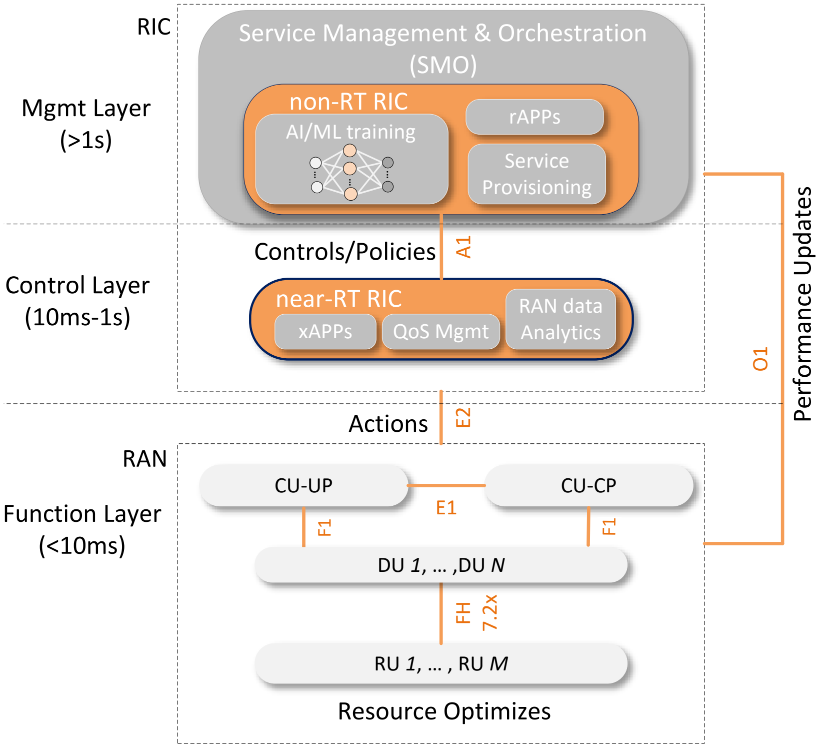


FIGURE 4.1: Open RAN architecture based on O-RAN Alliance

The Open RAN architecture based on the O-RAN Alliance is illustrated in Fig. 4.1, including three main layers (the management, control, and function layers) [96]. To further reduce the RAN expenditure, Open RAN fosters self-organizing networks by adding two unique modules of near-RT and non-RT RICs to enable a centralized network abstraction, which improves efficiency by cost-reducing the human-machine interaction. Following the disaggregation concept, BS functionalities are virtualized as network functions based on the 3GPP functional split and are distributed among various network nodes, *namely* central unit (CU), distributed unit (DU), and radio unit (RU) [68]. Hence, open interfaces (FH, A1, O1, E2, F1) are introduced to enable efficient multi-vendor interoperability, where a network operator can select RAN components from different vendors individually.

The unique feature of RICs is to create closed-control loops (*i.e.*, autonomous action and feedback loops) between RAN components and their controllers. In order to control traffic prediction, network slicing, and hand-over management, Open RAN defines three control loops, *namely* non-RT, near-RT, and RT running at different timescales ranging from 1 ms to thousands of ms, enabling real-time control of transmission methods and beamforming. Following Open RAN Alliance specifications, each loop that works on a timescale of at least one second is called a non-RT control loop, which involves the coordination between both RICs through the A1 interface. The near-RT control loop operates on a timescale between 10 ms and 1 s while it runs between the near-RT RIC and DU and CU components. The third loop working on sub-10 ms is labeled as the RT control loop, which is largely relevant to interactions between elements of DU and cell site.

In particular, the non-RT RIC carries out tasks with a temporal granularity greater than one second, like service provisioning and training AI/ML models, which rAPPs can be implemented in this controller. On the other hand, the near-RT RIC manages operations with timescales of more than 10 ms, hosts external applications (referred to as xApps), and incorporates intelligence in the RAN by data-driven control loops. RICs may execute applications created by independent third-party specialized software suppliers as a platform for hosting software. These applications are known as “rAPPs” and ”xAPPs” and act as key enablers to run on non-RT and near-RT RICs, respectively. rAPPs handle the non-RT functions that require more than 1 second to be executed, which may take minutes or hours. While xAPPs are external applications specific to handle radio functions that run between 10 ms and 1 s that interact with RAN elements and the upper layer by open interfaces to reconfigure some exposed functionality. To this end, Open RAN Alliance strives to steer the industry toward the development of AI/ML-enabled RICs.

4.2.2 Network Model

We consider a downlink OFDMA multi-user multiple-input single-output (MU-MISO) system in the Open RAN architecture, consisting of one CU, the set $\mathscr{N} \triangleq \{1, 2, \ldots, N\}$ of N DUs, and the set $\mathscr{M} \triangleq \{1, 2, \ldots, M\}$ of M RUs. For cost-effective deployment, each DU serves a cluster of RUs. Let denote by $\mathscr{M}_n \triangleq \{(n, 1), \ldots, (n, M_n)\}$ with $|\mathscr{M}_n| = M_n$ and $\sum_{n\in\mathscr{N}} M_n = M$ the set of RUs served by DU n. The m-th RU served by n-th DU is referred to as RU(n, m), which is equipped with K antennas while users are equipped with a single antenna. Let us denote by $\mathscr{U} \triangleq \{1, \ldots, U\}$ the set of users served by DUs, which can be further divided into two disjoint sets $\mathscr{U}^{\mathsf{ur}}$ of U^{ur} uRLLC users and $\mathscr{U}^{\mathsf{em}}$ of U^{em} eMBB users. The eMBB users generate the traffic with a large packet of size Z^{em} bytes, while the uRLLC users generate a sequence of small and identical packets of Z^{ur} bytes. In addition, as shown in Fig. 4.2, we assume that all data arriving from upper layers is stored in the user-specific

Table 4.1: Summary of Main Notations and Variables

Variable	**Meaning**
$\alpha[t]$	Bandwidth-split variable per frame t
$\varphi_{m,u}[t]$	Portion of data flow routed to user u via RU m per frame t
$\pi^{\mathsf{x}}_{m,u,f_i}[t_s]$	RB (t_s, f_i) allocated to user u (uRLLC or eMBB) via RU m
$p^{\mathsf{x}}_{m,u,f_i}[t_s]$	Transmit power from RU m to user u (uRLLC or eMBB) via RB (t_s, f_i)
$\lambda_u[t]$	Traffic demand of user u per frame t
Notation	**Meaning**
$\mathscr{N}$, $\mathscr{M}$, $\mathscr{M}_n$	Sets of DUs, RUs, and set of RUs covered by DU n
$\mathscr{U}$, $\mathscr{U}^{\mathsf{ur}}\mathscr{U}^{\mathsf{em}}$	Sets of all users, uRLLC users, and eMBB users, respectively
B, β_i, B_i	Total system BW, RB's BW, and BWP assigned to numerology i
Δ, δ_i	Frame's duration and TTI's duration
S_i, F_i	Numbers of TTI per frame and subcarriers per TTI
$a_u[t]$	Flow-split selection vector for user u
$\boldsymbol{h}_{m,u,f_i}[t_s]$, $g_{m,u,f_i}[t_s]$	Channel vector and effective channel gain between RU m and user u
$\bar{\boldsymbol{h}}_{m,u,f_i}[t]$, $\tilde{\boldsymbol{h}}_{m,u,f_i}[t_s]$	Line-of-sight (LoS) and non-LoS (NLoS) components
$\varrho_{m,u,f_i}[t]$, $\zeta_{m,u,f_i}[t]$	Rician factor and large-scale fading
N_0, N_1	Power of the AWGN
V, P_e, Q^{-1}	Channel dispersion, error probability, and inverse of Q-function
Γ, Γ_0	The received SNR, and minimum received SNR
λ^{max}, Λ	Maximum finite arrival traffic and the total of all traffic demands
Q^{max}, $q_{m,u}$	Maximum queue buffer capacity and queue-length of user u at buffer of RU m
$\mu_{cu/du}$, $f_{cu/du}$, C	Task rate, computation capacities of CU and DU and number of cycles
$\tau^{\mathsf{ur}}_u, \tau^{\mathsf{pro}}_{cu/du}$	The e2e latency of uRLLC user u and processing latency of all users at CU/DU
$\tau^{\mathsf{tx}}_{cu,du}$, $\tau^{\mathsf{tx}}_{du,ru}$, $\tau^{\mathsf{tx}}_{ru,u}$	Transmission latency under MH and FH links and from RU m to user u
D^{ur}, C^{MH}, C^{FH}	Latency requirement of uRLLC traffic and maximum MH and FH capacity

transmission buffers of the RUs till it is time to serve it. The RUs serve the users in the cell by allocating the frequency-time radio RBs and transmission power to each RB. The parameters used in this chapter are summarized in Table 4.1.

To meet the demands of exigent latency services, we investigate a mini-slot-based framework where each time slot is broken into two mini-slots. Each mini-slot has a duration of $\delta = 1/2^{\gamma+1}$ ms and comprises 7 OFDM symbols, where $\gamma \in \{0, 1, 2\}$ is the subcarrier spacing (SCS) index. Hereon, we suppose that several RUs operating in MC configuration are simultaneously providing eMBB and uRLLC services. Following [97], numerology with index $i = 1$ (*i.e.*, SCS index $\gamma = 1$) is appropriate for eMBB to meet the requirement of high data rate, while numerology with index $i = 2$ (*i.e.*, SCS index $\gamma = 2$) is more suitable for the uRLLC service's applications with the latency-critical and small data packet of uRLLC. From the mixed-numerologies point of view, eMBB service sorts the numerology $i = 1$ with RB's bandwidth (BW) of $\beta_i|_{i=1} = 360$ kHz and $\delta_i|_{i=1} = 0.25$ ms of transmission time interval (TTI) duration as the highest priority, and uRLLC service would prioritize numerology $i = 2$ with RB's BW of $\beta_i|_{i=2} = 720$ kHz and $\delta_i|_{i=2} = 0.125$ ms of TTI duration.

The multiplexing of mixed numerologies in the frequency domain is considered in this chapter, where the carrier BW that is accessible for the downlink transmissions is divided into several bandwidth parts (BWPs). According to this, each user is able to alter its RF bandwidth based on its required data rate by switching between numerous BWPs.

As illustrated in Fig. 4.3, the desirable BWP design to serve two types of services with different requirements is established based on the expected queue length of each service by introducing the BW-split variable $\alpha[t] \in [0, 1]$. Whereas this method does not call for tight time synchronization techniques, using various numerologies in the adjacent sub-bands causes inter-numerology interference (INI). Hence, to reduce INI, a fixed guard band B_G equal to one RB's BW (*i.e.*, 180 kHz) is configured between the two neighbor numerologies (*i.e.*, sub-bands). The scheduled BWP assigned to the uRLLC slice with numerology $i = 2$ is denoted by $B_i[t]|_{i=2} = \alpha[t]B$, to unload the existing packets in the uRLLC slice's queues at frame t, where B is the total carrier BW. In contrast, $B_i[t]|_{i=1} = (1 - \alpha[t])B - B_G$ the scheduled BWP assigned to eMBB slice with numerology $i = 1$.

Assume the proposed system model works in a discrete time-frame indexed by $t \in [1, 2, \ldots, T]$, which corresponds to one large-scale coherence time of $\Delta = 10$ ms duration for each frame, as shown in Fig. 4.3. Depending on the selected numerology i by each service, each frame in the time domain is divided into S_i TTIs where the duration of each TTI denoted by $t_s = (t - 1)S_i + s$ with $s = \{1, \ldots, S_i\}$ is δ_i. Thus, based on the selected numerology i, each BWP is partitioned into F_i number of sub-bands of frequency set $\mathscr{F}_i = \{1, \ldots, f_i, \ldots, F_i\}$ in the frequency domain and S_i number of TTIs in each frame, indexed by $t_s = \{(t - 1)S_i + 1, \ldots, (t - 1)S_i + s, \ldots, (t - 1)S_i + S_i\}$ in the time domain. Such that $F_i[t] = \lfloor B_i[t]/\beta_i \rfloor$ and $S_i = \Delta/\delta_i$. Therefore, a total $F_i[t] \times S_i$ number of RBs are

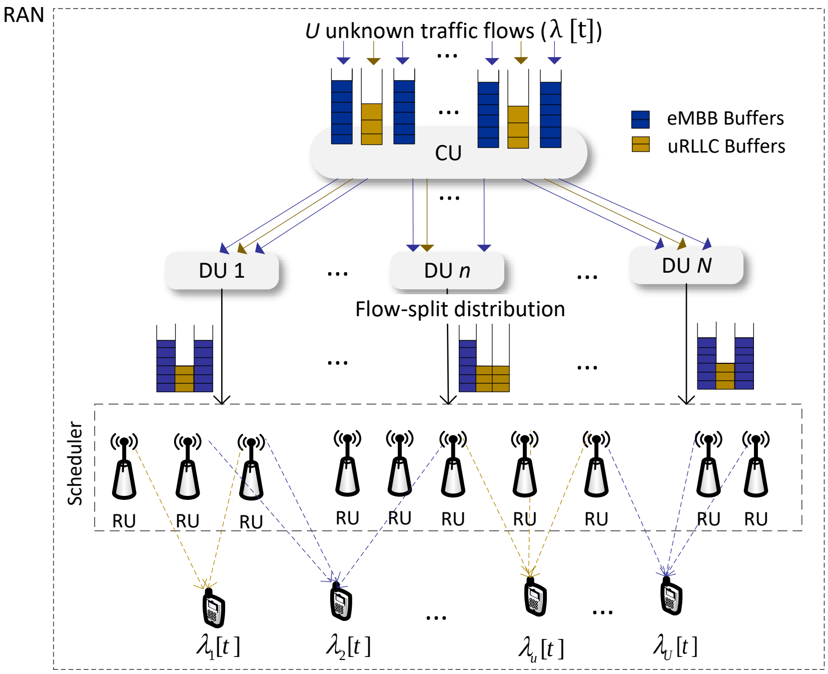


FIGURE 4.2: System model with the traffic-steering scheme.

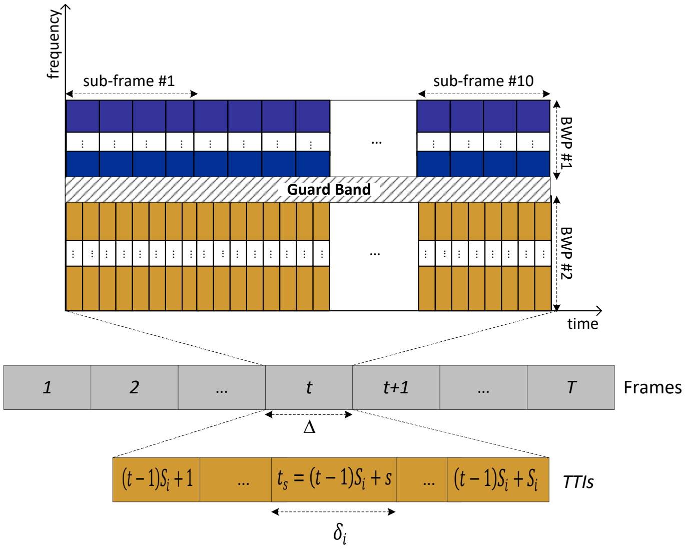


Figure 4.3: Time-frequency grid with different numerologies.

accessible for the services using the i-th numerology at each frame t via each RU.

As depicted in Fig. 4.2, the U independent data traffics with different demands at the CU layer are subsequently routed to virtual network functions (VNFs) in the DUs layer for parallel processing, referred to as data flows. We adopt the $M/M/1$ processing queue model on a first-come, first-serve basis to serve each user's packets. As it is clear from Fig. 4.2, the maximum number of paths for each user is M. According to the principle of the TS technique, the CU splits the data flow of the u-th user into several sub-flows, which are possibly transmitted via the maximum of M paths and then aggregated at this user. Because of the non-overlapped DUs' coverage, the resource optimization design at one DU is similar to that of other DUs. Thus, for ease of presentation, we drop the subscript index of DUs hereafter. To this end, we define $\boldsymbol{a}_u[t] \triangleq \big[a_{m,u}[t]\big]$ as the flow-split selection vector for the u-th data flow in time-frame t. In particular, if $a_{m,u}[t]=1$, the m-th RU is selected to transmit data of u-th data flow; otherwise, $a_{m,u}[t]=0$. In addition, let us denote by $\boldsymbol{\varphi}[t] \triangleq \{\boldsymbol{\varphi}_u[t],\ \forall u | \sum_m \varphi_{m,u}[t]=1, \varphi_{m,u}[t]\in[0,1]\}$ the global flow-split decision, in which $\boldsymbol{\varphi}_u[t] \triangleq \Big[\varphi_{m,u}[t]\Big]^T$ represents the flow-split portion vector of user u while $\sum_m \varphi_{m,u}[t]=1$, where $\varphi_{m,u}[t]\in[0,1]$ indicates a portion of data flow routed to user u via RU m in time t by selecting action $a_{m,u}[t]$.

Achievable data rate: The channel vector between RU m and the u-th user at the sub-band f_i in TTI t_s is denoted by $\boldsymbol{h}_{m,u,f_i}[t_s]\in\mathbb{C}^{K\times 1}$, which follows the Rician fading model with the Rician factor $\varrho_{m,u,f_i}[t]$. Within each frame, we assume that the channel remains temporally invariant, while it may be different across each short-time scale TTI. We model $\boldsymbol{h}_{m,u,f_i}[t_s]$ as

$$\boldsymbol{h}_{m,u,f_i}[t_s] = \sqrt{\zeta_{m,u,f_i}[t]}\big(\sqrt{\frac{\varrho_{m,u,f_i}[t]}{(\varrho_{m,u,f_i}[t]+1)}}\bar{\boldsymbol{h}}_{m,u,f_i}[t] + \sqrt{\frac{1}{(\varrho_{m,u,f_i}[t]+1)}}\tilde{\boldsymbol{h}}_{m,u,f_i}[t_s]\big), \quad (4.1)$$

where $\zeta_{m,u,f_i}[t]$ is the large-scale fading; $\bar{\boldsymbol{h}}_{m,u,f_i}[t]$ and $\tilde{\boldsymbol{h}}_{m,u,f_i}[t_s]$ are the line-of-sight (LoS) and non-LoS (NLoS) components, which follow a deterministic channel and Rayleigh fading model, respectively. Given the orthogonality constraint, this chapter considers that each RB of a RU is assigned to only one single user during one TTI, such as $\pi^{\mathsf{em}}_{m,u,f_i}[t_s] \in \{0,1\}$ and $\pi^{\mathsf{ur}}_{m,u,f_i}[t_s] \in \{0,1\}$ for eMBB and uRLLC traffics, respectively. Here, $\pi^{\mathsf{em}}_{m,u,f_i}[t_s] = 1$ if the $\mathrm{RB}(t_s, f_i)$ associated with sub-band f_i in TTI t_s of RU m assigned to the u-th eMBB user, and $\pi^{\mathsf{em}}_{m,u,f_i}[t_s] = 0$, otherwise; a similar definition is given for uRLLC users. Let define $\Pi[t_s] \triangleq \{\pi^{\mathsf{em}}_{m,u,f_i}[t_s], \pi^{\mathsf{ur}}_{m,u,f_i}[t_s] \in \{0,1\} | \sum_{m,u} \pi^{\mathsf{em}}_{m,u,f_i}[t_s] \leq 1; \forall f_i, t_s, i = 1, \sum_{m,u} \pi^{\mathsf{ur}}_{m,u,f_i}[t_s] \leq 1; \forall f_i, t_s, i = 2\}$ as the RB allocation constraint. This is to ensure the orthogonality constraint and QoS constraint for uRLLC service.

Thanks to the MC technique, the main interference of eMBB is eliminated, and the rest of the interference can be supposed as noise, which is also constant [98]. Hence, the instantaneous achievable rate in [bits/s] for a given set of channel realizations at the u-th eMBB user at TTI t_s is given by

$$R^{\mathsf{em}}_{m,u}(\boldsymbol{p}^{\mathsf{em}}[t_s]) = \sum_{f_i=1}^{F_i} \beta_i \log_2(1 + \frac{p^{\mathsf{em}}_{m,u,f_i}[t_s] g_{m,u,f_i}[t_s]}{N_0}), \quad (4.2)$$

where β_i, N_0 and $p^{\mathsf{em}}_{m,u,f_i}[t_s]$ are the bandwidth of each RB in numerology index i, power of the additive white Gaussian noise (AWGN), and transmit power from RU m to user u for eMBB traffic at sub-band f_i at the TTI t_s, respectively; $g_{m,u,f_i}[t_s]$ denotes the effective channel gain, given as $g_{m,u,f_i}[t_s] \triangleq \|\boldsymbol{h}_{m,u,f_i}[t_s]\|^2_2$. Let us define $\boldsymbol{p}^{\mathsf{em}}[t_s] \triangleq [p^{\mathsf{em}}_{m,u,f_i}[t_s]]$, $\forall f_i, u, m$. The transmit power must satisfy $p^{\mathsf{em}}_{m,u,f_i}[t_s] \leq \pi^{\mathsf{em}}_{m,u,f_i}[t_s] P^{\mathsf{max}}_m$ with P^{max}_m being the power budget at RU m, which guarantees that RU m allocates power to user u on $\mathrm{RB}(t_s, f_i)$ only if $\pi^{\mathsf{em}}_{m,u,f_i}[t_s] = 1$; otherwise $\pi^{\mathsf{em}}_{m,u,f_i}[t_s] = 0$ and $p^{\mathsf{em}}_{m,u,f_i}[t_s] = 0$. As a result, the data rate of eMBB user $u \in \mathscr{U}^{\mathsf{em}}$ in TTI t_s is given as $R^{\mathsf{em}}_u(\boldsymbol{p}^{\mathsf{em}}[t_s]) = \sum_m R^{\mathsf{em}}_{m,u}(\boldsymbol{p}^{\mathsf{em}}[t_s])$. The minimum QoS requirement for eMBB users is guaranteed by the constraint $R^{\mathsf{em}}_u(\boldsymbol{p}^{\mathsf{em}}[t_s]) \geq R^{\mathsf{th}}$, where R^{th} is a given QoS threshold.

In contrast, owing to the finite block-length in uRLLC traffics, the instantaneous achievable rate of u-th uRLLC user ($u \in \mathscr{U}^{\mathsf{ur}}$) from RU m in TTI t_s using the short block-length can be expressed as [76]

$$R^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s]) = \sum_{f_i=1}^{F_i} \beta_i \Big[\log_2\Big(1+\frac{p^{\mathsf{ur}}_{m,u,f_i}[t_s] g_{m,u,f_i}[t_s]}{N_0}\Big) - \log_2(e) \frac{\pi^{\mathsf{ur}}_{m,u,f_i}[t_s]\sqrt{V} Q^{-1}(P_e)}{\sqrt{\delta_i \beta_i}}\Big], \quad (4.3)$$

where V, P_e and Q^{-1}: $\{0,1\} \rightarrow \mathbb{R}$ denote the channel dispersion, error probability, and inverse of the Gaussian Q-function, respectively. Let us define $\boldsymbol{p}^{\mathsf{ur}}[t_s] \triangleq [p^{\mathsf{ur}}_{m,u,f_i}[t_s]]$ and

$\boldsymbol{\pi}^{\mathsf{ur}}[t_s] \triangleq [\pi^{\mathsf{ur}}_{m,u,f_i}[t_s]],\ \forall f_i, u, m$. It is observed that $V = 1 - \frac{1}{(\Gamma[t_s])^2} \approx 1$ when the received $\Gamma[t_s] = \frac{p^{\mathsf{ur}}_{m,u,f_i}[t_s] g_{m,u,f_i}[t_s]}{N_0} \geq \Gamma_0$ with $\Gamma_0 \geq 5$ dB. This can be easily achieved in cellular networks by arranging the uRLLC decoding vector into one possible null space of the reference subspace; the scheduler can eliminate inter-user interference of uRLLC [77]. Hence, we consider the constraint $\frac{N_0\Gamma_0}{g_{m,u,f_i}[t_s]}\pi^{\mathsf{ur}}_{m,u,f_i}[t_s] \leq p^{\mathsf{ur}}_{m,u,f_i}[t_s] \leq \pi^{\mathsf{ur}}_{m,u,f_i}[t_s]P^{\mathsf{max}}_m$ to guarantee the approximation $V \approx 1$ as well as the big-M formulation theory to avoid non-convexity of (4.2). Similar to the eMBB service, the data rate of uRLLC user $u \in \mathscr{U}^{\mathsf{ur}}$ in TTI t_s is given as $R^{\mathsf{ur}}_u(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s]) = \sum_m R^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s])$. We have the following power constraint,

$$\mathscr{P}[t_s] = \Big\{0 \leq p^{\mathsf{em}}_{m,u,f_i}[t_s] \leq \pi^{\mathsf{em}}_{m,u,f_i}[t_s]P^{\mathsf{max}}_m, \frac{N_0\Gamma_0\pi^{\mathsf{ur}}_{m,u,f_i}[t_s]}{g_{m,u,f_i}[t_s]} \leq p^{\mathsf{ur}}_{m,u,f_i}[t_s] \leq \pi^{\mathsf{ur}}_{m,u,f_i}[t_s]P^{\mathsf{max}}_m | \\ \sum_i \sum_{f_i,u} (p^{\mathsf{em}}_{m,u,f_i}[t_s] + p^{\mathsf{ur}}_{m,u,f_i}[t_s]) \leq P^{\mathsf{max}}_m\Big\}. \quad (4.4)$$

We denote $\lambda_u[t]$ in [packets/s] as the unknown traffic demand of user u in time-frame t with the length of Z^{x} bytes with $\mathsf{x} \in \{\mathsf{ur}, \mathsf{em}\}$, which is i.i.d. over time and upper bound by a finite constant λ^{max}, such as $\lambda_u[t] \leq \lambda^{\mathsf{max}} \leq \infty$. We consider that the retained independent queue at each RU for the u-th user, which is denoted by $\{\varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{x}}\}$ as the arrival processes of sub-flows, is controlled by a congestion scheduler. Thus, the queue-length of data flow u at RU m in TTI (t_{s+1}) is $q_{m,u}[t_s] = \max\{\big[q_{m,u}[t_{s-1}] + \varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{x}}\Delta - R^{\mathsf{x}}_{m,u}[t_s]\delta_i\big], 0\}$. In order to avoid the packet loss due to buffer overflow in each RU, the constraint $\sum_u q_{m,u}[t_s] \leq Q^{\mathsf{max}}, \forall m$ is imposed to ensure that the available packets in the buffer of RU shouldn't exceed the maximum queue-length of Q^{max} for each RU. Let $\boldsymbol{q}[t_s] \triangleq \big[q_{m,u}[t_s]\big]^T,\ \forall m, u$.

The e2e Traffic Latency for uRLLC: Denote by f_{cu} and f_{du} the computation capacities of CU and DU [cycles/sec], respectively. Considering the identical packet size, the required computation resource to process one packet of size Z is C (number of cycles). As a result, $\mu_{cu} = f_{cu}/C$ and $\mu_{du} = f_{du}/C$ are the task rates [1/sec] at CU and DU, respectively. As a result, $1/\mu_{cu}$ and $1/\mu_{du}$ represent the mean service times of the CU and DU layers, respectively. The processing latency of all data flows at the CU layer (τ^{pro}_{cu}) and DU layer (τ^{pro}_{du}) is computed as

$$\tau^{\mathsf{pro}}_{cu}[t] = \frac{\Lambda[t]}{\mu_{cu}},\ \text{and}\ \tau^{\mathsf{pro}}_{du}[t] = \frac{\Lambda[t]}{\mu_{du}},\ \forall n \in \mathscr{N}, \quad (4.5)$$

where $\Lambda[t] = \sum_u \lambda_u[t]$. Next, the arrival packets $\lambda_u[t]$ for the u-th user are transported to the DU layer via the midhaul (MH) link with the maximum capacity C^{MH} [bits/sec] between CU and DU. By Burke's theorem, the mean arrival data rate of the second layer, which is processed in the first layer, is still the same rate [78]. Hence, the data transmission latency of the traffic flow for user u under the MH limited capacity is

$$\tau^{\mathsf{tx}}_{cu,du}[t] = \frac{\Lambda[t]Z}{C^{\mathsf{MH}}}. \quad (4.6)$$

As mentioned previously, the maximum number of paths from DU n to each user is M_n. Since the packets for user u can be transmitted by multiple RUs, the effective response time $\tau^{\mathsf{tx}}_{du,ru}$ to transport all packets in the DUs layer should be computed by the worst average response time among its connected FH links with maximum capacity C^{FH}_m [bits/sec], *i.e.*,

$$\tau^{\mathsf{tx}}_{du,ru}[t] = \max_m \Big\{ \frac{\sum_{u\in\mathcal{U}^{\mathsf{ur}}} \varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{ur}}}{C^{\mathsf{FH}}_m} \Big\},\ \forall m \in \mathcal{M}_n. \tag{4.7}$$

The transmission latency from RU m to user u is then calculated as

$$\tau^{\mathsf{tx}}_{ru,u}[t_s] = \max_m \Big\{ \frac{\varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{ur}}}{R^{\mathsf{ur}}_{m,u}[t_s]} \Big\},\ \forall u \in \mathcal{U}^{\mathsf{ur}}. \tag{4.8}$$

Simply put, the e2e latency of each uRLLC user $u \in \mathcal{U}^{\mathsf{ur}}$ per each TTI is computed as

$$\tau^{\mathsf{ur}}_u[t] = \tau^{\mathsf{pro}}_{cu}[t] + \tau^{\mathsf{tx}}_{cu,du}[t] + \tau^{\mathsf{pro}}_{du}[t] + \tau^{\mathsf{tx}}_{du,ru}[t] + \sum_{t_s} \big(\tau^{\mathsf{tx}}_{ru,u}[t_s] + \tau^{\mathsf{pro}}_{ru}[t_s]\big),\ \forall u \in \mathcal{U}^{\mathsf{ur}}, \tag{4.9}$$

where τ^{pro}_{ru} is the process latency at RU m, which is bounded by three OFDM symbols duration that is typically very small. To ensure a minimum latency requirement for uRLLC user u, the e2e latency is bound by a predetermined threshold D^{ur}, *i.e.*, $\tau^{\mathsf{ur}}_u[t] \leq D^{\mathsf{ur}}$.

4.3 Problem Formulation and Overall Intelligent Traffic Steering Algorithm

4.3.1 Problem Formulation

Utility function: The ultimate goal is to optimize the joint intelligent traffic prediction, flow-split distribution, dynamic user association, and radio resource management in the presence of unknown dynamic traffic demand to serve eMBB and uRLLC users, subject to various resources constraints and diverse QoS requirements. Due to the conflict of objective functions in both services (*i.e.*, eMBB and uRLLC), the utility function should capture the eMBB data rate and worst-user e2e uRLLC latency separately such as $\mathcal{R}^{\mathsf{em}} = \sum_{u\in\mathcal{U}^{\mathsf{em}}} R^{\mathsf{em}}_u(\boldsymbol{p}^{\mathsf{em}}[t_s])$ and $\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\tau^{\mathsf{ur}}_u\}$ on two independent optimization problems. Based on the above definitions and discussions, the JIFDR problem is mathematically formulated as two independent optimization problems with common constraints as follows

$$
\begin{aligned}
\text{P1}: \max_{\boldsymbol{\lambda},\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p},\alpha} \quad & \mathcal{R}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) && \text{(4.10a)}\\
\mathsf{s.t.} \quad & \boldsymbol{\pi}[t_s] \in \Pi[t_s],\ \forall t_s && \text{(4.10b)}\\
& \boldsymbol{p}[t_s] \in \mathscr{P}[t_s],\ \forall t_s && \text{(4.10c)}\\
& \boldsymbol{\varphi}_u[t] \in \boldsymbol{\varphi}[t],\ \forall t, u \in \mathscr{U} && \text{(4.10d)}\\
& R_u^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) \geq R^{\mathsf{th}},\ \forall u \in \mathscr{U}^{\mathsf{em}} && \text{(4.10e)}\\
& \sum_u \left[R_{m,u}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) + R_{m,u}^{\mathsf{ur}}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s])\right] \leq C_m^{\mathsf{FH}}, \forall m \in \mathscr{M}_n && \text{(4.10f)}\\
& R_{m,u}^{\mathsf{ur}}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s]) \geq \varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{ur}}, \forall m \in \mathscr{M}_n, u \in \mathscr{U}^{\mathsf{ur}} && \text{(4.10g)}\\
& \tau_u^{\mathsf{ur}}(\boldsymbol{\lambda}[t], \boldsymbol{\varphi}[t], \boldsymbol{\pi}[t_s], \boldsymbol{p}[t_s]) \leq D^{\mathsf{ur}},\ \forall u \in \mathscr{U}^{\mathsf{ur}} && \text{(4.10h)}\\
& \sum_u q_{m,u}[t_s] \leq Q^{\mathsf{max}},\ \forall t_s, m \in \mathscr{M}_n && \text{(4.10i)}\\
& \sum_{f_i=1}^{F_i} \beta_i \leq B_i[t],\ i \in \{1,2\} && \text{(4.10j)}\\
& 0 \leq \alpha[t] \leq 1 && \text{(4.10k)}
\end{aligned}
$$

and,

$$
\begin{aligned}
\text{P2}: \min_{\boldsymbol{\lambda},\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p},\alpha} \quad & \max_{u \in \mathscr{U}^{\mathsf{ur}}} \{\tau_u^{\mathsf{ur}}\} && \text{(4.11a)}\\
\mathsf{s.t.} \quad & \text{(4.10b)} - \text{(4.10k)}, && \text{(4.11b)}
\end{aligned}
$$

where $\boldsymbol{\varphi}[t], \boldsymbol{\pi}[t_s]$ and $\boldsymbol{p}[t_s]$ are the vectors encompassing the flow-split portions, sub-band assignments, and power allocation variables at frame t and TTI t_s, respectively. Recall that, for each BWP with the given numerology, $B_i[t]|_{i=2} = \alpha[t]B$ and $B_i[t]|_{i=1} = (1-\alpha[t])B - B_G$. Constraint (4.10f) expresses the limited capacity of the FH link between DU n and RU m. Constraint (4.10g) ensures that each RB assigned to the u-th uRLLC user should transmit a complete data packet with the size Z^{ur}.

Challenges of solving JIFDR problem: The main challenges in solving problems (P1) and (P2) lie in the non-convexity of τ_u^{ur} and constraints (4.10f), (4.10g) and (4.10i) with respect to flow-split portions and transmit power variables. Furthermore, the binary nature of sub-band allocation variables in constraint (4.10b) makes these problems more difficult to solve directly, which is generally MINCP. Once may employ the MINCP solvers (*e.g.*, Gurobi) to directly solve binary $\boldsymbol{\pi}$. However, we argue that the exponential computation complexity of such a MINCP formulation limits its practical feasibility, especially when the number of variables exceeds a few thousand in large-scale scenarios. Besides, the traffic demand $\lambda[t]$ for the next time frame is unknown in practice. Such that the BW-split $\alpha[t]$ and flow-split vectors $\varphi[t]$ for frame t will be decided based on the previous states updated by the RAN layer and knowledge of the previous traffic demands $\{\lambda[t-1]\}_{\forall t}$. In order to attain high QoE for all users in each TTI, an efficient and adaptable solution to the long-term subproblem of (4.10) and (4.11) is required.

4.3.2 Sub-Optimization Problems

It is clear, both problems (4.10) and (4.11) must be solved on separate time scales, *i.e.*, on the long-term scale t and the short-term scale t_s. To reduce the computational complexity and information sharing as well as to provide a stable queuing system, the traffic demand vector $\boldsymbol{\lambda}[t]$, the flow-split decision vector $\boldsymbol{\varphi}[t]$ and the BW-splitting variable $\alpha[t]$ are only solved and updated once per time-frame t. In contrast, the power allocation vector $\boldsymbol{p}[t_s]$ and the RB allocation vector $\boldsymbol{\pi}[t_s]$ are optimized in every TTI t_s, adapting to dynamic environments. Although having different objective functions, we observe that P1 and P2 can share the solution development. In particular, the P2's objective function can be equivalently transformed to the maximization of the worst rate of the uRLLC services. By approximating the channel dispersion V in 4.2 as 1 for proper signal-to-noise ratio (SNR) ranges, the uRLLC rate has the same concavity as the eMBB rate in 4.1. Since both problems P1 and P2 have the same set of constraints, hereafter we propose solution development for only P1 to avoid redundancy.

Long-term Subproblem (L-SP): The joint optimization subproblem of the traffic demand, flow-split distribution, and dynamic RAN slicing at time-scale t is re-expressed as

$$\begin{aligned}
\text{L-SP}: \max_{\boldsymbol{\lambda},\boldsymbol{\varphi},\alpha}\ & \mathcal{R}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) && \text{(4.12a)}\\
\mathsf{s.t.}\quad & \boldsymbol{\varphi}_u[t] \in \boldsymbol{\varphi}[t], \forall t, u && \text{(4.12b)}\\
& R^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s]) \geq \varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{ur}} && \text{(4.12c)}\\
& \tau^{\mathsf{ur}}_u(\boldsymbol{\lambda}[t], \boldsymbol{\varphi}[t], \boldsymbol{\pi}[t_s], \boldsymbol{p}[t_s]) \leq D^{\mathsf{ur}},\ \forall u && \text{(4.12d)}\\
& \sum_{f_i=1}^{F_i} \beta_i \leq B_i[t],\ i \in \{1,2\} && \text{(4.12e)}\\
& 0 \leq \alpha[t] \leq 1. && \text{(4.12f)}
\end{aligned}$$

Although the L-SP (4.12) is non-convex due to the non-convexity of constraints (4.12c) and (4.12d), it cannot be solved directly by standard optimization techniques because $\boldsymbol{\lambda}[t]$ is completely unknown at the beginning of each frame. In the next section, three successive methods are proposed for solving this problem, that predict traffic demand, dynamic BW-split distribution, and dynamic flow-split variables as $\boldsymbol{\lambda}^*[t]$, $\boldsymbol{\alpha}^*[t]$ and $\boldsymbol{\varphi}^*[t]$ at the beginning of each frame t, respectively.

Short-term Subproblem (S-SP): Given $\boldsymbol{\lambda}^*[t]$, $\boldsymbol{\alpha}^*[t]$, and $\boldsymbol{\varphi}^*[t]$ forwarded from the non-RT RIC through the A1 interface, the resource allocation problem at time slot t_s in the near-RT RIC is expressed as

$$
\begin{aligned}
\text{S-SP} : \max_{\boldsymbol{\pi},\boldsymbol{p}} \quad & \mathcal{R}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) && \text{(4.13a)}\\
\mathsf{s.t.}\ & \boldsymbol{\pi}[t_s] \in \Pi[t_s],\ \forall t_s && \text{(4.13b)}\\
& \boldsymbol{p}[t_s] \in \mathscr{P}[t_s], \forall t_s && \text{(4.13c)}\\
& R^{\mathsf{em}}_{u}(\boldsymbol{p}^{\mathsf{em}}[t_s]) \geq R^{\mathsf{th}}, \forall u && \text{(4.13d)}\\
& \sum_{u} \left[R^{\mathsf{em}}_{m,u}(\boldsymbol{p}^{\mathsf{em}}[t_s]) + R^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s])\right] \leq C^{\mathsf{FH}}_{m}, \forall m && \text{(4.13e)}\\
& R^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s]) \geq \psi, \forall m, u && \text{(4.13f)}\\
& \tau^{\mathsf{ur}}_{u}(\boldsymbol{\pi}[t_s], \boldsymbol{p}[t_s]) \leq D^{\mathsf{ur}}, \forall u && \text{(4.13g)}\\
& \sum_{u} q_{m,u}[t_s] \leq Q^{\mathsf{max}}, \forall t_s, m \in \mathscr{M}_n && \text{(4.13h)}
\end{aligned}
$$

where $\psi = \varphi^{*}_{m,u}[t]\lambda^{*}_{u}[t]Z^{\mathsf{ur}}$. The S-SP (4.13) involves both binary ($\boldsymbol{\pi}$) and continuous ($\boldsymbol{p}$) optimization variables with nonlinear objective function and non-convex constraint (4.13e) at time slot t_s, which is still a MINCP problem. Since MINCP problems incorporate the optimizing challenges under integer variables with managing nonlinear functions, such problems comprise an immense class of difficult optimization problems.

4.3.3 Overall Intelligent Traffic Steering Deployment Architecture and Algorithm

In Fig. 4.4, we show the high-level organization of deployment scenarios and the end-to-end flow of the proposed algorithm within the Open RAN architecture. This is inspired by the second set of deployment scenarios listed in the technical report [99] by the Open RAN Alliance.

① The collected data, including performances/observations and resource updates from RAN components and near-RT RIC, are collected into a data collector located at the SMO. This process is done via the O1 interface. Based on these collected data in SMO, three rAPPs for solving L-SP are carried out at non-RT RIC. For $t = 1$, we assume a random traffic demand with a Poisson process and equal flow-split decision for all paths.

② Utilizing a data bus like Kafka, the collected data at the SMO is routed to non-RT RIC in the SMO.

③ The non-RT RIC queries the relevant ML/AI model, which is hosted in the AI server within the SMO. Once the model has been well-trained on the AI server, non-RT RIC is notified of the inference.

④ The scheduling xAPP in near-RT RIC is then loaded with inference results and policies via the A1 interface. Applications that are designed specifically for radio functions, or xAPPs, enable RAN components to be programmed.

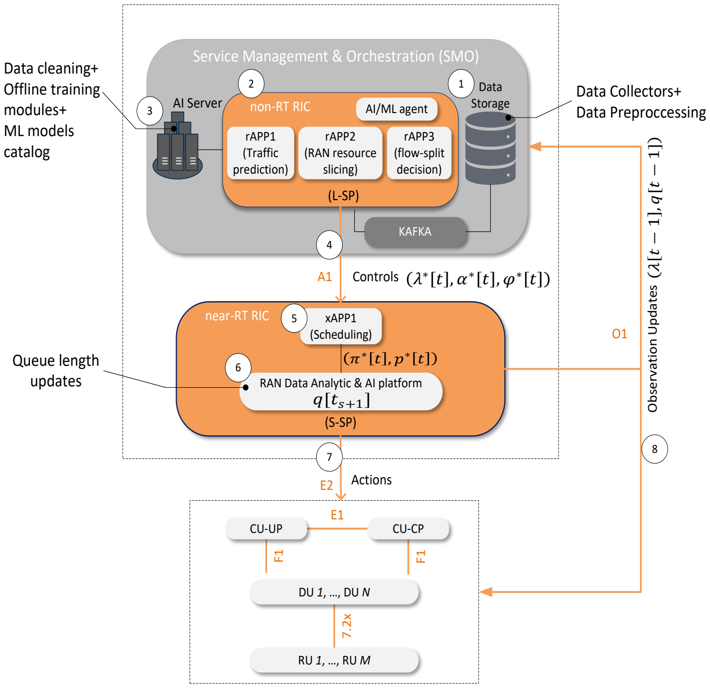


Figure 4.4: High-level structure of deploying the proposed intelligent traffic prediction and JIFDR management scheme within the Open RAN architecture.

(5) Given $\boldsymbol{\lambda}^*[t]$, $\boldsymbol{\alpha}^*[t]$ and $\boldsymbol{\varphi}^*[t]$, xAPP1 deployed in near-RT RIC controls congestion through MC technique and optimizes RAN resources and functions in each time-slot t_s by solving S-SP to obtain optimal solutions of RB allocation $\boldsymbol{\pi}^*[t_s]$ and power allocation $\boldsymbol{p}^*[t_s]$.

(6) Subsequently, the RAN Data Analytic component in near-RT RIC updates queue lengths.

(7) Through the E2 interface, the relevant solution is transferred to CU or DU layers.

(8) After S_i TTI (*i.e.*, one frame), the performance and observations (*e.g.* $\boldsymbol{q}[t-1]$, $\boldsymbol{\lambda}[t-1]$) are updated to SMO through the O1 interface to re-estimate the traffic demand $\boldsymbol{\lambda}^*[t+1]$ and flow-split decision $\boldsymbol{\varphi}^*[t+1]$.

The overall intelligent TS algorithm to solve the JIFDR problem (4.10) is summarized in Algorithm 2, where the solutions for subproblems will be detailed in Section 4.4. It is straightforward to develop a similar procedure to solve problem (4.11).

4.4 Proposed Frameworks for Solving Subproblems

We are now in a position to solve the L-SP and S-SP on different time scales. The optimal solutions for all optimization variables (α $\boldsymbol{\varphi}$, $\boldsymbol{\pi}$ and $\boldsymbol{p}$) strongly depend on the predicted

Algorithm 2 Proposed Intelligent Traffic Steering Algorithm to Solve JIFDR Problem (4.10)

```
**Initialization:** Set $t = 1$, $t_s = 1$, $\varphi_u[1] = \frac{1}{M}[1, \ldots, 1]$ and $\alpha[1] = \frac{1}{2}$; all initial queues are
  set to be empty $q_{m,u}[1] = 0$ and $\boldsymbol{q}[1] = 0$.
**for** $t = 1, 2, \ldots, T$ **do**
    **Traffic demand prediction:** Given ($\boldsymbol{\lambda}[t-1]$, $\boldsymbol{q}[t-1]$), non-RT RIC splits the available
    of all RUs' BW and traffic flows of all users by (4.14) and (4.15) based on the predicted
    traffic demand (or arrival data rate) $\boldsymbol{\lambda}^*[t]$ by solving the L-SP (4.12)
    **for** $t_s = 1, 2, \ldots, S_i$ with $s \in \{1, 2, \ldots, S_i\}$ **do**
        **Optimizing scheduling:** Given the queue-length vector $\boldsymbol{q}[t_s]$, and all long-term
        variables such as ($\boldsymbol{\lambda}^*[t], \boldsymbol{\alpha}^*[t]$, and $\boldsymbol{\varphi}^*[t]$), solve the problem (4.16) by Algorithm 3 to
        obtain the RB assignment ($\boldsymbol{\pi}^*$) and power allocation ($\boldsymbol{p}^*$)
        **Updating queue-lengths:** Queue-lengths are updated as

        $$q_{m,u}[t_{s+1}] = \max\{\Big[q_{m,u}[t_s] + \varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{x}}\delta_i - r^{\mathsf{x}}_{m,u}[t_s]\delta_i\Big], 0\}$$

        where $\mathsf{x} \in \{\mathsf{ur}, \mathsf{em}\}$.
        Set $s = s + 1$
    **end for**
    Update $\{\boldsymbol{q}[t], \boldsymbol{\lambda}[t]\} = \{q_{m,u}[t], \lambda_u[t]\},\ \forall u \in \mathscr{U},\ m \in \mathscr{M}_n$
    Set $t = t + 1$
**end for**
```

traffic demand vector $\boldsymbol{\lambda}$, which often require prior knowledge of the actual traffic of all services stored at the data collector in SMO. Moreover, due to the dynamic environment and data collected from the RAN components being only updated to non-RT RIC on a long-term scale (*i.e.*, frame), the assumption of complete information is unrealistic. In this chapter, we aim to leverage observable historical system knowledge gathered over previous time slots via the O1 interface to build a smoother optimal response to maximize the long-term utility.

4.4.1 LSTM for Solving L-SP

As mentioned previously, the L-SP cannot be solved directly by standard optimization techniques since $\boldsymbol{\lambda}[t]$ and $\boldsymbol{q}[t_s]$ are often unknown at the beginning of each frame. Besides, the main challenge in optimizing traffic steering is to predict traffic precisely before the beginning of the next frame. An optimal policy cannot be implemented with an imprecise prediction of future traffic. In this section, utilizing a deep learning approach, we develop a data-driven real-time traffic demand prediction method. We suppose that the queue length of data flows u in the next frame will depend on the traffic demand of data flows u in the current and previous ones. Basically, RNN models utilize the current input as well as the output of one layer as the input for the subsequent layer. In such models, each layer is fed by the very first layer's input. This allows the RNN model to learn from the current and former time steps and then provides more precise predictions for traffic flows. These

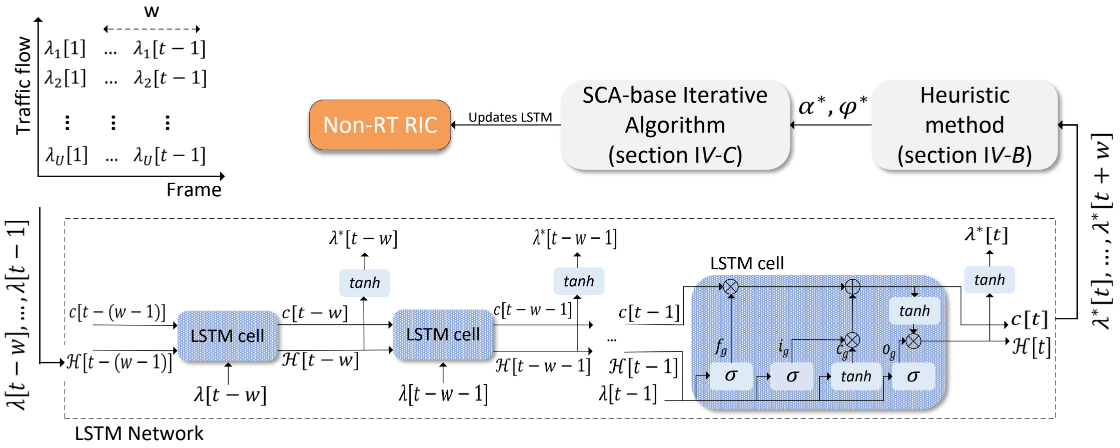


FIGURE 4.5: Implementing the proposed JIFDR management scheme at time-frame t.

standard RNN models suffer from short-term memory owing to the vanishing and exploding gradient problems, which appear with longer data sequences. Due to these difficulties, the gradient either entirely disappears or explodes to a very high value, which makes it difficult to learn some long-period dependencies. To address the long-term dependency issue, the LSTM model has seen extensive use in the field of traffic prediction due to its capabilities in dealing with long time-series flow data. As a result, we utilize the LSTM RNN to learn and predict the traffic pattern of all users in the considered Open RAN architecture.

The fact that LSTM includes a memory cell to keep observable data allows them to handle long-term time series. As shown in Fig. 4.5, the structure of standard LSTM cells learns through four main gates, *namely* input (i_g), forget (f_g), cell state-update (c_g), and output (o_g), that allow the input data to pass from the previous cells in the learning procedure. The output calculated by the input gate (i_g) and the cell state update (c_g) modify the current cell's state ($c[t]$), while the forget gate enables the current cell to discard or preserve the previous state value. To determine this, we take into account the output of the previous hidden state ($\mathcal{H}[t-1]$) and the actual input data ($\boldsymbol{\lambda}[t-1]$). The new cell state's value is based on the actual input and previous output of the cell. In contrast to other gates that employ the Sigmoid function, the cell state update benefits the hyperbolic tangent as an activation function that yields values between -1 and 1. Eventually, the input, forget, and cell state update gates are combined to create the current cell state. The current cell's output is determined as a function of the previous timestep's output ($\mathcal{H}[t-1]$), the actual input data ($\boldsymbol{\lambda}[t-1]$), and the cell state ($c[t-1]$) through the output gate. Lastly, after crossing through an activation function, the prediction value is calculated. Each LSTM layer comprises a chain of LSTM cells, in which the computed operation of each cell is transmitted to the next cell as an input. As illustrated in Fig. 4.5, the temporal pattern of the mentioned parameter is learned through the current and a window of previous traffic demands value with the length W $\{\boldsymbol{\lambda[t-W]}, \boldsymbol{\lambda[t-W+1]}, \dots, \boldsymbol{\lambda[t-1]}\}$ to predict future values.

The LSTM model is trained at non-RT RIC in the Open RAN architecture, using long-term data gathered from RAN via O1. The near-RT RIC of the Open RAN is then given access via the A1 interface to the trained model for inference. Upon the inference outcome, the intelligent TS is applied through the MC technique to enhance the associated key performance indicators (KPIs). Traffic demand prediction and the corresponding intelligent TS schemes are continually implemented till the desired KPI values, or the required QoS of traffic, are met. In the following, the network parameter of data arrival rate $\boldsymbol{\lambda}$ is continuously monitored across all cells of RUs. Upon predicting the data arrival rate per frame, the flow-split distribution, dynamic RAN slicing, and radio resource management with the MC technique can be applied to steer data flows. The weights of the RNN model are eventually updated depending on the actual parameter's value to reflect changes and enhance the performance till the goal KPI criteria are met if the prediction outcome is incorrect.

4.4.2 Heuristic Methods for Predicting $\boldsymbol{\alpha}[t]$ and $\boldsymbol{\varphi}[t]$

Upon the inference outcome of the LSTM model, the predicted traffic demands at the next frame $\boldsymbol{\lambda}^*[t]$ are transmitted immediately to two other embedded rAPPs in non-RT RIC for optimizing the dynamic bandwidth separation, $\boldsymbol{\alpha}[t]$ and flow-split decisions, $\boldsymbol{\varphi}[t]$. For efficient deployment, these parameters are designed in a longer time scale, *i.e.*, on the frame basis compared to the time slot basis of power allocation and RB assignment. Therefore, at the beginning of each frame, $\boldsymbol{\alpha}[t]$ and $\boldsymbol{\varphi}[t]$ should be determined upon getting the predicted traffic demands. Having optimum values of the bandwidth separation and flow split is very difficult, if not possible, because of the unknown CSI of future time slots in the current frame. Therefore, we propose an efficient heuristic algorithm to determine $\boldsymbol{\alpha}[t]$ and $\boldsymbol{\varphi}[t]$ based on $\boldsymbol{\lambda}^*[t]$. An intuitive way is to allocate the bandwidth to each service proportionally to the corresponding traffic demands. However, since the amount of uRLLC traffic is much smaller than the amount of eMBB traffic, this method is not efficient in meeting the stringent latency requirement of uRLLC applications. To tackle this, we incorporate the maximum tolerable delays of both services and the total traffic demands. Thus, the bandwidth separation between eMBB and URLLC services is computed as follows

$$\alpha^*[t] = \frac{\sum_{\mathcal{U}^{\mathsf{ur}}} \lambda_u^*[t]}{\sum_{\mathcal{U}^{\mathsf{em}}} \lambda_u^*[t]} \times \frac{\tau_{\mathsf{th}}^{\mathsf{em}}}{\tau_{\mathsf{th}}^{\mathsf{ur}}} \tag{4.14}$$

where $\tau_{\mathsf{th}}^{\mathsf{ur}}$ and $\tau_{\mathsf{th}}^{\mathsf{em}}$ represent the maximum allowed latency for uRLLC and eMBB services, respectively.

To plan the flow splitting factor $\varphi_u[t]$, we consider each DU's capacity in delivering user traffic demands u. Because we do not know the data rate for the user in the next frame, we take the moving average of the rate in the most recent time slots. For a generic user u (can be a uRLLC or eMBB user), let us define $\bar{R}_{m,u}[t] = \frac{1}{W}\sum_{l=t-W+1}^{t} R_{m,u}[l]$, where $R_{m,u}[l]$ is the achievable rate of user u served RU m at time slot l, and W is the window size. The flow split for user u to RU m is computed as follows

$$\varphi_{m,u}^*[t] = \frac{\bar{R}_{m,u}[t]}{\sum_{m\in\mathcal{M}_n} \bar{R}_{m,u}[t]},\ \forall m, u. \tag{4.15}$$

4.4.3 SCA-based Iterative Algorithm for solving S-SP

To solve the problem (4.13) as a MINCP, we first relax binary variables to continuous ones (*i.e.*, the box constraints between 0 and 1) and transform constraint (4.13e) into a more traceable form, which the SCA-based iterative algorithm can efficiently solve.

Algorithm 3 The Proposed SCA-based Iterative Algorithm to Solve S-SP (4.16)

```
1: Initialization: Set $j := 0$ and generate initial feasible points for $(\boldsymbol{\pi}^{(0)}[t_s], \boldsymbol{p}^{(0)}[t_s]) := (\boldsymbol{\pi}[t_{s-1}], \boldsymbol{p}[t_{s-1}])$ to constraints in S-SP2 (4.19)
2: repeat
3:     Solve (4.19) to obtain $(\boldsymbol{\pi}^*[t_s], \boldsymbol{p}^*[t_s])$ and $\Xi^*[t_s]$;
4:     Update $(\boldsymbol{\pi}^{(j)}[t_s], \boldsymbol{p}^{(j)}[t_s]) := (\boldsymbol{\pi}^*[t_s], \boldsymbol{p}^*[t_s])$ and $\Xi^{(j)}[t_s] := \Xi^*[t_s]$;
5:     Set $j := j + 1$;
6: until Convergence or $|\Xi^{(j)}[t_s] - \Xi^{(j-1)}[t_s]| \leq \epsilon$ {/*Satisfying a given accuracy level*/}
7: Recover an exact binary by computing $\boldsymbol{\pi}^*[t_s] = \lfloor \boldsymbol{\pi}^{(j)}[t_s] + 0.5 \rfloor$ and repeat step 1 to 5 for given $\boldsymbol{\pi}^*[t_s]$;
8: Output: $(\boldsymbol{\pi}^*[t_s], \boldsymbol{p}^*[t_s])$.
```

Penalty function: We bring forward the following penalty function to accelerate the convergence of the proposed iterative algorithm that will be detailed shortly $\mathcal{P}(\boldsymbol{\pi}) = \sum_{t_s, f_i, m, u} \big[(\pi^{\mathsf{em}}_{m,u,f_i}[t_s])^2 + (\pi^{\mathsf{ur}}_{m,u,f_i}[t_s])^2 - \pi^{\mathsf{em}}_{m,u,f_i}[t_s] - \pi^{\mathsf{ur}}_{m,u,f_i}[t_s]\big]$ which is convex in $\boldsymbol{\pi}[t_s]$. It is clear that $\mathcal{P}(\boldsymbol{\pi}) \leq 0$ for any $\pi^{\mathsf{x}}_{m,u,f_i}[t_s] \in [0,1]$, which is useful to penalize the relaxed variables to obtain near-precise binary solutions at optimum (*i.e.*, satisfying (4.13b)). By incorporating $\mathcal{P}(\boldsymbol{\pi})$ into the objective function of (4.13b), the parameterized relaxed problem is expressed as

$$\begin{aligned}
\text{S-SP1}: \max_{\boldsymbol{\pi}, \boldsymbol{p}} \quad & \mathcal{R}^{\mathsf{em}} + \omega \mathcal{P}(\boldsymbol{\pi}) && (4.16\text{a})\\
\text{s.t.} \quad & \boldsymbol{\pi}[t_s] \in \tilde{\Pi}[t_s], \quad \forall t_s, \forall u \in \mathcal{U} && (4.16\text{b})\\
& (4.13\text{c}) - (4.13\text{h}) && (4.16\text{c})
\end{aligned}$$

where $\tilde{\Pi}[t_s] \triangleq \{\pi^{\mathsf{em}}_{m,u,f_i}[t_s], \pi^{\mathsf{ur}}_{m,u,f_i}[t_s] \in [0,1] | \sum_{m,u} [\pi^{\mathsf{em}}_{m,u,f_i}[t_s] + \pi^{\mathsf{ur}}_{m,u,f_i}[t_s]] \leq 1\}$ and $\omega > 0$ denotes a determined penalty parameter.

Proposition 2. *Problems (4.13) and (4.16) share the same optimal solution, i.e.,* $(\boldsymbol{\pi}^*, \boldsymbol{p}^*)$*, considering an suitable positive value of* ω.

The proof is directly followed [79] by showing the fact that $\mathcal{P}(\boldsymbol{\pi}) = 0$ at optimum in maximizing of the objective function (4.16). It implies that a constant ω always exists to guarantee that $\boldsymbol{\pi}$ are binary at optimum and the relaxation is tight. Practically, it is acceptable if $\mathcal{P}(\boldsymbol{\pi}) \leq \varepsilon$ for a tiny ε, which results in a nearly precise optimal solution.

In problem (4.16), the objective function is non-concave due to $\mathcal{P}(\boldsymbol{\pi})$, while constraint (4.13e) is non-convex. Based on the SCA method, the first-order Taylor approximation is used to linearize the function $\mathcal{P}(\boldsymbol{\pi})$ at the j-th iteration as follows

$$\mathcal{P}^{(j)}(\boldsymbol{\pi}) \triangleq \sum_{m,u,f_i} \Big[\pi^{\mathsf{em}}_{m,u,f_i}[t_s](2\pi^{\mathsf{em},(j)}_{m,u,f_i}[t_s]-1) - (\pi^{\mathsf{em},(j)}_{m,u,f_i}[t_s])^2 + \pi^{\mathsf{ur}}_{m,u,f_i}[t_s](2\pi^{\mathsf{ur},(j)}_{m,u,f_i}[t_s]-1) - (\pi^{\mathsf{ur},(j)}_{m,u,f_i}[t_s])^2\Big], \quad (4.17)$$

where $\mathcal{P}(\boldsymbol{\pi}) \geq \mathcal{P}^{(j)}(\boldsymbol{\pi})$ and $\mathcal{P}(\boldsymbol{\pi}^{(j)}) = \mathcal{P}^{(j)}(\boldsymbol{\pi}^{(j)})$.

To address constraint (4.13e), we indicate its LHS as $R_m(\boldsymbol{p}[t_s]) \triangleq \sum_u \big[R^{\mathsf{em}}_{m,u}(\boldsymbol{p}^{\mathsf{em}}[t_s]) + R^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s],\, \boldsymbol{\pi}^{\mathsf{ur}}[t_s])\big]$, which is concave in $\boldsymbol{p}[t_s]$. Thus, the function $R_m(\boldsymbol{p}[t_s])$ can be approximated at the feasible point $\boldsymbol{p}^{(j)}[t_s]$ as

$$R^{(j)}_m(\boldsymbol{p}[t_s]) \triangleq R_m(\boldsymbol{p}^{(j)}[t_s]) - \sum_{u,f_i} \beta_i \log_2(e) \frac{\pi^{\mathsf{ur}}_{m,u,f_i}[t_s] Q^{-1}(P_e)}{\sqrt{\delta_i \beta_i}} + \frac{\beta_i}{\ln 2} \sum_{u,f_i,\mathsf{x}} (p^{\mathsf{x}}_{m,u,f_i}[t_s] - p^{\mathsf{x},(j)}_{m,u,f_i}[t_s]) \times \Big[\frac{g_{m,u,f_i}[t_s]}{N_0 + p^{\mathsf{x},(j)}_{m,u,f_i} g_{m,u,f_i}[t_s]}\Big]. \quad (4.18)$$

The convex approximate program of (4.16) solved at iteration j is stated as follows, taking into account all the aforementioned approximations

$$\begin{aligned} \text{S-SP2} : \max_{\boldsymbol{\pi},\boldsymbol{p}} \quad & \Xi^{(j)} \triangleq \mathcal{R}^{\mathsf{em}} + \omega \mathcal{P}^{(j)}(\boldsymbol{\pi}) && (4.19\text{a}) \\ \text{s.t.} \quad & (4.13\text{c}), (4.13\text{d}), (4.13\text{f}) - (4.13\text{h}), (4.16\text{b}) && (4.19\text{b}) \\ & R^{(j)}_m(\boldsymbol{p}[t_s]) \leq C^{\mathsf{FH}}_m, \forall m \in \mathcal{M}_n. && (4.19\text{c}) \end{aligned}$$

Algorithm 3 provides a summary of the SCA-based iterative algorithm. Step 6 is used to recover an exact binary solution, then Steps 1–5 are repeated to refine the final solution in order to ensure a feasible solution to the problem (4.16). The study gap to the global optimal solution is not considered in this chapter and is left for future study.

Convergence and complexity analysis: The development of the proposed iterative Algorithm 3 is based on the SCA method [100]. The approximations in (4.17) and (4.18) satisfy the three key inner approximation properties given in [80], while other constraints are already linear and quadratic. In particular, the solution of (4.19) is always feasible for the parameterized relaxed problem (4.16) but not vice versa. In addition, Algorithm 3 generates a sequence of the improved solutions $\{\boldsymbol{\pi}^{(j)}, \boldsymbol{p}^{(j)}\}$ in the sense that $\Xi^{(j+1)} \geq \Xi^{(j)}, \forall j$. By [100, Theorem 1], if the number of iterations is sufficiently large, the sequence $\{\boldsymbol{\pi}^{(j)}, \boldsymbol{p}^{(j)}\}$ converges to at least a local optimal solution of (4.16), satisfying the Karush-Kuhn-Tucker (KKT) conditions [100, Theorem 1]. On the other hand, for each numerology i, the convex approximate program (4.19) has $2MUF_i$ scalar decision variables and $2MUF_i + 4M + 3U$ linear and quadratic constraints. As a result, the worst-case computation complexity of Algorithm 3 in each iteration is estimated as $\mathcal{O}(\sqrt{2MUF_i + 4M + 3U}(2MUF_i)^3)$, following the interior-point method [101, Chapter 6].

4.5 Performance Evaluations And Numerical Results

4.5.1 Simulation Setup and Parameters

We consider a scenario where all users are uniformly distributed in a circular area with a radius of 500 m, while the locations of RUs are fixed. One RU is located in the central area, serving three sectors, each of which includes one RU. The RU-user channels are generated as Rayleigh fading with the path-loss $\mathsf{PL}_{\text{RU}-\text{USER}} = 128.1 + 37.6 \log_{10}(d/1000)$ dB. The penalty factor is set to decrease after each TTI as $\omega[t_s] = 20 + 10/(1 + t_s)$ to guarantee the convergence of the short-term subproblem. To estimate the future traffic for the upcoming frames, an RNN model's parameters, which include 2 fully connected hidden layers and 50 LSTM units (neurons), are trained. The operators can configure these parameters based on the provided data and its periodicity. In our setup, the Poisson traffic model has been used to generate traffic for both eMBB and uRLLC services. The RNN training is carried out over the traffic dataset of the cellular network following a Poisson distribution, with mean arrival rates of 20 and 2.5 for eMBB and uRLLC traffics, respectively [93]. The mean arrival rate is a configurable parameter of the simulator. Incoming traffic packets are sorted in a first-come, first-serve buffer. The dataset contains network measurement in terms of arrival rate collected from M RUs over a horizon of $T = 10000$ traffic observations over a duration of 100 seconds. The open-source, high-level TensorFlow version 1.13.1 application programming interface, Keras, is used to implement the RNN model. All experiments are done on a Dell desktop computer with an Intel R CPU @ 3.0 GHz. Simulation parameters, including the LSTM model's hyperparameters, are summarized in Table 4.2.

We put into practice the following five benchmark schemes for performance comparison:

1. *Fixed numerology (FIX-NUM)*: In this scheme, the TTI is considered the same for both services as the LTE standard (*i.e.*, 0.5 ms) with the SCS of 180 kHz. The resource allocation, flow-split decision, and dynamic BW-split for both traffic follow Algorithm 2 with some slight modifications.

2. *Equal Flow-Split Distribution (EFSD)*: In order to demonstrate the importance of optimizing the flow-split distribution per frame, this scheme considers the equal flow-split for each traffic to RUs, *i.e.*, $\varphi_{m,u} = \frac{1}{M}$, $\forall u \in \mathcal{U}$ and follows Algorithm 2.

3. *Equal Power Allocation (EPA)*: The RBs' allocation $\boldsymbol{\pi}$ is optimized by Algorithm 2 for an equal power allocated to all users and subcarriers.

4. *Single Connectivity with uRLLC Priority (SCUP)*: To reveal the performance improvement of MC in heterogeneous wireless networks, this scheme provides the single connectivity (SC) scheme with uRLLC priority in the presence of interference. Due to the stringent requirement of latency, uRLLC will be predominantly guaranteed, and then the remaining resources will be occupied by eMBB users. In this regard, this scheme considers M RUs with disjoint dedicated users while following Algorithm 2.

5. *Proposed Problem in Presence of Known Traffic Demand (PKTD)*: This scheme investigates the performance of both traffics in the presence of known traffic demand

Table 4.2: Simulation Parameters

Parameter	**Value**	**Parameter**	**Value**
No. of RUs	4	Predetermined uRLLC latency (D^{ur})	0.5 ms
No. of eMBB users	12	Predetermined eMBB data rate (R^{th})	1 Mbps
No. of uRLLC users	8	Maximum FH capacity (C^{FH})	100 Mbps
BW of RU	20 MHz	Maximum MH capacity (C^{MH})	5 Gbps
Error probability (P_e)	10^{-3}	Maximum RU's queue-length (Q^{max})	10 KB
Power of RU	46 dBm	No. of LSTM layer	2
Noise power (N_0)	-110 dBm	No. of LSTM unit	50
uRLLC packet size (Z^{ur})	1 KB	No. of epoch	50
eMBB packet size (Z^{em})	125 KB	Activation function	*tanh*
Length of time-frame	10 ms	Optimizer	*adam*

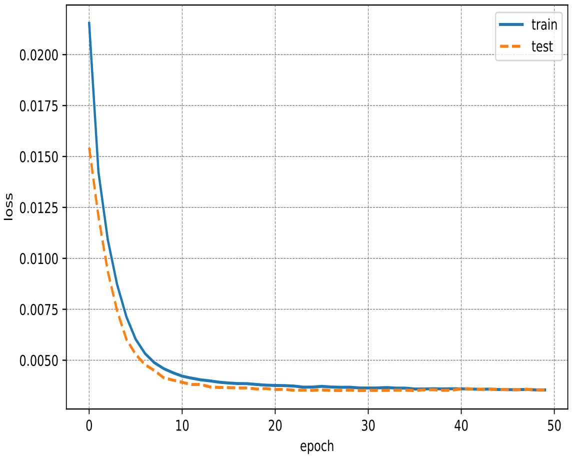


Figure 4.6: Training and validation loss for the LSTM RNN model.

λ. In practice, the obtained results of this scheme in the presence of unknown traffic demands show the accuracy of the LSTM model of the proposed method.

4.5.2 Numerical Results and Discussions

First, in order to investigate the LSTM's convergence, we monitor the value of the loss function as MSE and keep the training process until the training loss is typically identical to the validation loss after a specific number of epochs. Since the mean arrival rates of both traffics are not in the same range, we normalize traffic demands in the pre-processing phase through the *MinMaxScaler* normalization method from Sklearn. We then divide the data into two sets, which are 80% for training and 20% for validation. Fig. 4.6 plots the training and validation losses for the LSTM model with the most suitable turning hyperparameters,

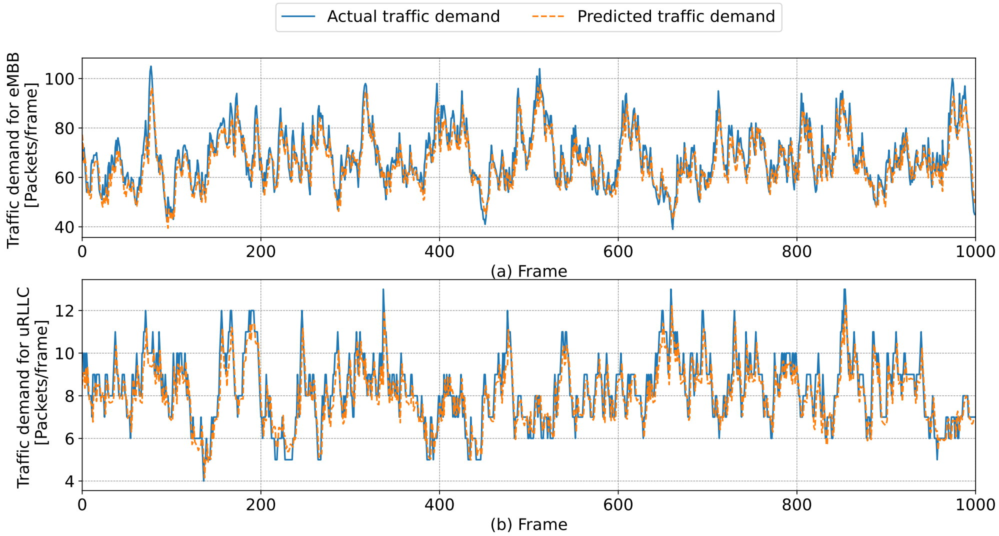


FIGURE 4.7: Traffic demand prediction in Open RAN.

which converge after 50 epochs. It should be mentioned that setting the desirable number of epochs prevents model overfitting. From Table 4.3, we find that the activation function of *tanh* works better than *relu* and *sigmoid*. In the same condition, increasing the number of LSTM layers and decreasing the number of units per layer do not help reduce the MSE value. Based on the search result, the *adam* optimizer converges faster than others, whereas it takes less time for the model's training. In our case, the dropout value is 0.01 for both hidden layers. As a result, Table 4.3 shows the search parameters to find the best parameters for the final LSTM-RNN model.

The effectiveness of the LSTM RNN model in both traffic demands is represented in Fig. 4.7 to illustrate the performance of the ML model prediction. The actual and predicted values for one of the eMBB and uRLLC traffic demands in the proposed system model are shown in Fig. 4.7 (a) and Fig. 4.7 (b), respectively. As it is clear from these figures, the trained LSTM-RNN model performs outstandingly in capturing the dynamic traffic demand of services over time. The difference between predicted and actual traffic demands is entirely small. The MSE value has been calculated as a performance measurement to validate the accuracy of the implemented LSTM model. For instance, the measured MSE values of the selected eMBB users in Fig. 4.7 (a) and uRLLC users in Fig. 4.7 (b) are 0.00315 and 0.00323, respectively.

To evaluate the eMBB data rate with different resource allocation schemes, Fig. 4.8 illustrates the sum data rate of eMBB users over different maximum RUs' power budgets from 10 to 46 dBm. Unsurprisingly, the PKTD provides the best performance and acts as the upper bound of all strategies. It can be observed that the gap between our proposed framework and PKTD is less than 2%, which proves the efficiency of the LSTM RNN model in predicting the dynamic traffic demand over time. While the proposed method provides

Table 4.3: Hyperparameters for the different Performing LSTM Models

No. of LSTM layers	No. of LSTM units	No. of epochs	Activation function	MSE
2	20	30	*relu*	0.00641
2	50	30	*relu*	0.00382
3	50	100	*relu*	0.00493
3	50	30	*sigmoid*	0.01281
2	50	30	*sigmoid*	0.00782
2	50	100	*tanh*	0.00421
3	20	30	*tanh*	0.00613
2	**50**	**50**	***tanh***	**0.00331**

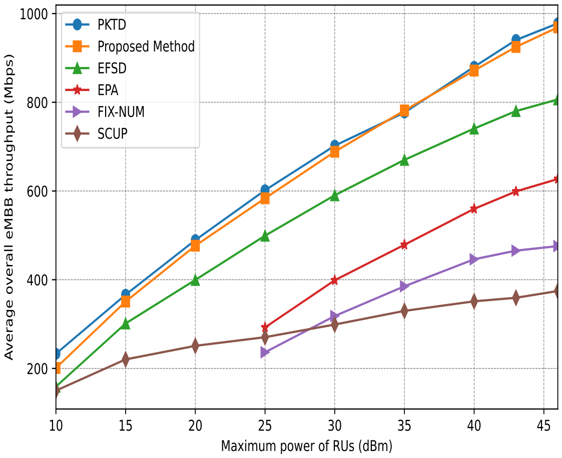


Figure 4.8: Average overall eMBB data rate versus $P^{\max}$.

the highest eMBB data rate compared to other benchmark schemes in Fig. 4.8. Compared to SCUP, FIX-NUM, EPA, and EFSD, the proposed method offers 130.89%, 116.32%, 71.92%, and 19.21% gains at the typical power value of $P^{\max} = 30$ dBm, respectively. Furthermore, EPA and FIX-NUM work over $P^{\max} \geq 25$ dBm, while they are infeasible when the maximum RUs' power is less than 25 dBm. Hence, this phenomenon shows the advantage of our proposed method over these schemes, especially at a small $P^{\max}$. Besides, as we mentioned previously, the MC technique plays a vital role in enhancing the eMBB data rate. The gap between the JIFDR framework considering the MC technique and SCUP grows with increasing the maximum power budget of RUs. While the overall eMBB data rates obtained via JIFDR, EFSD, and SCUP are close at $P^{\max} = 10$ dBm, by increasing $P^{\max}$, the MC-based schemes of JIFDR and EFSD significantly exceed that of SCUP.

In order to show the performance of the proposed method on uRLLC latency, Fig. 4.9 represents the worst-user uRLLC latency under different maximum power of RUs. Similar to the first optimization problem (P1), increasing the maximum power of RUs significantly

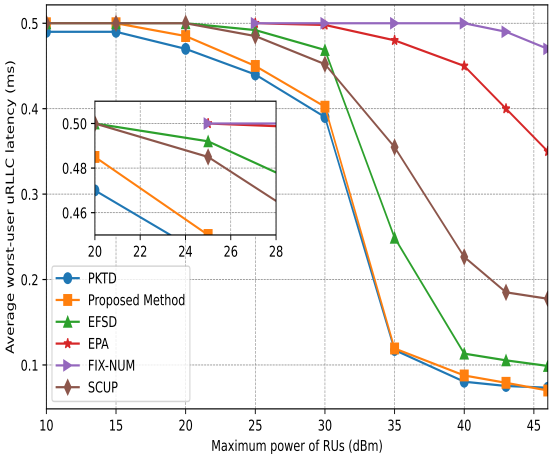


Figure 4.9: Average worst-user uRLLC latency versus $P^{\max}$.

affects the eMBB data rate improvement, resulting in an efficient reduction of uRLLC latency in the second optimization problem (P2). As we can see from Fig. 4.9, the uRLLC latency of the proposed method is almost equal to PKTD, which again confirms the accuracy of the LSTM RNN model in predicting the dynamic traffic demand. The performance gain in terms of latency of the proposed method is 181.32% and 49.47% Compared to SCUP and EFSD at $P^{\max} = 40$ dBm. According to the empty region of two benchmark schemes, FIX-NUM and EPA, in the range $P^{\max} \leq 25$ dBm, results from Fig. 4.9 show that these schemes are infeasible over the mentioned range of $P^{\max}$ while having a significant difference in uRLLC latency with the proposed method. Clearly, the EFSD scheme in Fig. 4.9 greatly outperforms the SCUP scheme. On the one hand, the uRLLC and eMBB traffic are sliced in various virtual slices in SCUP, while the size of the uRLLC traffic packet is considerably smaller than the eMBB packet size. Hence, the assigned slice to uRLLC could meet the uRLLC traffic's requirements alone without waiting in a queue. On the other hand, the SCUP scheme is not able to aggregate multiple links and allow users to connect to more than one RU to achieve the highest data rate.

Fig. 4.10 depicts the average backlog in the queue under the maximum power budget of RUs with different benchmark schemes. As can be seen, the higher the power budget $P^{\max}$, the lower the average queue length. Similar to the two previous figures, results from the proposed method and PKTD are very close to each other. As expected, the SCUP scheme yields the worst performance in terms of the average queue length, whereas the proposed method yields the best one in Fig. 4.10. Two FIX-NUM and EPA schemes are infeasible when $P^{\max} < 25$ dBm. Clearly, the EFSD benchmark scheme performs in a better way than the FIX-NUM, EPA, and SCUP schemes, while EFSD and EPA provide very close performance to each other for $P^{\max} > 35$ dBm. On the other hand, during the joint scheduling of uRLLC and eMBB traffics, we have numerically observed that uRLLC users always prefer to have only one link in various system setups. This issue indicates that

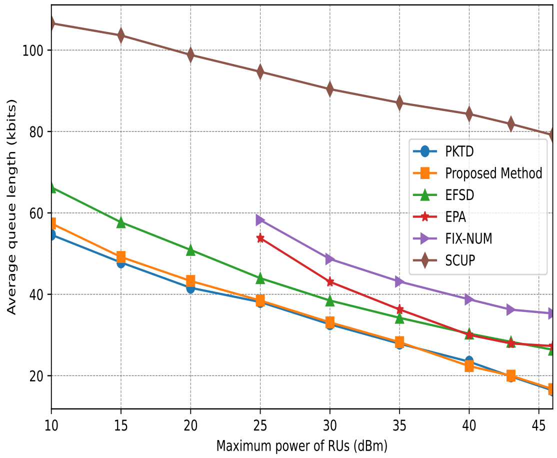


Figure 4.10: Average of queue lengths versus $P^{\max}$.

a single connection is generally the best option for traffic with a small data packet size. In contrast, the MC technique is typically a nice option for traffic with high data packet size, *i.e.*, eMBB.

As we mentioned before, the MC is an effective technique to improve the data rate for eMBB traffic, especially when the system model faces a limited bandwidth. To demonstrate this, Fig. 4.11 shows the impact of the increasing number of RUs on overall eMBB data rate. All simulation parameters are assumed unchanged during the simulation of Fig. 4.11, except the number of eMBB and uRLLC users, which are considered 21 and 14, respectively. It is clear from Fig. 4.11 that the eMBB data rate rises with the number of RUs, which means an increase in the number of available RBs. As we expected, the PKTD also works as an upper bound for all schemes, regardless of the number of RUs. The small gap between our proposed framework and PKTD (about 2%) shows the high accuracy of traffic prediction by embedded LSTM in the non-RT-RIC component. Compared to other existing benchmark schemes, the proposed method offers the highest data rate. Due to the crucial role of MC in the network, SCUP has the worst performance among all the schemes, with increasing the number of RUs ($M \geq 3$) in the network model. For $M = 5$, the schemes with MC (*i.e.*, the proposed method, EFSD, EPA, and FIX-NUM) have the performance gain of 168.2%, 90.67%, 42.68%, and 19.45% relatively compared to the without MC *i.e.*, SCUP. It should be noted that the SCUP outperforms FIX-NUM at $M \leq 3$, demonstrating the advantage of mixed numerology over fixed numerology. However, as the number of RUs increases to $M \geq 3$, the SCUP is no longer able to offset the weak performance of the single connection scheme. Since all users associate with only one RU, the performance of all schemes is almost the same, while the MC brings a large gap between MC and SCUP by increasing the number of RUs. It is noted that the gap between the proposed method and other schemes also grows with the number of RUs.

Finally, we examine the convergence behavior of the proposed Algorithm 2, comparing

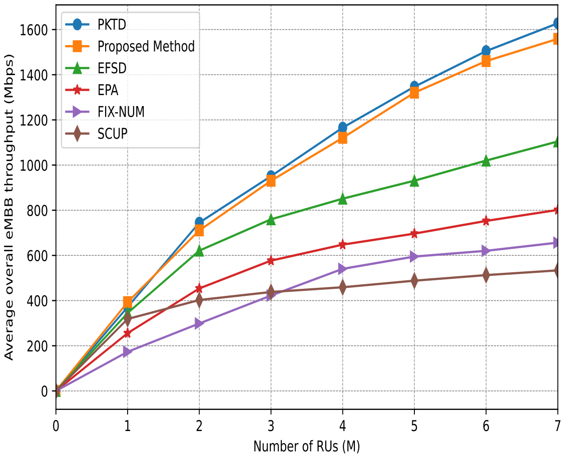


FIGURE 4.11: Average of eMBB data rate versus number of RUs (M) with considering 21 eMBB users and 14 uRLLC users.

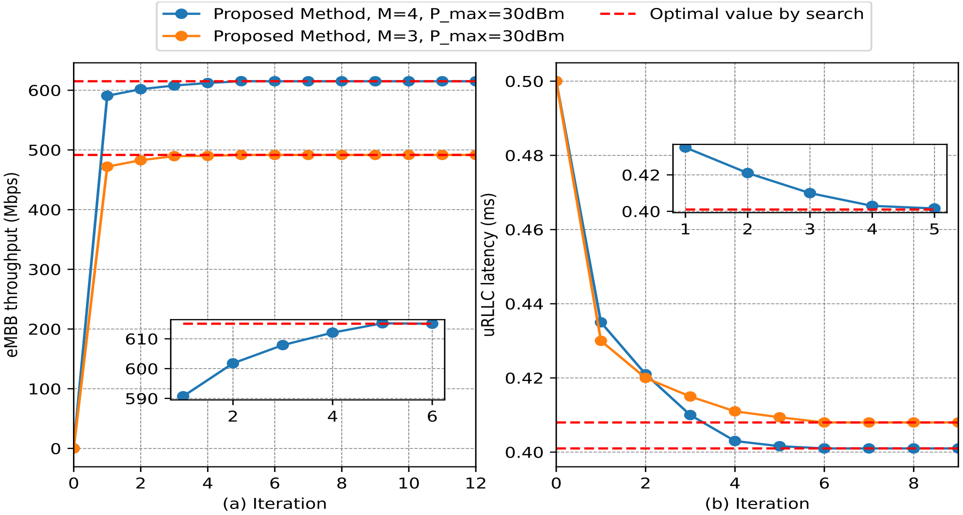


FIGURE 4.12: Convergence behaviour of the proposed Algorithm 3.

the optimal value through the exhaustive search for $P^{\max} = 30$ dBm under the different number of RUs in Fig. 4.12. It is shown that the proposed algorithm for both problems (P1) and (P2) converges quickly, taking less than 10 iterations to reach the optimal value within an increment, which is smaller than a given threshold $\epsilon = 10^{-4}$. As expected, based on Figs. 4.12(a) and . 4.12(b), as the number of RUs increases in such a network, the eMBB data rate increases, but it does not affect the uRLLC latency remarkably. As we mentioned before, uRLLC users frequently tend to link to only one RU because of their small packet size. There is almost the same convergence speed for both cases with 3 RUs and 4 RUs.

Nevertheless, the case with 4 RUs needs a little more time for CVXPY to solve the MINCP in each step due to additional optimization variables.

4.6 Summary

In this chapter, we have developed a novel intelligent TS framework in the presence of unknown dynamic traffic to meet the competing demands of uRLLC and eMBB services in beyond 5G networks based on dynamic MC. To achieve the maximum data rate for eMBB traffic while guaranteeing the minimum uRLLC latency requirement and vice versa, we have proposed a joint intelligent traffic prediction, flow-split distribution, dynamic RAN slicing, and radio resource management scheme to schedule joint RBs and transmission power with mixed numerologies based on standardization in 5G NR. We have carried out a thorough analysis of e2e uRLLC latency. Due to the execution of the proposed problems in two different timescales, we have divided them into two long-term and short-term subproblems. To solve them, the LSTM method and SCA-based iterative algorithm have been developed to solve the formulated subproblems effectively. Thanks to LSTM, which predicts future traffic with high accuracy, the proposed method based on MC and mixed numerologies greatly improves resource utilization by adapting to dynamic traffic demands compared to benchmark schemes. One of the future works is to deploy more advanced techniques (*e.g.*, deep reinforcement learning) to better estimate $\alpha[t]$ and $\varphi[t]$.

Chapter 5

Empowering Traffic Steering in 6G Open RAN with Deep Reinforcement Learning

The sixth-generation (6G) wireless network landscape is evolving toward enhanced programmability, virtualization, and intelligence to support heterogeneous use cases. The O-RAN Alliance is pivotal in this transition, introducing a disaggregated architecture and open interfaces within the 6G network. Chapter 5 explores an intelligent traffic steering (TS) scheme within the Open radio access network (RAN) architecture, aimed at improving overall system performance. Our novel TS algorithm efficiently manages diverse services, improving shared infrastructure performance amid unpredictable demand fluctuations. To address challenges like varying channel conditions and dynamic traffic demands, we propose a multi-layer optimization framework tailored to different timescales. Techniques such as long-short-term memory (LSTM), heuristics, and multi-agent deep reinforcement learning (MADRL) are employed within the non-real-time (non-RT) RAN intelligent controller (RIC). These techniques collaborate to make decisions on a larger timescale, defining custom control applications such as the intelligent TS-xAPP deployed at the near-real-time (near-RT) RIC. Meanwhile, optimization on a smaller timescale occurs at the RAN layer after receiving inferences/policies from RICs to address dynamic environments. The simulation results confirm the system's effectiveness in intelligently steering traffic through a slice-aware scheme, improving eMBB data rate by an average of 99.42% over slice isolation.

5.1 Introduction

The sixth-generation (6G) wireless networks face a major challenge in supporting various services with different key performance indicators (KPIs) on a unified air interface, including enhanced mobile broadband (eMBB) and ultra-reliable low-latency communication (uRLLC), each with specific values for latency, reliability, and data rate [102]. The inflexibility of classic radio access network (RAN), characterized by a “one-size-fits-all” infrastructure on black-box hardware, challenges reconfigurability and on-demand adjustments without manual on-site intervention. This limitation hinders the accommodation of diverse services and competition on the same network functions [103]. The Open RAN architecture, with its attributes of flexibility, virtualization, disaggregation, openness, and intelligence, disrupts the traditional RAN approach, ushering in a transformative paradigm in *NextG* wireless networks [104].

Combining the above principles results in virtualized and intricate architectures that enhance RAN performance through programmable and intelligent layers. To improve interoperability among vendors, machine learning (ML) and artificial intelligence (AI) methods are embedded in the architecture. A pivotal aspect of Open RAN involves unlocking network intelligence using two novel modules: the non-real-time (non-RT) and near-real-time (near-RT) RAN intelligent controllers (RICs). These modules enable closed-loop RAN control via standardized interfaces, allowing data collection and sharing between different components while incorporating centralized network abstraction [105]. These lead to the effective acquisition of a deep understanding of intricate cross-layer interactions among components, surpassing typical control heuristics and advancing toward the achievement of optimal resource management. An essential Open RAN use case is traffic steering (TS), which distributes traffic load across various radio access technologies (RATs) in the RAN, which involves overseeing the mobility of individual user equipment (UE) being served by the RAN [106].

In multi-traffic scenarios, TS incorporates emerging fifth-generation (5G) technologies and procedures, including multi-connectivity (MC), network slicing (NS), and handover management. The availability of network data and analytics in centralized locations (*i.e.*, RICs) transforms the traditional handover management of the RAN architecture into an intelligent TS framework within the Open RAN paradigm. However, this shift also presents new challenges [107]. Traditional resource management schemes rely primarily on heuristic approaches that consider channel quality and load thresholds. However, these methods are less suitable for making user-centric handover decisions in novel use-case scenarios. They often rely on localized information, which limits their effectiveness [108]. On the contrary, data-driven solutions at the RIC provide a more centralized perspective, allowing them to discern intricate relationships among various RAN parameters. This enables customization of optimization strategies to meet the unique quality of service (QoS) requirements of individual users.

Efficiency technologies such as NS and MC are well-regarded for their effectiveness in achieving optimal traffic management and accommodating the diverse demands of multiple types of traffic [109], [110]. These techniques encompass a spectrum of strategies and functionalities for resource management and connectivity. This includes dynamic allocation of radio units (RUs) and resource blocks (RBs) to users based on real-time conditions and user-specific preferences. Furthermore, MC extends its capabilities to higher-layer options that play a crucial role in enhancing overall network performance and ensuring an improved user experience. These higher-layer functionalities encompass various aspects of network operation, such as efficient handover management, load balancing, and seamless service continuity, which collectively contribute to the success of MC in modern wireless networks. MC with multi-link improves reliability and data rate [83]. NS segregates multiple types of traffic into logical network slices while using the same infrastructure, enabling simultaneous transmission on one channel [111]. However, meeting the latency-critical requirements of the uRLLC service, especially with small packet sizes, poses a challenge beyond the 5G capabilities. A promising solution to address this challenge is the mini-slot-based frame structure introduced by the third-generation partnership project (3GPP) on the new radio (NR) to support transmissions shorter than the regular slot duration [112]. It subdivides each time slot into two mini-slots, each with 7 orthogonal frequency-division multiplexing (OFDM) symbols [113], reducing slot duration and transmission/frame alignment time.

5.1.1 Motivation and Main Contributions

Despite the traditional user handover mechanisms in the RAN architecture, which predominantly rely on localized and limited information, an intelligent TS scheme can steer traffic flow within the RAN components via data-driven solutions in the RICs. RICs leverage a centralized point of view of RAN components to optimize the QoS requirements of each user. This requires incorporating intelligence into each Open RAN layer. Therefore, our research aims to explore intelligence within the Open RAN architecture. To accomplish this, three essential concerns arise: *(i) Predicting future traffic demands based on historical data. (ii) Intelligently distribute traffic flows to steer traffic to end users through appropriate RUs given the predicted traffic demands and data collected from the RAN in RICs*; and *(iii) Efficiently schedule radio resources to accommodate heterogeneous traffic with different demands while meeting QoS requirements, power limitations, and practical constraints.*

To address these concerns, we employ recurrent neural network (RNN) long-short-term memory (LSTM) and novel multi-agent (MA) deep reinforcement learning (MADRL) techniques for traffic prediction and radio resource scheduling, respectively. LSTM handles high-dimensional and large-space problems [114], while overcoming long-term dependencies and gradient issues [87]. However, the DRL model excels in complicated algorithmic learning, extreme generalization, and dynamic wireless environments [114]. We will demonstrate how our scheme complies with Open RAN demands, implementing MADRL- and LSTM-based closed control loops within Open RAN. Besides, we will present simulation results illustrating the significant improvement in network performance. In summary, our key contributions are as follows.

- We introduce a novel framework that optimizes traffic prediction, flow-split distribution, congestion control, and scheduling for uRLLC and eMBB within an Open RAN architecture. Utilizing dynamic MC, slice-aware RAN slicing, and mixed numerology multiplexing, the framework achieves uRLLC latency below 0.5 ms, multiple hundred Mbps eMBB data rate, and congestion-free operation. Our focus remains on illustrating the benefits of MC in a controlled environment, recognizing that real-world 3GPP systems incorporate additional mechanisms for efficient and reliable data transmission. To this end, the objective function simultaneously minimizes the long-term average queue length of eMBB users (maximizing eMBB data rate) and the long-term average uRLLC latency, considering QoS requirements, slice awareness, power budget, and traffic flow-split decisions.

- We propose a comprehensive TS algorithm using novel DRL to address intricate challenges in decision-making per time slot. It handles incomplete traffic demand, queue length, and time-varying wireless channels, relying on previous RAN layer updates. This framework reduces computational complexity by making decisions per frame instead of per time slot and addresses incomplete channel state information (CSI) knowledge. We employ a two-stage optimization approach on different timescales, handled by RICs at higher Open RAN layers and distributed units (DUs) at the function Open RAN layer. The xAPP introduced at the near-RT RIC manages the long-term subproblem (frame structure), including traffic prediction, flow split distribution, and RB assignment, through the deployment of inferences from trained models at the non-RT RIC thanks to the data collected at service management and orchestration (SMO). Simultaneously, DUs handle the short-term subproblem (time slot structure) for power control and transmission power allocation with mixed numerologies via the closed-control loop between DUs and near-RT RIC following 5G NR standardization guidelines.

- Utilizing historical information and combining tools from the LSTM and MADRL frameworks, we develop simple yet efficient algorithms to make an empirical distribution of the system dynamics to facilitate predicting the traffic demand and learning RB assignment, respectively. Furthermore, we present a heuristic method that relies on the predicted traffic demand and historical data collected from the RAN components stored in the SMO. This method dynamically estimates the ideal RUs to direct traffic to the end user for each frame. The training of these models takes place offline within the non-RT RIC of the SMO module. Training data is gathered from the RAN components through the O1 interface. Subsequently, the standard solver is applied to solve the short-term subproblem categorized as the convex optimization programming class.

- Finally, we conduct an analysis of the effectiveness of our approach considering slice awareness by an extensive set of simulations, leading to a notable performance improvement of 99.42% compared to slice isolation in terms of data rate.

The subsequent sections of this chapter are as follows: Section 5.2 provides the related literature on TS in the traditional RAN and Open RAN paradigms. Section 5.3 studies a general overview of Open RAN's concept and architecture, while Section 5.4 introduces the system model. In Section 5.5, the optimization problem is formulated, along with an overview of the DRL-based intelligent TS algorithm and its structure. Section 5.6 discusses a mathematical analysis of the proposed framework, including the LSTM model, heuristic method, and the novel MADRL model to solve the long-term subproblem. Extensive numerical results comparing the proposed approach with benchmark schemes are presented in Section 5.7. Finally, Section 5.8 concludes the chapter, summarizing key findings and insights.

5.2 Related Works

5.2.1 RAN Radio Resource Management

Extensive research studies have explored dynamic RAN resource allocation mechanisms in the traditional cellular RAN architecture. For instance, [11] and [92] have proposed a joint scheduling design for uRLLC and eMBB coexistence to maximize the overall data rate of the eMBB service, while the uRLLC service is designed on the eMBB's dedicated resources to meet the uRLLC requirement latency. For wireless access-based NS, [115] has used a preemptive puncturing approach, assigning contiguous RBs to the uRLLC and eMBB services. In [91], a dynamic downlink multiplexing was proposed for uRLLC and eMBB services on the same radio spectrum, in which if the preserved resources are not sufficient for uRLLC, part of eMBB traffic overlaps with uRLLC traffic. The authors in [29] have proposed a dynamic MC-based joint scheduling scheme with traffic steering for eMBB and uRLLC services to achieve high data rates in 5G networks. However, the mentioned puncturing approaches lead to poor eMBB performance under high uRLLC traffic, necessitating joint scheduling of uRLLC and eMBB across distinct network slices to avoid sizable uRLLC service queues. In addition, in [116] a joint user association and scheduling optimization scheme was proposed to maximize overall network utilization in the cloud-RAN (C-RAN) architecture. Due to the coupling between optimization variables and the combinatorial nature, most of the mentioned problems have utilized heuristic methods or the difference of convex algorithms to deal with. However, different demands in beyond 5G wireless networks cannot be satisfied by only allocating power and subcarriers. However, these efforts do not focus on enhancing the performance of individual users and do not entirely meet the requirement for individual user control and optimization.

5.2.2 Traffic Steering in Traditional RAN Architecture

In the literature, there are some studies that have primarily investigated TS in traditional RAN networks. For example, [44] introduced a TS technique to maximize quality of experience (QoE) for eMBB users in the MC scenario. In [117], a proposal was presented to alter user association in overlapped ultra-dense networks. Furthermore, [118] learned user association policies to direct traffic from different locations to transmitters, minimizing total delay and load on the wireless downlink in the presence of unknown dynamic traffic demand. Network slicing is a key innovation to meet diverse needs. In [119], a RAN slicing-based radio resource allocation scheme was proposed for dynamic TS RAN moderation in 5G. Praveen *et. al.* [93] investigated the RAN resource slicing for uRLLC and eMBB in downlink orthogonal frequency-division multiple access (OFDMA) 5G networks to maximize the sum rate. All the above-mentioned studies focused on fixed numerology in classic RAN architecture for scheduling schemes. Recent studies explore mixed numerologies in time and frequency domains for resource allocation, catering to services with conflicting demands. For instance, [94] explored flexible numerologies in the frequency domain to enhance the capacity of services with nonuniform requirements. Experimental field tests in [120] demonstrated the effectiveness of multiplexing mixed numerologies in the frequency domain for the performance assessment of OFDM-based 5G waveforms. In [95], joint optimization of power allocation and resource block scheme was investigated to serve heterogeneous traffic with mixed numerology-based frame constructions. To mitigate inter-numerology interference (INI) and prevent resource wastage, [121] suggested selecting a single numerology per time slot based on service priorities in multi-numerology resource allocation. In [122], a RAN slicing solution was developed for 5G networks, allocating time-frequency resources with different numerologies to support different services. Moreover, these studies often overlook crucial factors such as routing, congestion control, dynamic traffic demands, and user-centric conditions, which can render the attainment of multi-layer QoS in Open RAN unfeasible. Therefore, it needs to investigate the TS considering user-centric conditions in a flexible and intelligent RAN architecture (*i.e.*, Open RAN). Very recently, our previous work [123] proposed a slice isolation RAN resource allocation with mixed numerologies in the Open RAN architecture to enable the TS scheme even with imperfectly known traffic demands aimed at maximizing eMBB data rate while minimizing uRLLC latency.

5.2.3 ML-powered Intelligent Traffic Steering in Open RAN Architecture

ML-based traditional handover schemes were widely investigated and optimized in the literature. In traditional RAN in [124], a model of actor-critic reinforcement learning (RL) jointly optimized the selection of communication modes, the RB, and the allocation of power in the internet-enabled device-to-device communication networks. [125] has proposed a unified self-management mechanism based on fuzzy logic and RL to tune the handover parameters of adjacent cells. In [126] and [127], deep Q learning algorithms have solved the coexistence of uRLLC and eMBB, achieving flexible time slot scheduling. However, these related works mainly focused on optimizing or predicting RB assignments per time slot in TS schemes, resulting in high complexity. Existing RICs in Open RAN architecture can benefit from a centralized point of view to steer the traffic in an efficient way to target the QoS of each user. To our knowledge, there are only a few works that study intelligent TS frameworks in the Open RAN architecture. For example, the authors of [50] have proposed a TS-xAPP at near-RT RIC combined with a convolutional neural network (CNN) architecture, to optimally assign a serving base station to each user in the Open RAN architecture. [67] has explored the current O-RAN specifications, providing experimental results of the Open RAN data-driven closed-loop in a large-scale testbed with programmable RAN components and RICs. In [86], concepts, requirements, and principles of Open RAN proposed by the O-RAN Alliance were introduced, along with a general example of the use case of intelligent radio resource management. In [49], a multi-layer optimization framework was proposed to steer traffic in the Open RAN architecture to maximize utility functions.

However, most of the existing efforts did not develop an intelligent TS scheme for multi-traffic downlink OFDMA 5G systems considering mixed numerologies in the presence of unknown traffic demands. To achieve fully automated networks with improved control and optimization, the development of a DRL-based TS framework becomes crucial, supporting heterogeneous services and adapting to the dynamic wireless environment. To this end, we propose a multi-layer optimization framework interaction between the cell site and the higher layers, facilitating the system performance utilizing the closed-control loops between RICs and RAN components in the Open RAN paradigm. Thanks to the holistic perspective

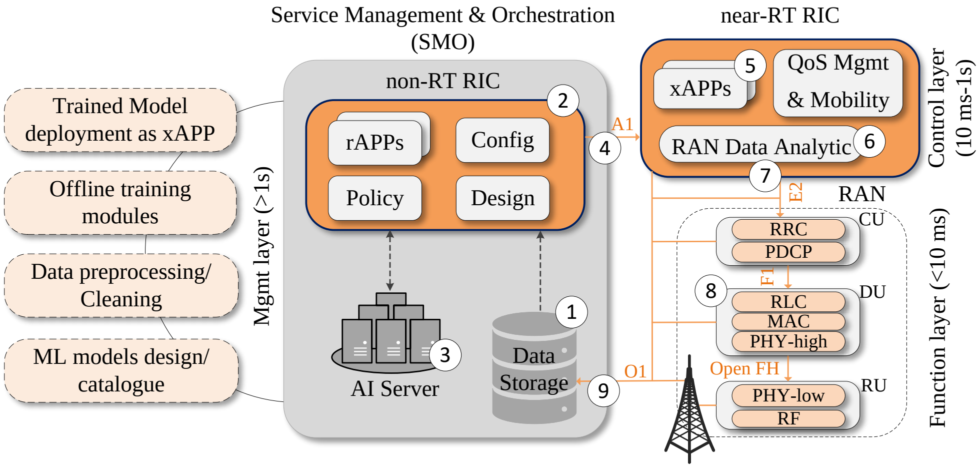


FIGURE 5.1: Learning integration in Open RAN architecture based on the O-RAN Alliance.

of RICs, which allows them to consider factors such as traffic loads, user demands, queue length, channel conditions, *etc.*, our proposed intelligent TS scheme can be centrally coordinated to achieve the required QoS for each user in dynamic wireless environments. This paves the way for fully automated networks with enhanced control and flexibility.

5.3 Intelligent TS Deployment on Open RAN architecture

5.3.1 Open RAN Background

Fig. 5.1 shows the learning-based Open RAN architecture relying on the O-RAN Alliance [96] with three layers: management, control, and function. To simplify human-machine interaction and network complexity, the O-RAN Alliance establishes two novelty modules (*i.e.*, near-RT and non-RT RICs) at higher layers for centralized network abstraction [128]. These components enhance RAN optimization by feedback and action loops within RAN elements (E2 nodes) and RICs. These modules enable mobile operators to effectively deploy and manage their Open RAN networks, ensuring interoperability with various vendors, seamless handovers between cells, intelligent resource allocation, interference mitigation, and balanced load distribution. The Open RAN architecture facilitates various networking procedures at multiple network points by boosting and supporting the 3GPP functional split. This split virtualizes BS functionalities as network functions distributed across various network nodes, including the RU, DU, and centralized unit (CU) [68]. In addition, open interfaces (F1, E1, E2, FH, O1, A1) allow connections for disaggregated deployments, ensuring efficient multi-vendor interoperability and enabling network operators to choose RAN elements from different vendors independently.

According to the Open RAN architecture described in our previous work [123], the system model's layers operate at different timescales, ranging from 1 to thousands of milliseconds. Non-RT RIC handles activities such as service provisioning, design, policy definition, and AI/ML model training at intervals greater than 1 second, and hosts remote applications (referred to as rAPPs). However, the near-RT RIC heads tasks with timescales surpassing 10 milliseconds, introduces intelligence into the RAN via data-driven control loops, and hosts external applications known as xAPPs, enabling the programmability of RAN components. Meanwhile, near-RT RIC is responsible for tasks such as real-time traffic and radio monitoring, QoS control, storing and upkeep of historical traffic demands, handover management, and collaboration with non-RT RIC. Non-RT and near-RT RICs, situated in the cloud's higher layer, connect through A1 and O1 interfaces. Meanwhile, CU and DU on the edge cloud link via the F1 interface, managed by the near-RT RIC through the E2 interface. At the cell site, the RU is installed and managed by the DU via the open fronthaul (FH) interface. Finally, for periodic reporting, CU, DU, and RU interact with the non-RT RIC via the O1 interface.

5.3.2 Deployment of Intelligent Traffic Steering

We outline the end-to-end (e2e) flow of the intelligent TS deployment within the Open RAN architecture inspired by the O-RAN Alliance and 3GPP specifications (Fig. 5.1).

① **Data pre-processing and cleaning:** Collect data from the near-RT RIC and the RAN components, including user traffic demands, CSI, resource updates, network condition data, *etc.*, periodically over the O1 interface. In addition, data pre-processing and transferring data into a format that is suitable for ML algorithms are done in this step.

② **AI/ML model query:** The related ML/AI model, hosted on the AI server inside the SMO, is queried by non-RT RIC. The non-RT RIC queries the AI/ML model from the SMO's AI server to apply to some consecutive rAPPs situated at the non-RT RIC.

③ **Model training:** Once the model is trained offline on the AI server, the inferences are sent back to the non-RT RIC.

④ **Inference transferring:** Policies and inference results—all trained ML models—are forwarded to the near-RT RIC through the A1 interface for making long-term decisions, including handover management, load balance, cell congestion, and radio resource management.

⑤ **Intelligent TS xAPP Deployment:** All inferences are deployed in an intelligent TS xAPP in the near-RT RIC to make RAN components programmable.

⑥ **RAN data analytics:** Afterwards, the RAN data analytic component in the near-RT RIC updates the queue lengths based on the data/metrics reported from the RAN components over the E2 interface (E2SM-KPM).[1]

[1] In O-RAN, E2 Service Models (E2SMs) play a crucial role in defining communication and management protocols between network functions. Two key E2SMs are E2SM-KPM (Key Performance Management) and E2SM-RC (Radio Control), addressing performance monitoring and radio resource management, respectively.

⑦ **RAN controlling:** Given the actions and policies of the upper layers, the RAN control (E2SM-RC)[1] is sent to the RAN components for execution via the E2 interface. The near-RT RIC continuously monitors the performance of the intelligent TS scheme at cell sites.

⑧ **Power adjusting:** DU adjusts power levels based on exchange of performance metrics, actual traffic demands, local observations, *etc.*, with near-RT RIC through the E2 interface and receiving control actions. Furthermore, DU is not only responsible for optimizing power allocation on a time-slot basis to deal with dynamic environments but is also in charge of buffer management.

⑨ **Continuous monitoring:** Finally, all updated information, observations (*i.e.*, network conditions, CSI, traffic demands, queue lengths, states, *etc.*), and performance metrics (*i.e.*, data rate, latency, and reward values) are reported to the SMO through the O1 interface on the effectiveness of the intelligent TS scheme. SMO continuously monitors the network and triggers retraining of ML models and xAPPs in response to congestion issues, inaccurate traffic predictions, or degraded user QoE.

5.4 System Model

We consider a downlink OFDMA multiuser multi-input single-output (MU-MISO) system, which consists of one CU, N DUs, and M RUs (see Fig. 5.1 with the RAN part). Toward cost-effectiveness, each DU forms a cluster of RUs. Let $\mathscr{N} \triangleq \{1,\dots,N\}$ and $\mathscr{M} \triangleq \{1,\dots,M\}$ denote the set of DUs and RUs, respectively ($M >> N$). Each RU is equipped with K_{tx} antennas to serve a set of U single-antenna users $\mathscr{U} \triangleq \{1,\dots,U\}$ through the shared wireless medium. To deploy the coexistence of uRLLC and eMBB, we divide the set of users into two disjoint sets: $\mathscr{U}^{\mathsf{ur}} \triangleq \{1,\dots,U^{\mathsf{ur}}\}$ of uRLLC users and $\mathscr{U}^{\mathsf{em}} \triangleq \{1,\dots,U^{\mathsf{em}}\}$ of eMBB users, with $\mathscr{U} \triangleq \mathscr{U}^{\mathsf{ur}} \cup \mathscr{U}^{\mathsf{em}}$. The traffic of eMBB users is generated with a large packet size of Z^{em} bytes, while that of uRLLC users is a sequence of small and identical packet sizes of Z^{ur} bytes ($Z^{\mathsf{em}} >> Z^{\mathsf{ur}}$). Long packet traffic requires much longer to transmit than short packet traffic. If each RU serves only one type of traffic at a time, uRLLC users may face significant delays in meeting their low-latency demands. In the context of MC, RUs can transmit multiple types of traffic in different frequency bands, making the links between RUs and UEs more flexible than traditional methods. Moreover, traffic can be sliced and transmitted on independent links based on the MC configuration. For simplicity, we assume that DUs cover the non-overlapped geographical area with a disjoint set of RUs, such as $\mathscr{M}_n \triangleq \{(n,1),\dots,(n,M_n)\}$ with $\sum_{n\in\mathscr{N}} M_n = M$.

5.4.1 Frequency-time-frame Numerologies

To efficiently cater to all users within the cell, RUs allocate RBs (frequency-time in OFDMA) while optimizing transmission power for each RB. The details of frequency-time-frame numerologies in three different modes are elaborated below.

Fixed numerology: In this mode (indexed as $i = 0$) for the upcoming 5G NR systems, each RB consists of 7 OFDM symbols per transmission time interval (TTI) of 0.5 ms. It

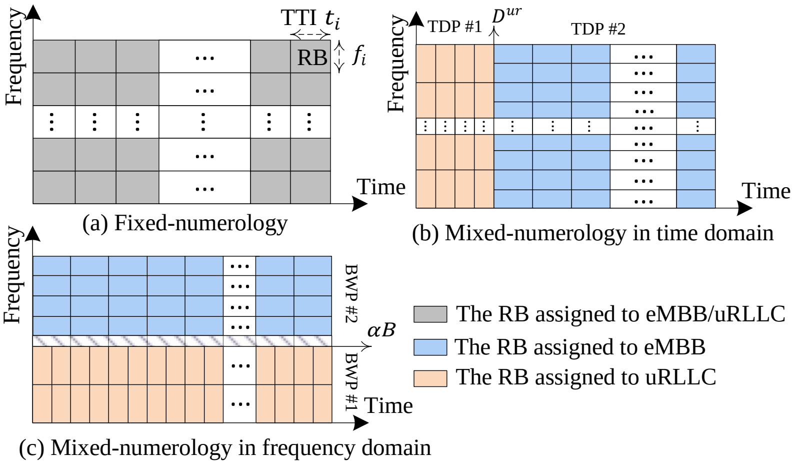


FIGURE 5.2: Illustration of fixed numerology and mixed numerologies in frequency and time domains.

comprises 12 consecutive subcarriers with a subcarrier spacing (SCS) of 15 kHz. Let β_i and δ_i denote the RB's bandwidth and time duration at the t-th frame with a large-scale coherence time of Δ (*i.e.,* 10 ms). The frequency-time resource grid consists of $F_i \times T_i$ RBs, as shown in Fig. 5.2(a). Given the available system bandwidth (BW), denoted by B, we have $F_i|_{i=0} = \lfloor B/\beta_i \rfloor$ and $T_i|_{i=0} = \Delta/\delta_i$ indicating the number of subcarriers indexed as $f_i = \{1, \ldots, F_i\}$ and the number of TTIs per frame indexed as $t_i = \{1, \ldots, T_i\}$, respectively. **Mixed numerology multiplexing in time domain:** Unlike 4G-LTE, 5G wireless systems use scalable numerologies to address the QoS demands associated with different types of traffic. This entails dividing each frame into multiple time duration parts (TDPs), each adopting a specific numerology tailored to meet the QoS demands of the corresponding assigned service slice. While this mode reduces spectrum waste and INI through the utilization of guard bands, it may introduce intermittent temporal gaps between numerologies, potentially hindering latency-sensitive applications' efficiency. Therefore, to meet the stringent latency requirements of uRLLC services, which are of utmost priority in our case, the time division of the frame is structured in such a manner that the first part of the time horizon is specifically allocated to uRLLC services, as illustrated in Fig. 5.2(b). Here, D^{ur} represents the minimum latency requirement for uRLLC traffic.

Regarding the numerology specifications, the system BW is divided into subcarriers, and the TDP is further divided into multiple TTIs (mini-slots), maintaining orthogonality between consecutive RBs. According to the findings in [97], it can be intuitively perceived how different numerologies can be used to meet the demands of each 5G service class. Accordingly, intermediate numerologies with index $i = 1$ are well-suited for eMBB, which demands higher data rates and significant bandwidth. In contrast, higher numerologies with index $i = 2$ are better suited for uRLLC services, particularly for applications with stringent latency requirements, as they involve the transmission of short bursts of data packets. Therefore, the scheduled slices assigned to uRLLC with numerology $i = 2$ (with $\beta_2 = 720$ KHz and $\delta_2 = 0.125$ ms) and eMBB with $i = 1$ (with $\beta_1 = 360$ KHz and $\delta_1 = 0.25$ ms) are indicated by $T_i|_{i=2} = D^{\mathsf{ur}}/\delta_i$ and $T_i|_{i=1} = (\Delta - D^{\mathsf{ur}})/\delta_i$. Note that $F_i|_{i=2} = \lfloor B/\beta_i \rfloor$ and $F_i|_{i=1} = \lfloor B/\beta_i \rfloor$ represent the scheduled number of sub-bands assigned to the uRLLC and eMBB slices in this mode, respectively.

Mixed numerology multiplexing in frequency domain: Multiple services can be served in the frequency domain by dividing the BW into multiple bandwidth parts (BWPs) (Fig. 5.2(c)). The min-slot-based framework and uRLLC sliced from eMBB cater to critical latency services and prevent uRLLC queuing. To this end, the BW-split variable $\alpha \in [0, 1]$ splits the available BW into two independent BWPs to handle dynamic service demands. A fixed guard band of 180 kHz (B_G) is implemented between neighboring numerologies to reduce INI within adjacent sub-bands. The scheduled BWP assigned to the uRLLC slice with numerology $i = 2$ with the RB's BW of $\beta_i|_{i=2} = 720$ kHz and $\delta_i|_{i=2} = 0.125$ ms of TTI duration is given as $B_i|_{i=2} = \alpha B$. Next, $B_i|_{i=1} = (1 - \alpha)B - B_G$ is the scheduled BWP assigned to the eMBB slice with numerology $i = 1$ and RB's BW of $\beta_i|_{i=1} = 360$ kHz and $\delta_i|_{i=1} = 0.25$ ms of TTI duration. Hence, it follows that $F_i = \lfloor B_i/\beta_i \rfloor$ and $T_i = \Delta/\delta_i$.

5.4.2 Transmission Model and Downlink data rate

In Fig. 5.1, U independent data flows at CU are steered to DUs for parallel processing. The processing queue follows the $M/M/1$ model, serving packets on a first-come, first-served basis. Following RIC policies based on MC, CU divides the data flow of each user u into sub-flows transmitted through a maximum of M_n paths and aggregated at the user. The careful selection of a subset of distinct paths for each data flow u is crucial to optimizing the system performance. In the discrete time frame indexed by $t \in \{1, 2, \ldots, T\}$, we define $\boldsymbol{x}[t] \triangleq \big[x_{m,u}[t]\big]^T_{\forall m,u}$ as the flow-split indicator vector. In particular, if $x_{m,u}[t] = 1$, RU m is selected to transmit data of u-th data-flow; otherwise, $x_{m,u}[t] = 0$. Let us denote by $\boldsymbol{\Psi}[t] \triangleq \big\{\boldsymbol{\varphi}_u[t], \forall u \big| \sum_{m \in \mathcal{M}_n} \varphi_{m,u}[t] = 1,\ \varphi_{m,u}[t] \in [0, 1], \forall m, u\big\}$ the global flow-split decision, in which $\boldsymbol{\varphi}_u[t] \triangleq \big[\varphi_{m,u}[t]\big]^T_{\forall m}$ represents the flow-split portion vector of the user u. It is noted that $\sum_{m \in \mathcal{M}_n} \varphi_{m,u}[t] = 1$, where $\varphi_{m,u}[t] \in [0, 1]$ indicates a portion of the data flow steered to the user u via RU m in time-frame t by selecting flow-split indicator $x_{m,u}[t]$.

The channel vector between RU m and user u per RB(f_i, t_i) associated with sub-band f_i in TTI t_i can be modeled as $\boldsymbol{h}^{f_i,t_i}_{m,u} = \sqrt{10^{-\mathsf{PL}_{m,u}/10}}\bar{\boldsymbol{h}}^{f_i,t_i}_{m,u}$, where $\mathsf{PL}_{m,u}$ is the path loss between RU m and user u, and $\bar{\boldsymbol{h}}^{f_i,t_i}_{m,u} \in \mathbb{C}^{1 \times K_{\text{tx}}}$ represents the circularly symmetric complex Gaussian random variables with zero means and unit variances. We consider the correlated channel during the whole available frames as $\boldsymbol{h}^{f_i,t_i}_{m,u} = \sqrt{\rho}\boldsymbol{h}^{f_i,t_i-1}_{m,u} + \sqrt{1 - \rho}Z$, where $Z \sim \mathcal{N}(0, N_0)$ is zero-mean additive white Gaussian noise (AWGN). Without multi-user interference, the maximum-ratio transmission is an optimal transmission scheme. Let us denote $\boldsymbol{G}[t] \triangleq \big[g^{f_i,t_i}_{m,u}\big]|_{\forall m,u,f_i}$ as the channel gain between all RUs' RBs to all users in frame t, where $g^{f_i,t_i}_{m,u} \triangleq \|\boldsymbol{h}^{f_i,t_i}_{m,u}\|^2_2$ is the effective channel gain. We use the binary variable $\pi^{f_i,t_i}_{m,u}[t] \in \{0, 1\}$ to indicate whether RB(f_i, t_i) of the m-th RU is allocated to the user u-th eMBB/uRLLC. To satisfy the orthogonality constraint, RB(f_i, t_i) of RU m allocated to the u-th generic user if $\pi^{f_i,t_i}_{m,u}[t] = 1$; otherwise $\pi^{f_i,t_i}_{m,u}[t] = 0$. $\boldsymbol{\pi}_{m,u}[t] \triangleq \big[\pi^{f_i,t_i}_{m,u}[t]\big]^T$ is denoted as the assigned RBs to the u-th generic user at the m-th RU in frame t. This chapter defines the RB matrices $\boldsymbol{\pi}[t] \triangleq \left[\boldsymbol{\pi}_{m,u}[t]\right]^{T}_{\forall m,u}$ for both services in frame t.

Following the Shannon-Hartley theorem, the downlink data rate of the u-th eMBB user [bits/s] at TTI t_i is given as

$$R^{\mathsf{em}}_{u}(\boldsymbol{p}[t_i]) = \sum_{m=1}^{M_n} R^{\mathsf{em}}_{m,u}(\boldsymbol{p}[t_i]) = \sum_{i\in\{1,2\}} \sum_{m=1}^{M_n} \sum_{f_i=1}^{F_i} \beta_i \times \log_2\left(1+\frac{p^{f_i,t_i}_{m,u} g^{f_i,t_i}_{m,u}}{N_0}\right), \forall t_i, u \in \mathcal{U}^{\mathsf{em}}, \quad (5.1)$$

where N_0 and $p^{f_i,t_i}_{m,u}$ are the noise power and the transmission power from RU m to user u, respectively. Next, we consider the scheduling constraint $0 \leq p^{f_i,t_i}_{m,u} \leq \pi^{f_i,t_i}_{m,u}[t] P^{\mathsf{max}}_{m}; \forall u \in \mathcal{U}^{\mathsf{em}}$ to ensure that if $\pi^{f_i,t_i}_{m,u}[t] = 0$, then $p^{f_i,t_i}_{m,u} = 0$, where P^{max}_{m} is the maximum available transmission power of RU m. Let $\boldsymbol{p}[t_i] \triangleq [p^{f_i,t_i}_{m,u}]|_{\forall m,u,f_i}$. To satisfy the QoS of the eMBB traffic, we impose the constraint $\sum_{i\in\{1,2\}} R^{\mathsf{em}}_{m,u}(\boldsymbol{p}[t_i]) \geq \varphi_{m,u}[t] R^{\mathsf{th}}$ for each eMBB user, where R^{th} is a given QoS threshold. Next, the maximum channel coding rate that the uRLLC user u may achieve at time t_i with a certain block-length and error probability is provided roughly as

$$R^{\mathsf{ur}}_{u}(\boldsymbol{p}[t_i]) = \sum_{m=1}^{M_n} R^{\mathsf{ur}}_{m,u}(\boldsymbol{p}[t_i]) = \sum_{i\in\{1,2\}} \sum_{m=1}^{M_n} \sum_{f_i=1}^{F_i} \beta_i \times \Bigg[\log_2\left(1+\frac{p^{f_i,t_i}_{m,u} g^{f_i,t_i}_{m,u}}{N_0}\right) - \log_2(e) \times \frac{\pi^{f_i,t_i}_{m,u}[t]\sqrt{V} Q^{-1}(P_e)}{\sqrt{\delta_i \beta_i}}\Bigg], \quad (5.2)$$

where V, $Q^{-1}: \{0,1\} \to \mathbb{R}$, and P_e are the channel dispersion, the inverse of the Gaussian Q-function, and error probability, respectively. Since achieving an signal-to-noise (SNR) ($\Gamma^{\mathsf{ur}} = \frac{p^{f_i,t_i}_{m,u} g^{f_i,t_i}_{m,u}}{N_0}$) higher than 5 dB in the cellular network is highly obtainable, in this chapter we approximate $V = 1 - \frac{1}{1+(\Gamma^{\mathsf{ur}})^2} \approx 1$ [77]. To meet the Big-M formulation theory and the approximation $V \approx 1$, we impose the constraint $\frac{N_0 \Gamma^{\mathsf{ur}}}{g^{f_i,t_i}_{m,u}} \pi^{f_i,t_i}_{m,u}[t] \leq p^{f_i,t_i}_{m,u} \leq \pi^{f_i,t_i}_{m,u}[t] P^{\mathsf{max}}_{m}; \forall u \in \mathcal{U}^{\mathsf{ur}}$. Besides, the constraint $\sum_i \sum_{f_i,u} p^{f_i,t_i}_{m,u} \leq P^{\mathsf{max}}_{m}, \forall u \in \mathcal{U}$ guarantees that the total transmission power is no larger than each RU power budget, P^{max}_{m}. Accordingly, the power constraint associated with both services (*i.e.*, eMBB and uRLLC) is defined as

$$\mathscr{P}[t_i] = \Bigg\{ \boldsymbol{p}[t_i], \forall t_i \Bigg| 0 \leq p^{f_i,t_i}_{m,u} \leq \pi^{f_i,t_i}_{m,u}[t] P^{\mathsf{max}}_{m}; \forall u \in \mathcal{U}^{\mathsf{em}}, \frac{N_0 \Gamma^{\mathsf{ur}}}{g^{f_i,t_i}_{m,u}} \pi^{f_i,t_i}_{m,u}[t] \leq p^{f_i,t_i}_{m,u} \leq \pi^{f_i,t_i}_{m,u}[t] P^{\mathsf{max}}_{m} ; \forall u \in \mathcal{U}^{\mathsf{ur}}, \sum_{i\in\{1,2\}} \sum_{f_i,u} p^{f_i,t_i}_{m,u} \leq P^{\mathsf{max}}_{m}, \forall m, t_i, u \in \mathcal{U} \Bigg\}. \quad (5.3)$$

5.4.3 Slice-aware RB Allocation

To efficiently exploit radio resources, especially under low traffic demands, we propose a slice-aware strategy instead of the isolated slice-based radio resource allocation method. In this approach, the slices share the available radio resources while adhering to specific constraints. This strategy allows traffic to be assigned to a specific numerology, but it also permits access to other numerologies, known as “slice awareness”. When there is increased traffic demand for uRLLC, additional RBs are allocated from the eMBB slice. Conversely, to enhance eMBB’s data rate, this service may access the underused resources of the uRLLC slice. While the slice-aware method is more complex to create and operate than the slice-isolation method, it improves resource utilization by dynamically distributing resources based on service traffic arrivals.

Let $\lambda_u^{\mathsf{x}}[t]$ [packets/frame] denote the (unknown) traffic demand of the u-th generic user at the time-frame t with $\mathsf{x} \in \{\mathsf{ur}, \mathsf{em}\}$. We assume that $\lambda_u^{\mathsf{x}}[t]$ follows the Poisson distribution with the mean arrival rate $\mathbb{E}\{\lambda_u^{\mathsf{x}}[t]\} = \bar{\lambda}_u^{\mathsf{x}}$, where the size of each packet is identical and equal to Z^{x}. To respond to the priority of the scheduled uRLLC service, we consider the following constraint.

$$\sum_{t_i=1}^{D^{\mathsf{ur}}/\delta_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i}[t] \geq e_u^{\mathsf{ur}}[t], \ \forall t, u \in \mathscr{U}^{\mathsf{ur}}, i = 1, \tag{5.4}$$

where $e_u^{\mathsf{ur}}[t] = \big\lceil \max\big(\lambda_u^{\mathsf{ur}}[t] - \Omega_u[t], 0\big)/2 \big\rceil$ is the slice awareness for the mixed numerologies in the frequency domain. Herein, $\lambda_u^{\mathsf{ur}}[t]$ and $\Omega_u[t] = \frac{\lambda_u^{\mathsf{ur}}[t]}{\sum_u \lambda_u^{\mathsf{ur}}[t]}(F_i \times D^{\mathsf{ur}}/\delta_i)|_{i=2}$ are the number of available packets in the queue of the u-th uRLLC user and the maximum number of RBs for each uRLLC user in a dedicated slice of uRLLC per frame. In contrast, the following constraint allocates all underused RBs of the other slices to eMBB users to increase the data rate given

$$\sum_{t_i=1}^{T_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i}[t] \geq e_u^{\mathsf{em}}[t], \ \forall t, u \in \mathscr{U}^{\mathsf{em}}, i = 2, \tag{5.5}$$

where $e_u^{\mathsf{em}}[t] = \max\Big(\lfloor[(F_i \times T_i) - \sum_{u \in \mathscr{U}^{\mathsf{ur}}} \min(\lambda_u^{\mathsf{ur}}[t], \Omega_u[t])]/U^{\mathsf{em}}\rfloor, 0\Big)|_{i=2}$ is the slice awareness for the mixed numerologies in the frequency and time domain. Herein, $\min(\lambda_u^{\mathsf{ur}}[t], \Omega_u[t])$ represents the number of RBs scheduled for the uRLLC user in the t-th frame. Furthermore, we impose this constraint $\sum_{m,u} \pi_{m,u}^{f_i,t_i}[t] \leq 1; \forall f_i, t_i$ to guarantee the orthogonality restriction of OFDMA systems, *i.e.,* each RB can only be allotted to a single user. As a result, we define the set of RB allocation constraints as

$$\boldsymbol{\Lambda}[t] = \Big\{ \boldsymbol{\pi}[t], \forall t \Big| \pi_{m,u}^{f_i,t_i}[t] \in \{0,1\}, \sum_{m,u} \pi_{m,u}^{f_i,t_i}[t] \leq 1, \forall f_i, t_i; \sum_{t_i=1}^{D^{\mathsf{ur}}/\delta_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i}[t] \geq e_u^{\mathsf{ur}}[t], \forall t,$$
$$u \in \mathscr{U}^{\mathsf{ur}}, i=1; \sum_{t_i=1}^{T_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i}[t] \geq e_u^{\mathsf{em}}[t], \forall t, u \in \mathscr{U}^{\mathsf{em}}, i=2 \Big\}. \quad (5.6)$$

5.4.4 Network Queues and e2e Latency

We denote $\boldsymbol{\lambda}[t] \triangleq \big[\lambda_u^{\mathsf{x}}[t]\big]_{\forall u,\mathsf{x}}$ as the total arrival packet rate. We assume that the buffers are embedded at DUs—*i.e.*, radio link control (RLC)—to store the data arriving in a distinct queue for each user assigned to a given RU based on the predicted flow-split distribution, which should not exceed a finite constant $\lambda^{\mathsf{max}} < \infty$. The queue lengths for updating and controlling the congestion cell are done at the near-RT RIC through the information feedback over the closed-control loop between DU and near-RT RIC based on E2SM-KPM. It has played a pivotal role in enhancing network performance and adaptability to ensure that our network can meet the diverse and dynamic requirements of the Open RAN framework. The queue length [bits] of the generic data flow u at RU m is computed as

$$q_{m,u}[t_i] = \max\big(q_{m,u}[t_i - 1] + \varphi_{m,u}[t]\lambda_u^{\mathsf{x}}[t]Z^{\mathsf{x}}\Delta - R_{m,u}^{\mathsf{x}}(\boldsymbol{p}[t_i])\delta_i, 0\big) \quad (5.7)$$

where $\varphi_{m,u}[t]\lambda_u^{\mathsf{x}}[t]Z^{\mathsf{x}}$ (bits/frame) is the sub-flow of user u at RU m. In order to avoid congestion and packet loss caused by buffer overflow in each RU, the constraint $\sum_u q_{m,u}[t_i] \leq q_m^{\mathsf{max}}$, $\forall m$ is imposed to ensure that the available packets in the RU buffer should not exceed the maximum buffer size of q_m^{max}. Let us define $\boldsymbol{q}[t_i] \triangleq \big[q_{m,u}[t_i]\big]_{\forall m,u}^T$. At each frame, the proposed model predicts data arrivals and flow-split decisions based on analyzing the queues' status and provides the RBs' allocation accordingly.

The uRLLC e2e latency of user u at time-frame t can be given by

$$\tau_u^{\mathsf{ur}}[t] = \tau_{cu}^{\mathsf{pro}}[t] + \tau_{cu,du}^{\mathsf{tx}}[t] + \tau_{du}^{\mathsf{pro}}[t] + \tau_{du,ru}^{\mathsf{tx}}[t] + \tau_{ru,u}^{\mathsf{tx}}[t] + \tau_{ru}^{\mathsf{pro}}[t], \ \forall u \in \mathscr{U}^{\mathsf{ur}} \quad (5.8)$$

where $\tau_{cu}^{\mathsf{pro}}[t]$, $\tau_{du}^{\mathsf{pro}}[t]$, $\tau_{ru}^{\mathsf{pro}}[t]$, $\tau_{cu,du}^{\mathsf{tx}}[t]$, $\tau_{du,ru}^{\mathsf{tx}}[t]$ and $\tau_{ru,u}^{\mathsf{tx}}[t]$ represent the CU process time, DU process time, RU process time, transmission latency under the midhaul (MH), FH, and RU-user links, respectively. In addition, we define $\tau_{cu}^{\mathsf{pro}}[t] = \sum_u \lambda_u[t]/\mu_{cu}$ and $\tau_{du}^{\mathsf{pro}}[t] = \sum_u \lambda_u[t]/\mu_{du}$, where μ_{cu} and μ_{du} are the task rates [1/sec] at the CU and the DU, respectively; and $\tau_{ru}^{\mathsf{pro}}[t]$ is limited by the duration of the three OFDM symbols, which is commonly very small and refers to the equipment's computing capacity [123]. Since eMBB and uRLLC traffic are served in different slices, the eMBB queue does not affect the uRLLC queue. uRLLC users have higher priority and are served immediately upon arrival due to their stringent requirements and small data packet size. Thus, the main factor affecting uRLLC latency is the RU-user transmission time, which is calculated by the gap (measured in TTIs) between the time the specific uRLLC packet enters the queue and the time it is scheduled and leaves the queue, denoted as $\tau_{ru,u}^{\mathsf{tx}}[t] = \delta_i.\operatorname{argmax}_{t_i}\{\pi_{m,u}^{f_i,t_i}[t]\}$ for $u \in \mathscr{U}^{\mathsf{ur}}$. To ensure a minimum latency requirement for the u-th uRLLC user, the e2e latency is bound by a predefined threshold D_u^{ur}, *i.e.*, $\tau_u^{\mathsf{ur}}[t] \approx \tau_{ru,u}^{\mathsf{tx}}[t] \leq D^{\mathsf{ur}}$.

5.5 Problem Formulation

Utility Function: This chapter addresses key questions: *how to slice RAN resources, optimize the distribution of data flows, and allocate resources (subcarrier, power) under diverse QoS requirements of eMBB and uRLLC users in the presence of unknown traffic demands and time-varying channels.* We propose an intelligent TS scheme optimizing traffic demand prediction, flow-split distribution, and scheduling. To do so, the utility function considers both the eMBB data rate and the worst-user e2e uRLLC latency at the same time, as follows.

Let $\bar{q}_u \triangleq \lim_{t_i \to \infty} \frac{1}{t_i} \sum_{\tau=1}^{t_i} \sum_m q_{m,u}[\tau]$, $\forall u \in \mathcal{U}^{\mathsf{em}}$ be the long-term average queue length of the u-th eMBB data flow. It is clear that the shorter queue lengths lead to a higher eMBB data rate. Besides, controlling congestion to avoid large queues is crucial, especially for eMBB traffic with large packet sizes. This chapter aims to minimize the overall queue length for eMBB users and worst-case latency for uRLLC users. In particular, the objective function is given as $\omega \sum_{u \in \mathscr{U}^{\mathsf{em}}} \frac{\bar{q}_u}{q_0} + (1-\omega) \max_{u \in \mathscr{U}^{\mathsf{ur}}} \frac{\delta_i . \mathbb{E}_t \left(\operatorname{argmax}_{t_i} \{ \pi_{m,u}^{f_i,t_i}[t] \} \right)}{\tau_0}$, where $\omega \in [0,1]$ and $\mathbb{E}_t(.)$ denote the regulatory factor to control the influence of queue length and latency and the expectation function over time-frame t, respectively. In addition, $q_0 > 0$ and $\tau_0 > 0$ are the reference queue length of eMBB and the latency of uRLLC, respectively. These parameters are used to balance the two different dimensions of the two quantities. Based on the above definitions, the joint optimization problem of traffic demand prediction, flow-split distribution, congestion control, and scheduling is mathematically formulated as

$$\min_{\boldsymbol{\lambda}, \boldsymbol{\varphi}, \boldsymbol{\pi}, \boldsymbol{p}} \quad \omega \sum_{u \in \mathscr{U}^{\mathsf{em}}} \frac{\bar{q}_u}{q_0} + (1-\omega) \frac{\bar{\tau}^{\mathsf{ur}}}{\tau_0} \tag{5.9a}$$

$$\text{s.t.} \quad \boldsymbol{\pi}[t] \in \boldsymbol{\Lambda}[t], \ \forall t \tag{5.9b}$$

$$\boldsymbol{p}[t_i] \in \mathscr{P}[t_i], \forall t_i \tag{5.9c}$$

$$\boldsymbol{\varphi}[t] \in \boldsymbol{\Psi}[t], \ \forall t \tag{5.9d}$$

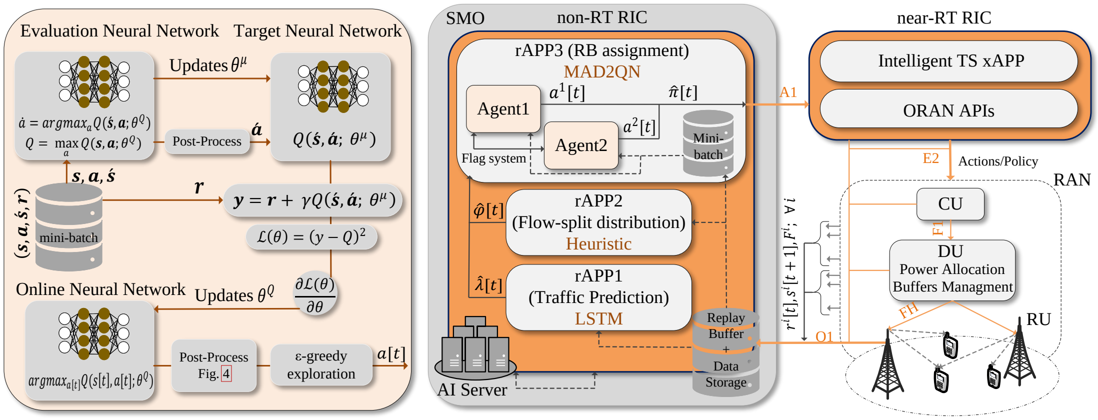
$$\sum_i R_{m,u}^{\mathsf{em}}(\boldsymbol{p}[t_i]) \geq \varphi_{m,u}[t] R^{\mathsf{th}}, \ \forall m \in \mathscr{M}_n, u \in \mathscr{U}^{\mathsf{em}} \tag{5.9e}$$

$$\Gamma^{\mathsf{ur}}(\boldsymbol{\pi}[t], \boldsymbol{p}[t_i]) \geq \pi_{m,u}^{f_i,t_i}[t] \Gamma^{\mathsf{th}} \tag{5.9f}$$

$$\tau_u^{\mathsf{ur}}(\boldsymbol{\pi}[t]) \leq D^{\mathsf{ur}}, \ \forall u \in \mathscr{U}^{\mathsf{ur}} \tag{5.9g}$$

$$\sum_u q_{m,u}[t_i] \leq q^{\mathsf{max}}, \ \forall t_i, m \in \mathscr{M}_n, u \in \mathscr{U} \tag{5.9h}$$

where $\bar{\tau}^{\mathsf{ur}} \triangleq \max_{u \in \mathscr{U}^{\mathsf{ur}}} \mathbb{E}_t \big(\delta_i . \operatorname{argmax}_{t_i} \{ \pi_{m,u}^{f_i,t_i}[t] \} \big)$. While $\boldsymbol{\varphi}[t], \boldsymbol{\pi}[t]$ and $\boldsymbol{p}[t_i]$ represent the vectors encompassing the flow-split portions, sub-band assignments and power allocation vectors at frame t and TTI t_i, respectively. The constraint (5.9f) guarantees that the SNR of each RB assigned to the user u-th uRLLC via m-th RU must be greater than $\pi_{m,u}^{f_i,t_i}[t]\Gamma^{\mathsf{th}}$,



(a) Structure of Double DQN (D2QN) agent

(b) The streamlined workflow for deploying ML applications in the Open RAN architecture.

FIGURE 5.3: A comprehensive workflow for intelligent TS deployment using ML application in Open RAN architecture.

where Γ^{th} represents the given SNR threshold for each RB assigned to the uRLLC service.

Challenges of solving problem (5.9): The non-convexity of constraints (5.9f) and (5.9h) and the binary nature of the RB assignment variables in constraint (5.9b) make problem (5.9) NP-hard. Additionally, the stochastic nature of the expected objective function further complicates the direct solution. Although existing optimization solvers (*e.g.*, Gurobi, SCA) can be used to solve mixed-integer non-convex programming (MINCP), their stochastic nature cannot guarantee a (near)-optimal and feasible solution for all subsequent TTIs due to dynamic and uncertain channel conditions at the small timescale. Moreover, the exponential computational complexity of these solvers limits their practical feasibility, especially in large-scale scenarios with a high number of variables. Additionally, traffic demand, queue length, and wireless channels are initially incompletely known (or unknown) at each frame. Traffic demand, flow-split decision, and RB assignment rely mainly on previous states updated by the RAN layer.

Given dynamic traffic with fluctuating packet arrivals between frames, it is essential to precisely tailor our proposed method for optimizing long-term variables on a frame-by-frame basis. In this chapter, we consider that the traffic demand vector $\boldsymbol{\lambda}[t]$, the global flow-split vector $\boldsymbol{\varphi}[t]$ and the RB assignment vector $\boldsymbol{\pi}[t]$ are only updated once per frame t, aiming to reduce the computational complexity and information exchange and ensuring a stable queueing system in dynamic scheduling scenarios. On the other hand, the power allocation vector $\boldsymbol{p}[t_i]$ and the achievable instantaneous rate $\boldsymbol{R}[t_i]$ are optimized based on the effective real-time CSI in time slot t_i, adapting to dynamic environments. To achieve a high QoE in each frame, an efficient and adaptable solution to the long-term subproblem of (5.9) is needed. MADRL is a promising technique to solve non-convex optimization problems with reduced computational complexity. The proposed Algorithm 4 summarizes the overall approach to solving problem (5.9), with detailed solutions for each step to follow. Fig. 5.3 shows the end-to-end high-level intelligent TS deployment.

Algorithm 4 Proposed Intelligent TS Algorithm for Solving Problem(5.9)

```
Initialization: Set $t = 1$, $\boldsymbol{\varphi}_u[1] = \frac{1}{M}\mathbf{1}_{\boldsymbol{M}\times\mathbf{1}}, \forall u$, and the initial queues are set to be empty $q_{m,u}[1] = 0$.
for [/long-timescale] $t = 1, 2, \ldots, T$ do
    Traffic demand prediction: Given the sorted data $(\boldsymbol{\lambda}[t-1], \boldsymbol{q}[t-1])$ at SMO, rAPP1 predicts the traffic demand $\hat{\boldsymbol{\lambda}}[t]$ based on an LSTM agent.
    Traffic flow splitting estimation: The heuristic method embedded in rAPP2 splits the traffic flows of all users $\hat{\boldsymbol{\varphi}}[t]$ by (5.10) based on the moving average of the rate in the most recent TTIs.
    RB assignment prediction: Given the sorted data $(\hat{\boldsymbol{\lambda}}[t], \hat{\boldsymbol{\varphi}}[t], \boldsymbol{q}[t-1], \boldsymbol{G}[t-1], \boldsymbol{\pi}[t-1], \boldsymbol{e}^{\mathsf{x}}[t])$ at SMO, rAPP3 with the two DRL agents predicts binary RB assignments $\hat{\boldsymbol{\pi}}[t]$.
    for [/short-timescale]$t_i = 1, 2, \ldots, T_i$ do
        Optimizing power allocation: Given the queue length vector $\boldsymbol{q}[t_i]$, and all predicted long-term variables $(\hat{\boldsymbol{\lambda}}[t], \hat{\boldsymbol{\varphi}}[t]$, and $\hat{\boldsymbol{\pi}}[t])$ solve problem (5.15) to obtain power allocation $\boldsymbol{p}^*[t_i]$.
        Updating queue-lengths: Queue-lengths are updated as:
            $$q_{m,u}[t_i] = \max\Big\{\Big(q_{m,u}[t_i-1] + \varphi_{m,u}[t]\lambda_u^{\mathsf{x}}[t]Z^{\mathsf{x}}\delta_i - R_{m,u}^{\mathsf{x}}(\boldsymbol{p}^*[t_i])\delta_i\Big), 0\Big\}, \mathsf{x} \in \{\mathsf{ur}, \mathsf{em}\}.$$
    end for
    Update $\{\boldsymbol{\lambda}[t], \boldsymbol{\varphi}[t], \boldsymbol{q}[t-1], \boldsymbol{G}[t-1], \boldsymbol{\pi}[t-1], \boldsymbol{e}^{\mathsf{x}}[t]\}$ to the data storage located at SMO via the O1 interface.
end for
```

5.6 The Proposed Algorithms

In this section, we develop effective algorithms to solve subproblems. An optimal TS policy relies on accurate predictions of long-term variables such as traffic demand, flow-split decisions, and RB scheduling. This requires prior knowledge at the non-RT RIC. The data collected from the lower layer (RAN components) is updated on a long-term scale in the data storage at the non-RT RIC. The main aim of this chapter is to leverage observable historical system knowledge via the O1 interface to build a smoother and optimal response.

5.6.1 LSTM Model for Predicting $\boldsymbol{\lambda}[t]$ at rAPP1

The main challenge in the TS scheme is accurately predicting the arrival rate of all services. Since $\boldsymbol{\lambda}[t]$ and $\boldsymbol{q}[t]$ are incompletely known at each frame, standard optimization techniques for long-term variables are inapplicable. Furthermore, the queue length of the generic data flow u in the next frame depends on $\lambda_u^{\mathsf{x}}[t]$ in the previous and current frames. To address this, the RNN-LSTM model is adopted to learn and predict the traffic patterns of all users in the considered Open RAN architecture [123]. LSTM, as a flexible model, can be trained for any type of traffic model and different network scenarios via fine-tuning to capture the dynamics of the system.

Once offline training of the RNN-LSTM model is done in non-RT RIC, the inference is forwarded to other rAPPs to learn flow-split decisions and RB assignments between traffic flows. The long-term variable prediction in rAPPs and short-term power allocation in xAPP are continuously implemented until desired KPI values or required QoS of traffic are met. To achieve optimal TS with an unknown data arrival rate, the LSTM agent continuously monitors $\boldsymbol{\lambda}[t]$ throughout the network. The RNN's weights are updated based on actual parameters to reflect changes and enhance performance until the goal KPI criteria are met.

5.6.2 The Heuristic Approach for Optimizing the Flow-split $\boldsymbol{\varphi}[t]$ at rAPP2

Given the LSTM model's inference result, the predicted traffic demands for the next frame $\hat{\boldsymbol{\lambda}}[t]$ are sent to rAPP2 and rAPP3 at non-RT RIC to optimize and estimate the flow-split decision $\boldsymbol{\varphi}[t]$ and RB assignment $\boldsymbol{\pi}[t]$. At the beginning of each frame, we lack information on the number of arriving traffic packets. Devising optimal flow-split and RB assignment policies under service request dynamics is challenging due to unknown future time slot CSI. As a simple yet efficient solution, we propose a heuristic-based approach to plan the traffic flow splitting factor $\boldsymbol{\varphi}[t]$. Considering that the data rate of users in the next frame is unknown, we use the average data rate in the most recent frames. Let us define $\bar{R}_{m,u}[t] = \frac{1}{W}\sum_{l=t-W+1}^{t} R_{m,u}[l]$, where $R_{m,u}[l]$ is the achievable rate of user u served by RU m at the frame l, and W is the window size. The traffic flow-split for user u to RU m is computed as follows

$$\hat{\varphi}_{m,u}[t] = \frac{\bar{R}_{m,u}[t]}{\sum_{m\in\mathcal{M}_n}\bar{R}_{m,u}[t]},\ \forall m,u. \tag{5.10}$$

The choice of window size involves a balance between precision and responsiveness. A larger window size offers a more stable estimate but might exhibit slower reactions to changes. Conversely, a smaller window size can respond swiftly but might exhibit more noise due to short-term variations in packet arrivals. Considering the dynamic nature of network traffic, it is crucial to periodically assess and, if necessary, adapt the window size according to evolving network conditions and application demands. The estimated flow-split decision $\hat{\boldsymbol{\varphi}}[t]$ is promptly transferred to the embedded rAPP3 to predict the RB assignment $\boldsymbol{\pi}[t]$.

5.6.3 Multi-agent Double Deep Q-Network for Optimizing RB Scheduling $\boldsymbol{\pi}[t]$ at rAPP3

The most recent works on dynamic scheduling for eMBB and uRLLC have often relied on assigning RBs per TTI, resulting in high computational complexity. To overcome this, we adopt a different approach by allocating RBs to all users based on their requirements at the beginning of each frame. However, the unknown channel behavior and queue length make it impossible to obtain an optimal RB assignment. To address this issue, we propose a MADRL-based approach to predict the RB assignment matrix $\boldsymbol{\pi}[t]$ per frame, providing an efficient solution for RB allocation in dynamic scheduling scenarios.

MADRL Framework: DRL combines reinforcement learning with deep learning to train agents through interactions with the environment. The problem at hand is commonly modeled as a Markov decision process (MDP), where the agent interacts with the environment over time steps (or time frames) denoted as t. In this MDP framework, the agent resides in a state $\mathbf{s}[t] \in \mathcal{S}$ and selects an action $\mathbf{a}[t] \in \mathcal{A}$ at each time-step based on a policy denoted as $\Pi(\mathbf{a}[t]|\mathbf{s}[t])$. Here, $\mathcal{S}$ and $\mathcal{A}$ represent the state space and the action space, respectively. This formulation allows for a systematic and decision-based approach to address the problem. After observing the reward $\mathbf{r}[t]$, it transitions to the next state $\mathbf{s}[t+1]$. The probability of selecting action $\mathbf{a}$ given the state $\mathbf{s}$ is expressed by the policy $\Pi(\mathbf{a}|\mathbf{s}) := \mathbb{P}(\mathbf{s}[t+1] = s'|\mathbf{s}[t] = \mathbf{s}, \mathbf{a}[t] = \boldsymbol{a})$.

Traditional Q-learning algorithms suffer from slow convergence, especially for problems with large state/action spaces. Deep Q-networks (DQNs) are used to approximate the Q-function, but they can sometimes overestimate action values, leading to instability with high oscillation and variance due to correlations among observations, affecting policy quality. To address slow convergence and multi-binary action scenarios, we adopt a novel double deep Q-networks (D2QNs) with a customized activation function introduced in this chapter.

Multi-agent D2QN (MAD2QN)-based Approach: D2QN improves DQN by defining *evaluation* and *target* neural networks, decoupling action selection from evaluation. The evaluation network Q handles action selection and policy evaluation, while the target network μ calculates future Q-values. To enhance DQN algorithm stability, we utilize an iterative update technique. It updates the target network every C steps and uses mean square loss $\mathcal{L}(\boldsymbol{\theta})$ to minimize correlations between Q-values and target values. To further improve the policy and stabilize the learning model, we employ *replay memory*, as shown in Fig. 5.3. Hence, the transition $(\mathbf{s}, \mathbf{a}, \mathbf{r}, \mathbf{s}')$ is stored in the replay memory based on the first-come-first-serve buffer with limited capacity to be used in the training phase.

We denote the Q-function in each time-step t as

$$Q(\mathbf{s}, \mathbf{a}; \boldsymbol{\theta}) \leftarrow (1-\eta)Q(\mathbf{s}, \mathbf{a}; \boldsymbol{\theta}) + \eta\Big(\mathbf{r} + \gamma \max_{\mathbf{a}} Q(\mathbf{s}', \mathbf{a}; \boldsymbol{\theta})\Big), \tag{5.11}$$

where $\eta \in [0,1]$, $\gamma \in [0,1]$ and $\boldsymbol{\theta}$ are the learning rate, the discount factor, and the trainable parameters (weights and biases) of the neural network, respectively. It is necessary to optimize $\boldsymbol{\theta}$. To do this, we minimize the distance between $Q(\mathbf{s}, \mathbf{a}; \boldsymbol{\theta}^Q)$ and TD-target (or temporal distance of Q as $y = \mathbf{r} + \gamma Q(\mathbf{s}', \operatorname{argmax}_{\mathbf{a}} Q(\mathbf{s}', \mathbf{a}; \boldsymbol{\theta}^Q); \boldsymbol{\theta}^\mu))$, which is expressed as the loss function $\mathcal{L}(\boldsymbol{\theta}^Q)$

$$\mathcal{L}(\boldsymbol{\theta}^Q) = \mathbb{E}\big(y - Q(\mathbf{s}, \mathbf{a}; \boldsymbol{\theta}^Q)\big)^2, \tag{5.12}$$

where $\boldsymbol{\theta}^\mu$ and $\boldsymbol{\theta}^Q$ are the trainable parameters of the target network and Q-network, respectively. While the target neural network μ evaluates the quality of the action, the Q-network Q determines the action. This procedure is in contrast to the vanilla implementation of DQN, where the target neural network is responsible for both action selection and evaluation.

Unlike single-agent D2QN related to the learning process of only one single agent, our proposed MAD2QN-based model involves more than one agent, where all agents operate autonomously and concurrently in a sharing environment. Given the large-scale and inherent intricacies of the proposed wireless network, the development of an efficient MAD2QN model with specific characteristics is imperative. Moreover, the dynamic nature of packet arrivals and the constraints of slice awareness within our proposed problem demand that

Algorithm 5 MAD2QN-based RB Scheduling deployed at rAPP3

```
Initialization: Initialize random weights $\boldsymbol{\theta}^{\mu} = \boldsymbol{\theta}^{Q}$; set flags $\mathtt{F}^i = 0$, replay buffer capacity $C^{\mathsf{max}}$ and reward values to $\mathbf{r}^i[t] = 0$.
for epoch do
    Receive initial states for all agents $\mathbf{s}[1]$;
    for $t = 1, 2, \dots, T$ do
        for agent do
            Generate a random number rand();
            if rand() $< \epsilon$ then
                Random generating action $\mathbf{a}^i[t]$;
            else
                Select action $\mathbf{a}^i[t]$ for $\mathbf{s}^i[t]$ predicted by $i$-th Q-network;
            end if
            Check action feasibility by passing through the post-process filter;
            if $\mathbf{a}^i[t]$ does not satisfy constraints (5.9b) and (5.9g) then
                Set reward value as $\mathbf{r}^i[t] \mathrel{+}= \mathtt{penalty}$;
            else if $\mathbf{a}^i[t]$ satisfies constraints (5.9b) and (5.9g) then
                Set reward value as $\mathbf{r}^i[t] \mathrel{+}= \mathtt{bonus}$, and set flag $i$ to 1, i.e. $\mathtt{F}^i = 1$;
            else if $\prod_i \mathtt{F}^i = 1$ then
                Set reward values via joint action $\boldsymbol{a}[t] = \{\mathbf{a}^i[t]; \forall i\}$ and updates from RAN as $\mathbf{r}^i[t] + = \mathtt{global\ reward}$;
            end if
            Observe new state $\mathbf{s}^i[t+1]|_{\forall i}$;
            Store transition $(\mathbf{s}^i[t], \mathbf{a}^i[t], \mathbf{r}^i[t], \mathbf{s}^i[t+1])|_{\forall i}$ into $i$-th replay buffer;
            Sample the random mini-batches of $K$ transitions from $i$-th reply buffer;
            Update $i$-th Q-network $Q$ by minimizing the loss function: $\mathcal{L}(\boldsymbol{\theta}^{Q}) = \mathbb{E}(y - Q(\mathbf{s}^i, \mathbf{a}^i; \boldsymbol{\theta}^{Q}))^2$;
            Update the parameters of target neural network $\mu$ of agent $i$ every $C$ steps by resetting $\boldsymbol{\theta}^{\mu} = \boldsymbol{\theta}^{Q}$;
        end for
    end for
end for
```

each agent adapts to these fluctuations in the environment. Another challenge in our approach using MAD2QN is the incorporation of binary actions, which require customized neural networks to address this specific requirement. This adds to the complexity of our model while catering to the intricacies of the wireless network.

The Proposed Design: To handle the complex binary multi-action scenario, we assume one agent per slice with index i to exploit the intrinsic properties formulated problem. Specifically in this learning method, two agents $i = \{1, 2\}$ are defined: an agent with index $i = 1$ for the eMBB slice and an agent with index $i = 2$ for the uRLLC slice. This approach simplifies the problem and leads to faster and more stable convergence. Each dedicated agent takes actions and receives rewards based on its specific state, different from the other agent in a given time frame. Personalized decision-making is achieved with one agent per slice, tailored to each slice's unique requirements. Note that when each agent is motivated solely by its individual reward, it can exhibit self-centered behavior, potentially causing a decline in overall network performance [129]. To mitigate this, we introduce a flag system (one flag for each agent) that facilitates information exchange among agents to obtain the global optimum. This enables communication and insight into the actions and performance of their counterparts per time frame, leading to more informed decisions and a comprehensive understanding of system dynamics. Here, $\mathtt{F}^i \in \{0, 1\}$ shows the flags of slice-1 and slice-2, respectively.

State and Action Spaces: In particular, each agent i operates within its own state $\mathbf{s}^i[t] \in \mathcal{S}^i$ and action $\mathbf{a}^i[t] \in \mathcal{A}^i$ space. The state space captures the subset of environment observations (associated with the assigned slice) that each agent has access to. While the action space represents the independent set of actions that each agent can choose from. The joint action $\mathbf{a}[t] = \{\mathbf{a}^i[t]; \forall i\}$ is a combination of individual actions, impacting the overall system dynamics. Distinct state and action spaces allow agents to personalize perception and interaction with the environment while collaborating toward their objectives. The state vector $\mathbf{s}^i[t]$ at frame t is composed of the traffic demand vector at the current frame $\boldsymbol{\lambda}[t]$, the estimated flow-split distribution $\boldsymbol{\varphi}[t]$ in t-th frame, the previous queue length vector $\boldsymbol{q}[t-1]$, the channel gain matrix of each slice $\boldsymbol{G}^i[t-1]$, the action selected at previous frame $t-1$ as $\boldsymbol{a}^i[t-1]$ and $\boldsymbol{e}^{\mathsf{x}}[t] = [e_u^{\mathsf{x}}[t]]^T$, *i.e.*,

$$\begin{aligned}
\mathcal{S}^i|_{i=1} &:= \Big\{\mathbf{s}^1[t] \Big| \mathbf{s}^1[t] = (\boldsymbol{\lambda}[t], \boldsymbol{\varphi}[t], \boldsymbol{q}[t-1], \boldsymbol{G}^1[t-1], \boldsymbol{a}^1[t-1], \boldsymbol{e}^{\mathsf{ur}}[t])\Big\}, \\
\mathcal{S}^i|_{i=2} &:= \Big\{\mathbf{s}^2[t] \Big| \mathbf{s}^2[t] = (\boldsymbol{\lambda}[t], \boldsymbol{\varphi}[t], \boldsymbol{q}[t-1], \boldsymbol{G}^2[t-1], \boldsymbol{a}^2[t-1], \boldsymbol{e}^{\mathsf{em}}[t])\Big\}.
\end{aligned} \tag{5.13}$$

We define $\mathcal{A}^i$ as the multi-action space

$$\mathcal{A}^i := \Big\{\mathbf{a}^i[t] \Big| \mathbf{a}^i[t] = \big[\pi_{m,u}^{f_i,t_i}[t]\big]^T; \pi_{m,u}^{f_i,t_i}[t] \in \{0, 1\}\Big\}. \tag{5.14}$$

Note that the transitions are updated to $(\mathbf{s}^i, \mathbf{a}^i, \mathbf{r}^i, \mathbf{s}'^i)$ stored in replay memory to be used to update the Q-networks parameters.

Customized Activation Functions: A customized activation function is designed in the last layer of the Q-network and target network. This ensures that the Q-values align appropriately with our specific action spaces. Fine-tuning the activation function tailors Q-values to match multi-action requirements and constraints, enhancing compatibility with the action selection process and improving overall performance. As illustrated in Fig. 5.4, the output of the last layer of each neural network per slice with a size of $M \times U \times F_i \times T_i$ is divided into $F_i \times T_i$ parts, including $M \times U$ cells. Hence every $M \times U$ cells belong to RB(f_i, t_i). We adopt a *Softmax* activation function on given Q-values to convert them into the range [0, 1] so that the sum of all $M \times U$ Q-values equals 1. Then, we apply a post-process filter to convert Q-values into binary values, making problem (5.9) feasible. The cell with the highest Q-value is assigned "one", indicating RB(f_i, t_i) allocation to user u in the m-th RU. All other cells are assigned "zero", indicating that there is no RB allocation to those users. This filter converts continuous Q-values into binary RB allocations, facilitating straightforward RB scheduling decisions based on the highest Q-values.

Construction of the Reward Function: For an effective reward function, we propose a penalty-based approach that incorporates action constraints (*i.e.* (5.9b), (5.9g)). Violating constraints leads to a negative reward value (`penalty`) to discourage such behavior, while satisfying all constraints results in a positive reward value (`bonus`) and a flag of "one".

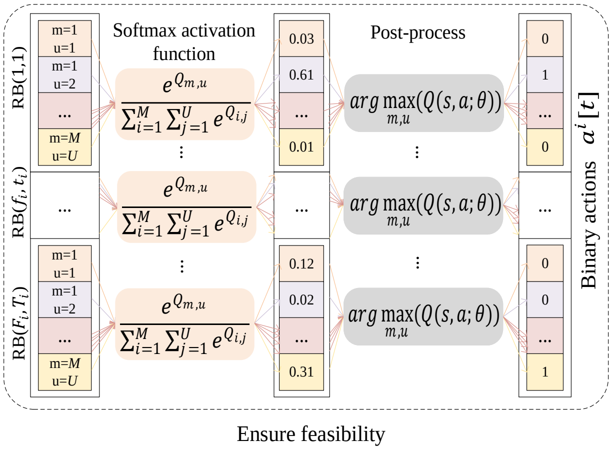

RB(1,1)
RB(f_i, t_i)
RB(F_i, T_i)
m=1 u=1
m=1 u=2
...
m=M u=U
Softmax activation function
$\frac{e^{Q_{m,u}}}{\sum_{i=1}^{M}\sum_{j=1}^{U} e^{Q_{i,j}}}$
0.03
0.61
0.01
0.12
0.02
0.31
Post-process
$arg\max_{m,u}(Q(s,a;\theta))$
0
1
0
0
0
1
Binary actions $\boldsymbol{a}^i[t]$
Ensure feasibility

FIGURE 5.4: The customized activation function design for ensuring the feasibility of last layer D2QN outputs per agent.

This approach encourages agents to prioritize decisions that comply with constraints and accomplish the overarching objectives. Once the flags for all agents are set to 1, the agents are rewarded with a `global reward` based on joint actions $\boldsymbol{a}[t]$ aligned with the objective function of the proposed model as $\omega\big(\frac{\sum_{u\in\mathcal{U}^{\mathsf{em}}}\mathcal{R}_u^{\mathsf{em}}(\boldsymbol{p}[t_i])}{R_0}\big) - (1-\omega)\big(\frac{\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\tau_u^{\mathsf{ur}}\}}{\tau_0}\big)$.

5.6.4 Solving the Short-term Subproblem at DU

After implementing the inferences of the trained models in the intelligent TS-xAPP in the near-RT RIC via the A1 interface, the given xAPP is in charge of controlling and managing long-term variables $(\hat{\boldsymbol{\lambda}}[t], \hat{\boldsymbol{\varphi}}[t], \hat{\boldsymbol{\pi}}[t])$ in dynamic environments. The subsequent step is optimizing the power control problem at the RAN layer located at DUs thanks to the closed-control loop between DUs, CU, and near-RT RIC as follows

$$\min_{\boldsymbol{p}} \sum_{u\in\mathcal{U}^{\mathsf{em}}} q_u[t_i] \tag{5.15a}$$

$$\mathsf{s.t.} \quad (5.9c), (5.9e), (5.9f), \text{and}(5.9h). \tag{5.15b}$$

Problem (5.15) is inherently convex in $\boldsymbol{p}[t_i]$. The worst-case complexity of the interior-point method [101, Chapter 6] used to solve (5.15) is $\mathcal{O}\big(\sqrt{c}(v)^3\big)$, where $c = M_nU(F_1 + F_2) + M_nU + 2M_n$ and $v = M_nU(F_1 + F_2)$ are the numbers of constraints and scalar variables, respectively. It is noted that all the constraints in (5.15) are linear, which can be effectively solved using the standard convex solvers (*i.e.*, MOSEK, SeDuMi). Furthermore, the proximity of DUs to RUs ensures minimal latency when transmitting decisions to RUs.

Table 5.1: Simulation Parameters

Parameter	Value	Parameter	Value
No. RUs	4	Pre. uRLLC latency (D^{ur})	0.5 ms
No. eMBB users	9	No. of antennas (K_{tx})	8
No. uRLLC users	3	Pre. eMBB data rate (R^{th})	10 Mbps
BW of RU	10 MHz	uRLLC SNR threshold (Γ^{th})	10.6 dB
Error prob. (P_e)	1e-03	Pre. RU's queue-length (Q^{max})	10 KB
Power's RU (P^{max})	43 dBm	Discount factor (γ)	0.9
Noise power (N_0)	-110 dBm	Learning rate (η)	0.0001
Packet size (Z^{ur})	32 B	Buffer capacity (C^{max})	1e+05
Packet size (Z^{em})	64 KB	Batch size	64
Time-frame (Δ)	10 ms	`penalty, bonus`	-10, 5

5.7 Performance Evaluation

This section begins by presenting the simulation setup, parameters, and a set of benchmark schemes in Section 5.7.1. Section 5.7.2 provides numerical results and performance comparisons of the proposed Algorithm 4 against the mentioned schemes, showcasing their respective strengths and weaknesses in addressing the discussed challenges.

5.7.1 Simulation Setup, Parameters and Benchmark Schemes

We consider a network topology comprising four RUs, nine eMBB users, and three uRLLC users where each RU serves three sectors, as illustrated in Fig. 5.5. All users are uniformly distributed within a circular region of a radius of 500 m. The channels between RUs and UEs undergo both follow Rayleigh fading with a path-loss model given by $\mathsf{PL}_{m,u} = 128.1 + 37.6 \log_{10}(d/1000)$ dB. We employ an RNN model with the activation function, *adam* optimizer, and 50 epochs to predict future frame traffic. The model has two hidden layers (fully connected), each including 50 LSTM units. The traffic arrival process follows the Poisson process model for uRLLC and eMBB with mean arrival rates of 1.12 and 21.12 [packets/frame] [93], respectively, while the mean arrival rates are configurable parameters. We assume the inter-arrival time is modeled uniformly (*i.e.*, per frame). We then store them in a buffer using a first-come, first-serve scheduling policy. The dataset consists of 10000 traffic observations collected from four RUs over 100 seconds. In the following experiments, Algorithm 4 is executed for 1000 sub-frames (equivalent to 100 frames in the 5G NR context). The D2QN model architecture has five hidden layers with neuron counts of 256, 512, 512, 512, and 256, respectively. The activation function used in the input and hidden layers is *relu*, while a customized activation function is used in the last layer. The training process employs the *binary-crossentropy* loss function and the *adam* optimizer with a learning rate of $\frac{1}{(\text{epoch}+1)^{0.5}}$. The LSTM RNN and MAD2QN models are implemented using TensorFlow version 2.12.0 with the Keras API. The simulations are run on a Dell desktop computer with an Intel(R) core(TM) i7-10610U CPU @ 1.80 GHz and 16 GB of RAM. Table 5.1 summarizes the main simulation parameters used in the experiments.

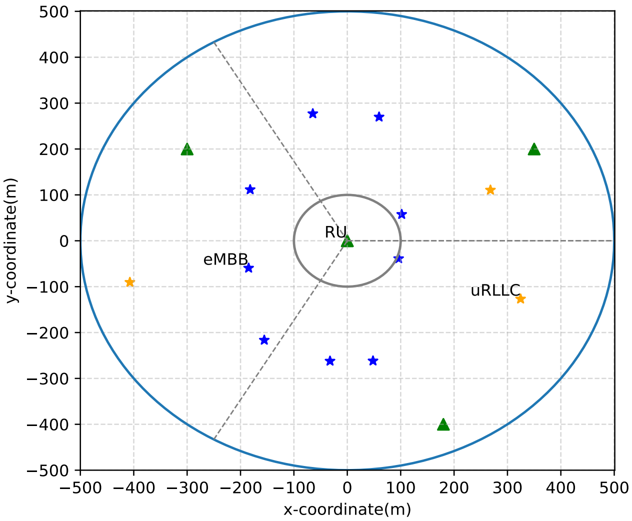


FIGURE 5.5: The considered topology with $M = 4$ RUs, $U^{\mathsf{ur}} = 3$ uRLLC users and $U^{\mathsf{em}} = 9$ eMBB users.

Benchmark schemes: To assess the efficacy of the proposed algorithm, we consider the following five benchmark schemes:

1. *Successive convex approximation (SCA)*: Binary variables $\boldsymbol{\pi}$ are first relaxed to continuous ones, and then an SCA-based iterative algorithm is developed to solve the approximate convex program [123]. This scheme also considers mixed numerologies in the frequency domain and perfect CSI per TTI, which serves as the upper bound of the proposed method.

2. *Uniform $\boldsymbol{\varphi}$*: This scheme aims to highlight the importance of optimizing the flow-split distribution. We assume an equal flow-split for all traffic to RUs, *i.e.*, $\varphi_{m,u} = \frac{1}{M}$ for all $u \in \mathcal{U}$, and consider multiplexing in the frequency domain.

3. *Uniform $\boldsymbol{\pi}$*: This scheme demonstrates the performance improvement achieved by predicting RB scheduling using the multi-agent D2QN. RBs are assigned uniformly to all users in the frequency domain.

4. *Fixed numerology*: In this approach, the TTI is set to match the LTE standard, with a duration of 0.5 ms and an SCS of 180 kHz. The flow-split decisions, resource allocation, and power allocation for both services are determined using Algorithm 4 with slight modifications.

5. *Slice isolation (SI)*: This scheme examines the performance of multiplexing in the frequency and time domains. It emphasizes the importance of incorporating awareness into the NS technique.

The above benchmark schemes are used to comprehensively evaluate the proposed Algorithm 4 to solve the problem 5.9 and demonstrate its superiority over existing approaches.

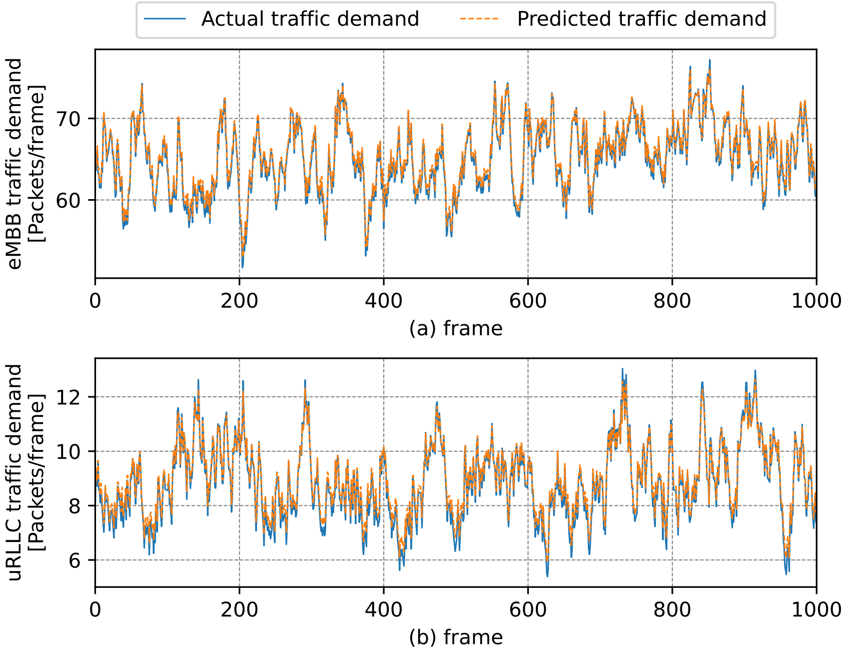


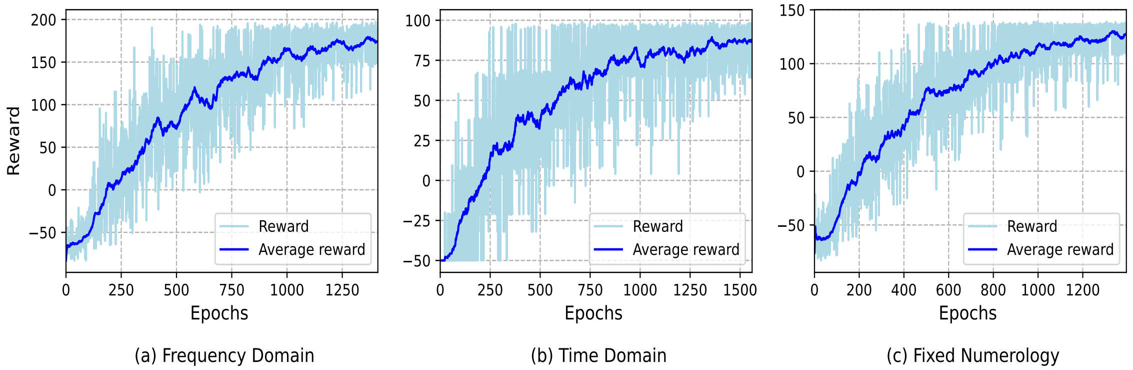
FIGURE 5.6: The actual and predicted traffic demands via LSTM model for both eMBB and uRLLC services per frame.

These benchmark schemes are employed to provide a comprehensive evaluation of the proposed Algorithm 4 to solve the optimization problem 5.9 and its superiority over existing approaches.

5.7.2 Numerical Results and Performance Comparison

The performance of the LSTM RNN model in predicting both eMBB and uRLLC traffic demands is illustrated in Fig. 5.6. Fig. 5.6(a) displays the predicted and actual values for one of the eMBB users, while Fig. 5.6(b) shows the uRLLC traffic demands. The results clearly demonstrate the effectiveness of the trained LSTM RNN model in capturing the dynamic nature of traffic demands over the frames. The predicted values closely align with the actual values, indicating the model's ability to accurately forecast traffic demands. The discrepancy between the predicted and actual values is minimal. To quantify the accuracy of the LSTM model, the mean squared error (MSE) is calculated as a performance metric. For selected eMBB users depicted in Fig. 5.6(a), the MSE value is measured to be 0.00232. Similarly, for the uRLLC users shown in Fig. 5.6(b), the MSE value is calculated as 0.00291. These low MSE values further validate the accuracy of the implemented LSTM model in predicting traffic demands for both eMBB and uRLLC services.

In Fig. 5.7, we present a comprehensive visualization of the performance achieved by Algorithm 5 across different numerologies: Fig. 5.7(a) with the mixed numerologies in the frequency domain with slice awareness (SA), Fig. 5.7(b) with the mixed numerologies in



(a) Frequency Domain

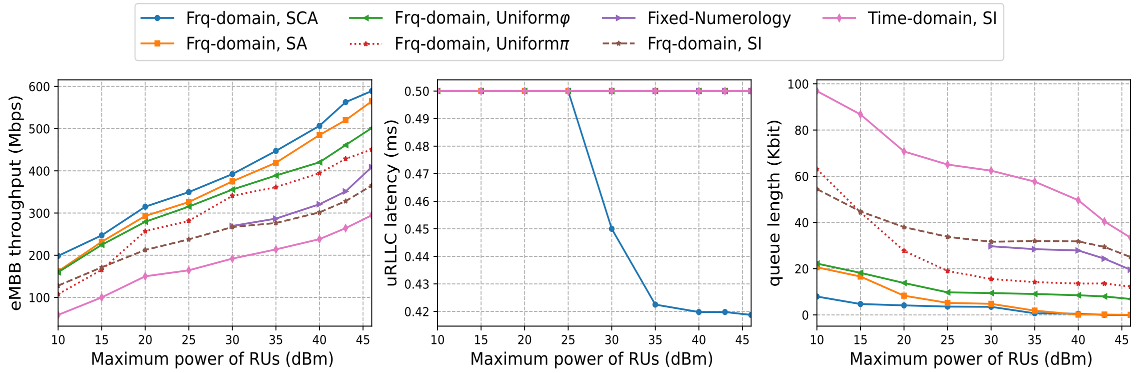
(b) Time Domain

(c) Fixed Numerology

Figure 5.7: The converge behavior of Algorithm 5 in different frequency-time grid numerologies: (a) mixed numerology in the frequency domain, (b) mixed numerology in the time domain, and (c) fixed numerology.

the time domain with SI, and Fig. 5.7(c) with the fixed numerology with SA. In particular, when examining mixed numerologies in the frequency domain combined with SA, we observe a slightly superior performance compared to the fixed numerology combined with SA and mixed numerologies in the time domain combined with SI. While it is worth noting that some RBs are wasted in mixed numerologies in the frequency domain due to guard bands between adjacent numerologies, which can reduce eMBB data rates, the SA scheme stands out as it offers high performance in terms of eMBB data rates, ultimately leading to the improved overall system performance. Furthermore, even though the concept of mixed numerologies in the time domain with SI appears to promise superior performance compared to the fixed numerology with SA, the opposite holds true. Surprisingly, the SA technique proves to be remarkably effective, even in a fixed numerology scheme, which is the underlying reason for this phenomenon. On the other hand, these figures demonstrate the adaptability of agents in response to dynamic changes in the channel and arrival packets over time frames. The figures show that agents quickly learn and improve their performance, as indicated by the increasing average reward during the training episodes for all numerology schemes. Moreover, a higher number of epochs leads to higher average rewards. Among these three schemes, the proposed scheme with mixed numerology in the frequency domain with SA achieves the highest reward under the same conditions. These findings highlight the effectiveness of the Algorithm 5 and the benefits of considering different numerologies and SA in optimizing the system performance.

In Fig. 5.8, we assess the performance of Algorithm 4 using various strategies compared to the above-mentioned benchmark schemes. To assess the eMBB data rate under various resource allocation schemes, Fig. 5.8(a) presents the total data rate of eMBB users across the maximum power budget for RUs, ranging from 10 to 46 dBm. As expected, the SCA strategy demonstrates the highest performance among all the schemes and serves as an upper bound for comparison. This can be attributed to the fact that the SCA scheme utilizes perfect CSI in each frame, allowing it to have precise knowledge of the current frame's channel gain. In contrast, the proposed method relies on the channel gain of the previous frame to allocate RB to the current one. However, the performance gap is less than



(a) Average overall eMBB throughput

(b) Worst-user uRLLC latency

(c) Average queue length

FIGURE 5.8: The performance comparison between Algorithm 4 and existing benchmark schemes versus maximum power budget of RU $P^{\max}$ in terms of (a) average overall eMBB data rate, (b) worst end-to-end uRLLC users, and (c) backlog queue length.

4%, which showcases the efficiency of the LSTM RNN and MAD2QN models in accurately predicting dynamic arrival packets and RB scheduling over time. Moreover, the proposed scheme achieves the highest eMBB data rate when compared to the others. In particular, the proposed method offers a performance improvement of 99.42%, 43.39%, 40.74%, 11.76% and 8.57% compared to the time and frequency domain considering SI, fixed numerology, uniform $\boldsymbol{\pi}$, and uniform $\boldsymbol{\varphi}$, respectively, at the typical power value of $P^{\max} = 30$ dBm. It is worth noting that the benchmark scheme of uniform $\boldsymbol{\varphi}$ performs closely to our proposed method, particularly in lower power budgets. This suggests that allocating equal flow-split to all users across RUs can effectively meet the QoS requirements when the power budget is limited. However, as the power budget increases, the advantages of the proposed method in optimizing resource allocation and maximizing eMBB data rate become more prominent. Besides, the fixed numerology scheme works well over $P^{\max} \geq 30$ dBm, but becomes infeasible when the maximum RUs' power is less than 30 dBm. This observation highlights the advantage of our proposed scheme over the mentioned schemes, particularly at lower power levels. Lastly, we can observe that the SI schemes exhibit the worst performance compared to the SA-based schemes, confirming the importance of incorporating awareness techniques into resource allocation.

In Fig. 5.8(b), the worst-user uRLLC latency is analyzed where all schemes successfully meet the required uRLLC latency threshold of 0.5 ms. The fixed numerology scheme exhibits an empty region for $P^{\max} < 30$ dBm, since the corresponding problem becomes infeasible under these power constraints. Figure 5.8(c) presents the average queue backlog with different maximum power budgets of RUs. As seen, the average queue length decreases as $P^{\max}$ increases. Interestingly, the performance gap between the proposed method and the SCA scheme is negligible. Both schemes can effectively manage the queue length and demonstrate the ability to handle varying power budgets while maintaining a low average backlog. On the other hand, the SI-based schemes yield the poorest performance. Notably, the proposed method excels and exhibits the most superior performance among all

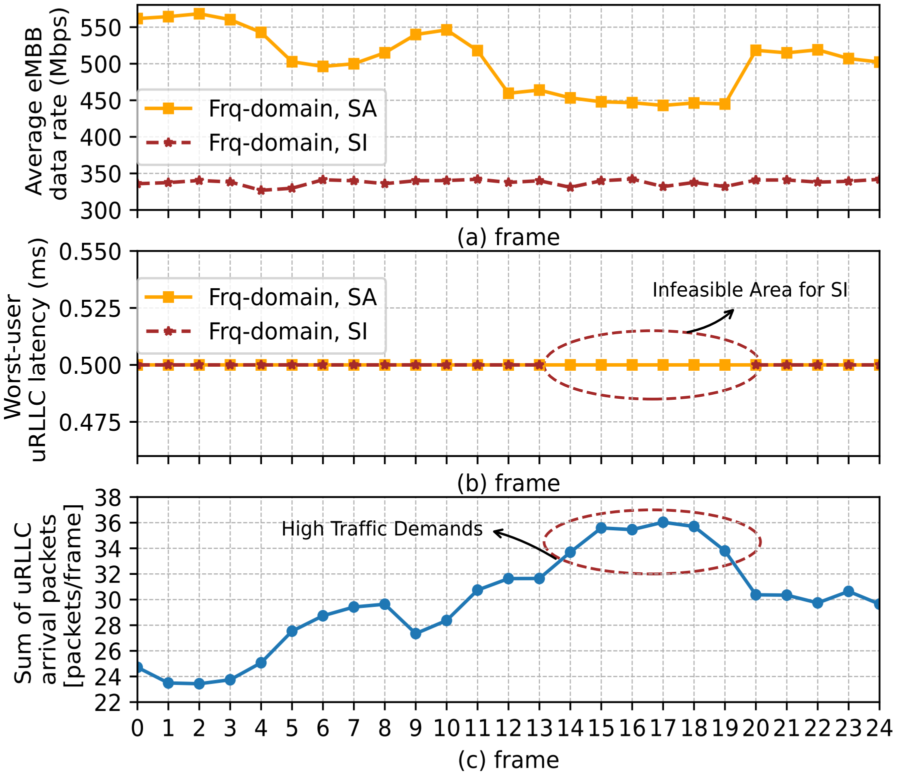


FIGURE 5.9: Impact of the rate of uRLLC arrival packets/frame on the overall performance of eMBB and uRLLC in both SA and SI scenarios with $P^{\mathsf{max}} = 43$ dBm.

benchmark schemes. Similarly, the fixed numerology scheme is infeasible when $P^{\mathsf{max}} < 30$ dBm, and the uniform $\boldsymbol{\varphi}$ benchmark scheme outperforms the uniform $\boldsymbol{\pi}$, fixed numerology and SI-considered schemes. Conversely, we have noticed numerically that uRLLC users consistently tend to maintain only one link in diverse system configurations.

Fig. 5.9 highlights the superiority of the SA scenario, especially in high traffic conditions, compared to the SI approach. Upon closer examination of Fig. 5.9(c), it becomes evident that during frames 14 to 19, the packet demands of uRLLC users surpass the available RBs allocated to their dedicated slice. To address this issue, SA can request additional RBs from the eMBB slice through preemption. On the contrary, the SI technique was shown to be infeasible during these frames. As we expected in Fig. 5.9(b), the SI scenario is incapable of accommodating the uRLLC service during frames in which uRLLC users require more RBs than what is available in their dedicated uRLLC slice. Meanwhile, Fig. 5.9(a) plots the average eMBB data rate of Algorithm 4 with SA and SI in different frames. The figure proves the effectiveness of the SA scenario in achieving a higher data rate compared to SI. Importantly, it shows that the performance of uRLLC does not negatively impact the other slice, namely eMBB, in the SI scenarios. This comparison validates that SA not only increases the eMBB data rate but also enables a response to uRLLC users with unexpectedly high packet demands. Overall, the insights provided by Fig. 5.9 confirm that

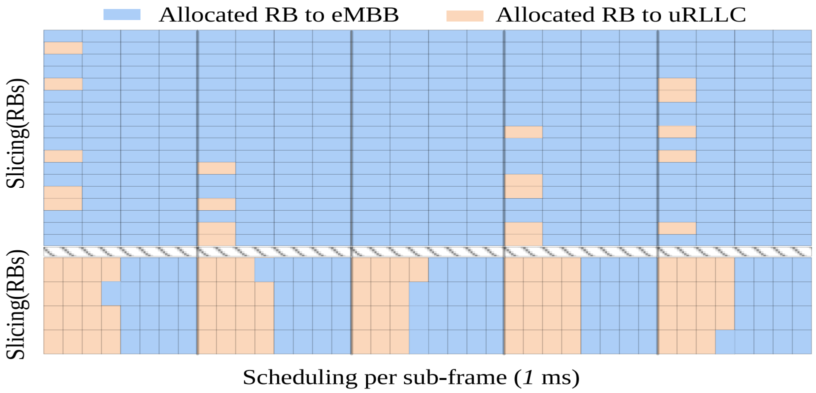


FIGURE 5.10: Demonstration of RB scheduling to both eMBB and uRLLC services based on Algorithm 4 considering mixed numerology in the frequency domain. The vertical solid lines separate the sub-frame from the next sub-frame.

SA is a beneficial technique in improving the eMBB data rate and effectively managing the demands of uRLLC users that exceed the allocated RBs in their dedicated slice.

In Fig. 5.10, we provide a more detailed visualization of how the SA technique works in uRLLC preemption and its impact on improving eMBB data rate. The figure focuses on five sub-frames as an example, illustrating the RBs' allocation to different services. We assume that uRLLC users have failed if all packets per frame are not transmitted, and then it becomes necessary to request extra RBs from another slice. Conversely, eMBB users are also allowed to access the unused RBs of the uRLLC slice, resulting in a significant improvement. From Fig. 5.10, we can observe that in sub-frame 3, the uRLLC service does not require more RBs than what is available in its dedicated slice. However, in the other sub-frames, the uRLLC traffic requests additional RBs from the eMBB slice to meet its demand. Note that to meet the minimum uRLLC latency requirement, the agents will prioritize allocating RBs to uRLLC users at the beginning of each sub-frame. This example clearly shows how the SA technique facilitates the RB allocation, enabling uRLLC users to acquire extra resources when necessary and improving the eMBB data rate. By dynamically adapting the RB allocation based on the various demands, the proposed algorithm optimizes resource utilization and effectively meets the stringent demands of uRLLC and eMBB.

5.8 Summary

We proposed a novel DRL-aided intelligent TS scheme within the Open RAN architecture, aiming to efficiently steer multi-traffic flows. Toward self-optimizing autonomous networks, variables on various timescales are predicted/optimized in different Open RAN layers. Three rAPPs with offline-trained agents were designed in non-RT RIC for long-term variable predictions. The short-term variable was optimized at DUs in the function layer thanks to the long-term inferences deployed at the intelligent TS xAPP in near-RT RIC. In particular, we focused on implementing scalable numerology mechanisms for beyond 5G wireless networks by leveraging the MC, NS, and multiplexing of numerologies. The chapter proposed a multi-agent scenario with a data-driven MADRL algorithm at the non-RT RIC to effectively address complex optimization problems with partial observations. Extensive numerical results were presented to highlight the effectiveness of the proposed approach in jointly optimizing numerology scenarios, slicing adaptation, and scheduling strategies. They also reveal valuable insights into data-driven algorithms' development and their potential for enhancing network performance in the Open RAN architecture. The current framework presents a theoretical foundation for integrating ML into the O-RAN architecture, demonstrating the potential benefits of ML for traffic management and performance optimization under the assumption that there is negligible latency for exchanging decisions and policies between RICs and RAN. An intriguing area for future investigation involves analyzing the scalability and adaptability of the suggested framework within comprehensive simulators, such as OpenRAN Gym, which encompass all specifications and protocols of the O-RAN interfaces. To advance our future endeavors, we aim to enhance efficiency by transitioning from considering individual RBs to utilizing group RBs, thus alleviating computational burdens.

Chapter 6

Intelligent User Association and Scheduling in Open RAN with 7.2x Functional Split: A Hierarchical Optimization Framework

In the ever-evolving landscape of *NextG* wireless networks, Open radio access network (RAN) emerges as a transformative paradigm, revolutionizing network architectures and fostering innovation through its open, intelligent, and disaggregated approach. By integrating RAN intelligent controllers (RICs), we can seamlessly implement machine learning (ML) algorithms to cater to diverse vertical applications and deployment environments without the need for intricate planning. However, this architecture suffers from two critical challenges: frequent handovers and load balancing amid varying traffic demands of different services in dynamic environments. To address these issues, this chapter proposes a joint intelligent user association, congestion control, and resource scheduling (IUCR) scheme. Aligning with the 7.2x functional split (FS) option recommended by the O-RAN Alliance, we present a hierarchical optimization framework incorporating heuristic methods, successive convex approximation (SCA), and a distributed deep reinforcement learning (DRL) approach across different Open RAN components, such as RICs and RAN layers. The simulation results convincingly demonstrate the superior performance of the proposed scheme compared to centralized approaches, validating its effectiveness.

6.1 Introduction

The *NextG* wireless network landscape is experiencing a surge in demand for diverse services, such as enhanced mobile broadband (eMBB) and ultra-reliable low-latency communication (uRLLC). However, current radio access network (RAN) architectures struggle to meet these varied service requirements due to their rigid, monolithic nature [102, 130]. This rigidity hampers efficient resource allocation and adaptation to diverse service demands, leading to suboptimal performance and compromised user experiences. In response, the Open RAN architecture has emerged as a promising solution that enables operators to adapt dynamically to changing traffic patterns, latency requirements, and data rate demands across different services [131].

In an Open RAN architecture, machine learning (ML) methods play the fundamental roles in enabling intelligent and autonomous optimization of RAN performance. ML-based optimization within RANs facilitates autonomous deployment by integrating RAN intelligent controllers (RICs). These controllers utilize embedded ML/artificial intelligence (AI) algorithms to analyze vast amounts of data collected from various RAN components and user equipment (UE). This analysis enables informed decision-making and dynamic adjustment of network parameters [107, 132]. Key tasks include predictive maintenance, anomaly detection, dynamic resource allocation, congestion management, and quality of service (QoS) optimization.

In multi-traffic environments with eMBB and uRLLC services, RAN slicing and scalable numerologies are crucial [111,133]. RAN slicing customizes network resources for each service type, ensuring quick and reliable provision of uRLLC with minimal delays. This resource segregation prioritizes uRLLC traffic, reducing queueing delays and ensuring timely data delivery. Scalable numerology techniques, such as the mini-slot concept, further enhance uRLLC service efficiency by enabling immediate transmission of small packets without buffering [134]. By integrating these techniques, network operators can optimize resource usage, minimize delays, and improve overall network performance and user satisfaction [135].

While Open RAN provides an agile and intelligent framework for IMT-2020 infrastructure, it faces challenges like cell congestion, load balancing, and frequent handovers in UE-centric multi-traffic scenarios. Addressing these issues requires an intelligent user association (UA) and congestion control scheme to optimize RAN performance in dynamic environments. To minimize unnecessary handovers, handover parameters such as user data rate thresholds, queue length capacity, and resource utilization rates should be adapted to the environment's dynamics. The 7.2x functional split (FS) option adds complexity by distributing tasks like linking UEs to radio units (RUs), allocating physical resource blocks (RBs), and managing transmission power across different Open RAN components [136]. This necessitates both distributed management and centralized coordination. Fortunately, Open RAN's architecture, with its open interfaces and RICs, facilitates communication among components, making these processes manageable.

6.1.1 Related Works

Numerous research investigations have explored user handover, load balancing, and traffic management issues within the traditional RAN architecture. For instance, in [137], a joint resource allocation and device-to-device routing problem was proposed to maximize the total system data rate in an orthogonal frequency division multiple access (OFDMA) ultra-dense small cell network. The authors in [138] introduced a ping-pong timer to discern types of handovers induced by movements. In [116], a joint user access control and scheduling optimization scheme was proposed to maximize overall network utilization in the cloud RAN architecture. To evenly distribute loads across base stations (BSs), Zhang *et al.* [29] proposed a dynamic multi-connectivity-based joint scheduling framework via traffic steering for eMBB and uRLLC, enhancing eMBB data rate while ensuring uRLLC latency requirements are met. The authors in [92] formulated an optimal joint eMBB and uRLLC scheduler with the dual objectives of maximizing utility for eMBB traffic while satisfying uRLLC demands. Due to the coupling between optimization variables and their combinatorial nature, most of these problems have relied on differences of convex algorithms with high computational complexity or degraded-accuracy heuristic approaches for solutions in a "one-size-fits-all" architecture.

Moreover, the load-aware user access control scheme was introduced in [139] by exploiting link quality assessments and traffic load data across various BSs to effectively distribute loads. The authors in [140] proposed a handover skipping technique for highly mobile users to bypass inefficient handovers. To address non-stationary traffic demand fluctuations over time, the author of [118] formulated an online learning problem for optimal user association policies using the online convex optimization framework. Additionally, [141] investigated a transport-aware mobility load balancing approach integrated into the LTE traffic steering mechanism to handle radio overload and transport congestion. However, all the mentioned works have focused on user access control in traditional RAN architecture without considering heterogeneous services with conflicting requirements. In such deployments, optimizing user access exclusively within the RAN might become impractical due to excessive complexity and signaling overheads.

Recently, ML-enabled handover schemes have been explored to address traditional methods' complexities and inefficiencies. For instance, in [124], an actor-critic reinforcement learning (RL) model jointly optimizes communication mode selection, RB allocation, and power distribution in device-to-device networks. Similarly, the offline RL-based handover control model was studied in [142] to autonomously optimize handover decisions, enhancing user connectivity and data rate. To manage global network information and frequent handovers, Cao *et al.* [143] proposed a deep reinforcement learning (DRL) model to assist access decision' users in dynamic network environments. However, individual user performance and requirements were overlooked therein. Therefore, intelligent user access (UA) and resource scheduling, considering user-centric conditions, and congestion control are necessary to deal with traffic demand fluctuations.

Few works have explored the UA scheme within the Open RAN architecture. For instance, the work in [144] proposed a DRL-based scheme to optimize user access and avoid frequent handovers in massive BS deployments. The user-specific traffic steering intelligent handover framework was developed in [50] to maximize the system data rate by selecting the appropriate new radio (NR) serving cell. The authors in [49] presented a multi-layer optimization framework to steer traffic through multiple BSs based on the estimated future demand. However, these studies have not adhered to the practical 7.2x FS in Open RAN architecture and have not employed cutting-edge technologies available in beyond fifth-generation (5G), such as slice awareness, mini-slots, multiple numerologies, and others, to enhance overall network efficiency.

6.1.2 Motivation and Main Contributions

Most existing efforts lack an intelligent UA and resource scheduling scheme for multi-traffic downlink (DL) sixth-generation (6G) systems that consider O-RAN and 3GPP standards, such as mixed numerologies in dynamic environments and function split. On one hand, the recommended 7.2x FS complicates the joint design of UA and radio resource management (RRM) so that the existing solutions (*e.g.*, in [49,50,144]) cannot be applied, which requires a new design strategy for UA and RRM in Open RAN architecture. On the other hand, the DRL method is essential to enable automated, adaptive, and efficient network management. It allows networks to self-optimize, dynamically adjust to changing conditions, and improve overall performance, making it a key component in the evolution of advanced wireless networks. Within the Open RAN network, the distributed training process with centralized coordination is crucial, particularly those tailored to 7.2x FS. This involves multiple DRL agents sharing local observations and considering actions taken by other agents to ensure performance convergence [145]. However, such observation-sharing mechanisms can result in significant communication overhead. This calls for a novel framework that encourages individual agents to train their local models based on their own observations while being minimally aware of neighboring agents' behavior, reducing the need for extensive information exchange.

Unlike traditional user handover methods, where decisions are made by the base station or core network based on fixed criteria like signal strength and signal-to-noise ratio (SNR), this chapter presents an innovative approach that utilizes a near-real-time (near-RT) RIC and customized xAPPs. This approach considers factors such as data rate, network load, QoS, and user behavior, leveraging data-driven solutions to enhance both network performance and the user experience. By utilizing RICs, which provide a holistic view of RAN components, we optimize the user-centric scenarios across diverse traffic types. Building on these principles, we develop an intelligent UA, congestion control, and resource scheduling, so-called IUCR, robust against dynamic environments. In summary, our key contributions are summarized as follows:

- In contrast to the mentioned literature, which has not incorporated 5G cutting-edge techniques or adhered to the 7.2x FS in Open RAN, we propose a novel IUCR optimization framework that complies with the 7.2x FS and addresses both eMBB and uRLLC services within the Open RAN architecture. The framework employs RAN slicing and mixed numerologies multiplexing in the frequency domain, with a slice-aware approach to optimize resource utilization. A mini-slot-based frame structure, compliant with 3GPP NR specifications, is incorporated to enable shorter transmission duration. The formulated problem aims to minimize eMBB queue lengths and uRLLC latency while considering congestion control, power budgets, and other constraints.

- We propose a hierarchical optimization approach combining heuristics, iterative successive convex approximation (SCA), and distributed DRL-based algorithms. By leveraging closed-control loops between RAN components and RICs, the proposed solution addresses challenges such as incomplete queue lengths, frequent handovers, channel interference, and partial environmental observations using historical RAN data. The framework first updates UA variables via a heuristic algorithm that reduces frequent handovers by assessing cell congestion and focusing on average dynamics. It then employs a distributed DRL-based algorithm to optimize power distribution, guided by an optimized RB assignment centrally determined through the SCA method in central units (CUs).

- When the network alarms indicate potential cell congestion, the heuristic method at the near-RT RIC updates UA variables using averaged historical data and sends them to RAN components. An iterative SCA algorithm located at DUs then schedules RBs based on these variables, UE traffic demands, and network conditions, using channel feedback from cell sites. In the distributed DRL-based power allocation scheme, each RU trains two deep Q-networks (DQNs) with local observations to make decisions. A reward mechanism and flag system minimize signaling overhead and prevent selfish behavior, addressing incomplete information.
- We provide extensive numerical results that demonstrate the superior performance of the proposed solution, leading to a small performance gap of 1.1% compared to the centralized scheme in terms of eMBB data rate.

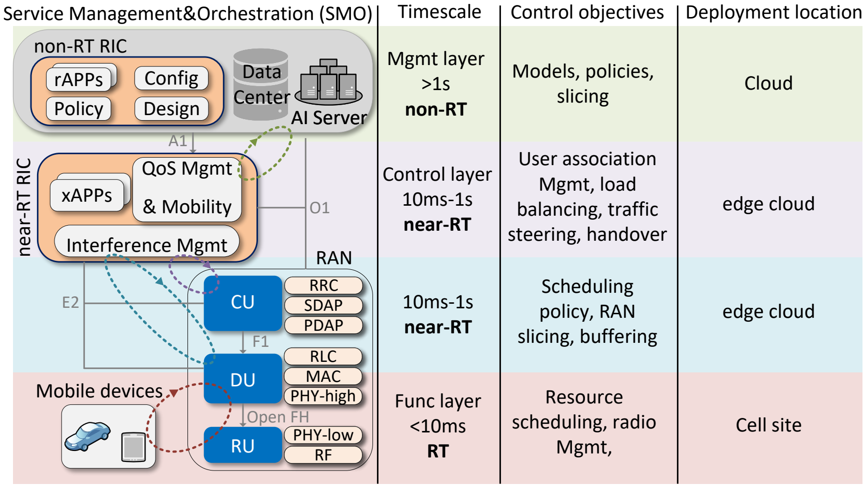
The remainder of this chapter is organized as follows: Section 6.2 presents the complete end-to-end (e2e) workflow in the Open RAN architecture with the 7.2x functional split. The optimization problem of the intelligent user association, congestion control, and resource scheduling problems is formulated in Section 6.3. Section 6.4 describes the proposed methodology, including a heuristic method, iterative SCA, and a multi-agent DRL model. Section 6.5 provides extensive numerical results, comparing the proposed approach to benchmark schemes, while Section 6.6 concludes the chapter.

6.2 System Model

6.2.1 7.2x Functional Split in Open RAN Architecture

Following the O-RAN Alliance principles, the Open RAN architecture in *NextG* can be divided into three main components: CUs, distributed units (DUs), and RUs [132]. The framework includes management, control, and functionality layers, operating on timescales from over one second to under ten milliseconds. As shown in Fig. 6.2.1, O-RAN specifications introduce two key modules: the non-real-time (non-RT) RIC and near-RT RIC at the management and control layers, respectively, connected via the A1 interface. All RAN components are linked to these RICs through the E2 and O1 interfaces. Fig. 6.2.1 illustrates the integration of intelligence across components in a disaggregated cellular network. Each closed-loop control is designed to optimize RAN operations at different timescales, user counts, and data sources [67]. The *non-RT control loop* handles resource orchestration for thousands of devices, coordinating non-RT and near-RT RICs via the A1 interface. The *near-RT control loop* connects near-RT RICs to CUs and DUs, managing hundreds of UEs with xApps for tasks like inference and load balancing. The *RT control loop* primarily operates within the DU but may also extend to RUs or UEs. O-RAN specifications currently do not address ML model deployment within DUs, leaving this for future development.

Disaggregation in Open RAN allows the selection of components from different vendors, making FS a key challenge. To optimize network processes, Open RAN follows 3GPP's



Timescale	Control objectives	Deployment location
Mgmt layer >1s **non-RT**	Models, policies, slicing	Cloud
Control layer 10ms-1s **near-RT**	User association Mgmt, load balancing, traffic steering, handover	edge cloud
10ms-1s **near-RT**	Scheduling policy, RAN slicing, buffering	edge cloud
Func layer <10ms **RT**	Resource scheduling, radio Mgmt,	Cell site

Figure 6.1: Learning-based control loops in Open RAN architecture.

FS guidelines to virtualize BS functions across CUs, DUs, and RUs. Fully centralized RANs are impractical due to the high costs and limited availability of high-capacity fronthaul (FH) networks [146]. This has led to flexible architectures with some BS functions at CUs. However, deciding on the degree of centralization and function allocation among CUs, DUs, and RUs is complex. Each FS option imposes different computational and data flow requirements, sometimes necessitating multiple CUs, which increases costs. The 3GPP Release 14 [147] outlines eight FS options for distributing functions across RAN components. In Open RAN, the protocol stack operates as virtual network functions (VNFs) within virtual machines (VMs) across CUs, DUs, and RUs. Implementing most of the O-RAN stack as VNFs in the CU increases southbound interface capacity demands, while critical applications like uRLLC require low latency and high computation at the edge.

We note that the optimal solution for FS involves balancing centralized approaches, which require a comprehensive network view but also increase overhead and latency. While the distributed approaches offer lower latency but gather data from fewer sources [148]. This study utilizes the O-RAN Alliance's 7.2x FS framework to navigate these trade-offs. In this setup, CU acts as the centralized and virtualized component of the RAN, managing the Radio Resource Control (RRC), Service Data Adaptation Protocol (SDAP), and Packet Data Convergence Protocol (PDCP) layers. The DU handles baseband processing, scheduling, Radio Link Control (RLC), Medium Access Control (MAC), and the upper part of the Physical Layer (PHY). Finally, RU is responsible for the lower PHY layer processing, including Fast Fourier transform (FFT)/inverse FFT, beamforming, and RF-related functions.

Table 6.1: Summary of Notations

Notation	**Definition**
$\mathscr{N}, \mathscr{M}, \mathscr{U}$	Sets of DUs, RUs, and UEs
$\lambda^{\mathsf{x}}, Z^{\mathsf{x}}$	Traffic demand and packet size, $\mathsf{x} \in \{\mathsf{ur}, \mathsf{em}\}$
B_i, Δ	BW of the i-th slice and frame duration
β_i, δ_i	RB's bandwidth and TTI duration of the i-th slice
$T_i,\ F_i$	Number of subcarriers and TTIs per frame in the i-th slice
π, p, φ	RB scheduling, transmission power and UA decision variables
g, h, I	Channel vector, channel gain, and interference
N_0	The noise power
ρ, κ	Number of symbols per RB and number of bits per symbol
$\varepsilon_u^{\mathsf{ur}}$	Number of needed RBs for the u-th uRLLC user per its dedicated slice
e_u^{x}	Number of needed RBs for the u-th user from other slice in slice-awareness scenario
$\tau^{\mathsf{arv}}, \tau^{\mathsf{tx}}, \tau^{\mathsf{prog}}, \tau^{\mathsf{proc}}$	The packet arrival time, transmission, propagation, and processing latencies
$\mu_{cu,du}$	CU and DU task rates
q_0, τ_0	The reference values for eMBB queue length and uRLLC latency
$\omega, \epsilon_k, \upsilon$	The regulatory factor, priority factor in the k-th iteration, and priority factor in the reward function
W	Window size
K_m, L	The total number of RBs available for the m-th agent and transmission power level
$\hat{p}_l$	The l-th transmission power level
$\boldsymbol{s}_m[t], \boldsymbol{a}_m[t], \boldsymbol{r}_m[t], \mathtt{F}_m[t]$	The state, action, reward and flag of the m-th agent per frame t
$r^{\mathsf{indv}}, r^{\mathsf{com}}, r^{\mathsf{neg}}$	The individual, common and negative reward values
$\mathbb{E}\{.\}$	Expected value of the given random variable
$(.)^{+}$	The positive part of the value or function
$\lfloor . \rfloor$	The largest integer less than or equal to the given value

6.2.2 Network Model

Network Characteristics: We consider a DL cellular network composed of one CU, a set of $N = |\mathcal{N}|$ DUs, and a set of $M = |\mathcal{M}|$ multi-antenna (K_{tx}) RUs serving a set of $U = |\mathcal{U}|$ single-antenna UEs. We denote $\mathcal{U}^{\mathsf{ur}}$ and $\mathcal{U}^{\mathsf{em}}$ by the sets of uRLLC and eMBB users, respectively, with $\mathcal{U} = \mathcal{U}^{\mathsf{ur}} \cup \mathcal{U}^{\mathsf{em}}$, $U^{\mathsf{ur}} = |\mathcal{U}^{\mathsf{ur}}|$ and $U^{\mathsf{em}} = |\mathcal{U}^{\mathsf{em}}|$. We consider two slices to serve eMBB and uRLLC services, which request different types of traffic with different packet sizes Z^{x} [Bytes], with $\mathsf{x} \in \{\mathsf{ur}, \mathsf{em}\}$ ($Z^{\mathsf{em}} >> Z^{\mathsf{ur}}$). Let $\lambda_u^{\mathsf{x}}[t]$ [packets/frame] denote the traffic demand of the u-th generic UE in the discrete-time frame indexed by $t \in \{1, 2, \ldots, T\}$, which is bounded by $\lambda^{\mathsf{max}} \leq \infty$. We assume that $\lambda_u^{\mathsf{x}}[t]$ follows the 3GPP model (*ex.*, FTP3 model) with the mean arrival rate $\mathbb{E}\{\lambda_u^{\mathsf{x}}[t]\} = \bar{\lambda}_u^{\mathsf{x}}$ ($\bar{\lambda}^{\mathsf{em}} >> \bar{\lambda}^{\mathsf{ur}}$), in which the size of each packet is identical and equal to Z^{x}. For simplicity, we assume that DUs cover the non-overlapped geographical areas with a disjoint set of RUs, such as $\mathcal{M}_n \triangleq \{(n, 1), \ldots, (n, M_n)\}$ with $\sum_{n \in \mathcal{N}} M_n = M$.

We leverage mixed numerologies in the frequency domain to address conflicting service demands [149]. A RB refers to the smallest time-frequency unit of resources that can be allocated to a UE and is composed of 12 subcarriers and 7 orthogonal frequency division multiplexing (OFDM) symbols. Based on this technique, the available bandwidth (BW) B is divided into two independent parts $B_i; i = \{1, 2\}$ that make two slices to handle dynamic service demands, each adapting a specific numerology denoted by the index i. As in [149], this chapter selects numerology with indices $i = 1$ for eMBB and $i = 2$ for uRLLC applications. The assigned slices for uRLLC and eMBB are defined by ($\beta_2 = 720$ KHz and $\delta_2 = 0.125$ ms) and ($\beta_1 = 360$ KHz and $\delta_1 = 0.25$ ms), respectively. Herein, β_i and δ_i denote the bandwidth of RB and the duration of the i-th numerology at frame with a large-scale coherence time of Δ, respectively. Each RB includes the frequency-time grid, where $F_i = \lfloor B_i/\beta_i \rfloor$ and $T_i = \Delta/\delta_i$ indicate the number of subcarriers indexed as $f_i = \{1, \ldots, F_i\}$ and the number of transmission time intervals (TTIs) per frame indexed as $t_i = \{1, \ldots, T_i\}$, respectively.

Channel Model: We assume that the large-scale fading coefficients remain constant throughout frames, while the small-scale fading components are presumed to remain unchanged during the time frame t but vary independently in the subsequent time frame. Accordingly, we model the quasi-static channel vector between RU m and UE u on all TTIs within one frame t as $\boldsymbol{h}^{f_i}_{m,u} = \sqrt{10^{-\mathsf{PL}_{m,u}/10}}\bar{\boldsymbol{h}}^{f_i}_{m,u}$, where $\mathsf{PL}_{m,u}$ is the path loss and $\bar{\boldsymbol{h}}^{f_i}_{m,u} \in \mathbb{C}^{1\times K_{\mathsf{tx}}}$ represents the complex circularly symmetric Gaussian random variables with zero means and unit variances, *i.e.* $\bar{\boldsymbol{h}}^{f_i}_{m,u} \sim \mathcal{CN}(0, \boldsymbol{I})$. Let us denote by $\boldsymbol{G}[t] \triangleq [g^{f_i}_{m,u}]; \forall m, u, f_i$ the channel gain between RBs and all UEs in frame t, where $g^{f_i,t_i}_{m,u} = g^{f_i}_{m,u} \triangleq \|\boldsymbol{h}^{f_i}_{m,u}\|^2_2; \forall t_i = \{1, \ldots, T_i\}$ is the effective channel gain.

Achievable Downlink Data Rate: We consider the DL data rate [bits/s] of the u-th eMBB UE served by RU m at TTI t_i using the Shannon-Hartley theorem. This approximation is reliable provided that the channel state information (CSI) is available and the actual data rates attained by the 3GPP modulation and coding scheme (MCS) schemes stay within the permitted signal-to-interference-plus-noise ratio (SINR) range that can be expressed as

$$R^{\mathsf{em}}_{m,u}(\boldsymbol{\varphi}, \boldsymbol{p}, \boldsymbol{\pi}) = \varphi_{m,u} \sum_{f_i=1}^{F_i} \beta_i \log_2 \Big(1 + \frac{\pi^{f_i,t_i}_{m,u} p^{f_i,t_i}_{m,u} g^{f_i,t_i}_{m,u}}{N_0 + I^{f_i,t_i}_{m,u}}\Big) \tag{6.1}$$

where $\varphi_{m,u}$, N_0, $\pi^{f_i,t_i}_{m,u} \in \{0, 1\}$, $p^{f_i,t_i}_{m,u}$ and $I^{f_i,t_i}_{m,u}$ are the UA decision variable, the noise power, the binary scheduling variable indicates whether RB(f_i, t_i) is assigned to u-th UE through RU m, the transmission power from RU m to UE u in RB(f_i, t_i), and the sum of interfering signals' power from adjacent RUs that are using the same RB(f_i, t_i), respectively. Since RBs are allocated to each slice, there is no interference between signals transmitted over different slices. We note that if $\pi^{f_i,t_i}_{m,u} = 1$, RB(f_i, t_i) of RU m allocated to the u-th UE; otherwise $\pi^{f_i,t_i}_{m,u} = 0$, in which $\sum_u \pi^{f_i,t_i}_{m,u} \leq 1; \forall m, t_i, f_i$. Considering interference from other RUs using the same RB (intra-slice), $I^{f_i,t_i}_{m,u}$ for a generic service can be represented as

$$I^{f_i,t_i}_{m,u} = \sum_{m' \in \mathcal{M}_n \setminus \{m\}} \sum_{u' \in \mathcal{U}_{m'}} \pi^{f_i,t_i}_{m',u'} p^{f_i,t_i}_{m',u'} g^{f_i,t_i}_{m',u} \tag{6.2}$$

where $\mathcal{U}_{m'}$ is the set of UEs served by RU m'.

6.2.3 Constraints

Power-related Constraints: The total transmission power assigned to all scheduled RBs occupied by m-th RU is subject to $\sum_{i,f_i,u} p_{m,u}^{f_i,t_i} \leq P_m^{\mathsf{max}}$ should not be greater than each RU power budget, while $p_{m,u}^{f_i,t_i} \geq 0$. Thus, the set of the power-related constraints is defined as

$$\mathscr{P}[t] \triangleq \{\boldsymbol{p}_{m,u}[t]|_{\forall m,u} \Big| p_{m,u}^{f_i,t_i} \geq 0, \sum_{i,f_i,u} p_{m,u}^{f_i,t_i} \leq P_m^{\mathsf{max}}|_{\forall m,t_i}\} \tag{6.3}$$

where $\boldsymbol{p}_{m,u}[t] \triangleq \big[p_{m,u}^{f_i,t_i}\big]^T$ and $\boldsymbol{p}[t] \triangleq \big[p_{m,u}[t]\big]^T$.

UA Constraints: Let $\varphi_{m,u} \in \{0,1\}$ be the UA decision variable. In particular, if $\varphi_{m,u} = 1$, RU m is selected to transmit data of the u-th data flow; otherwise, $\varphi_{m,u} = 0$. We define $\boldsymbol{\varphi}_u \triangleq \big[\varphi_{m,u}\big]_{\forall m}$ as the UA vector of the u-th UE, satisfying $\sum_{m \in \mathcal{M}_n} \varphi_{m,u} \leq 1; \forall u$. Hence, the global UA decision is formulated as

$$\boldsymbol{\Psi} \triangleq \Big\{\boldsymbol{\varphi}_u, \forall u \big| \sum_{m=1}^{M_n} \varphi_{m,u} \leq 1,\, \varphi_{m,u} \in \{0,1\},\, \forall m, u\Big\}. \tag{6.4}$$

eMBB QoS Constraints: To satisfy the eMBB users' QoS, each scheduled user must achieve a minimum data rate, denoted by R^{min}, from the serving RU m per time-frame t as

$$R_u^{\mathsf{em}}(\boldsymbol{\varphi}, \boldsymbol{p}, \boldsymbol{\pi}) = \sum_{m=1}^{M_n} \sum_{t_i=1}^{T_i} R_{m,u}^{\mathsf{em}}(\boldsymbol{\varphi}, \boldsymbol{p}, \boldsymbol{\pi}) \geq R^{\mathsf{min}}. \tag{6.5}$$

Scheduling Constraints: We also impose a minimum number of RBs from the dedicated uRLLC slice to vacate the arrival packets of uRLLC users in the queue in the specific duration of time D^{ur} after the arrival time τ_u^{arv}, which is known in advance along with λ_u^{ur}. This is expressed as $\sum_{t_i=\tau_u^{\mathsf{arv}}/\delta_i}^{(D^{\mathsf{ur}}+\tau_u^{\mathsf{arv}})/\delta_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i} \geq \varepsilon_u^{\mathsf{ur}}[t]; \forall t, u \in \mathcal{U}^{\mathsf{ur}}, i = 2$; $\varepsilon_u^{\mathsf{ur}}[t] \triangleq \big\lceil \frac{\lambda_u^{\mathsf{ur}}[t] Z^{\mathsf{ur}}}{\rho.\kappa} \big\rceil$, where ρ and κ represent the number of symbols per RB and the number of bits per symbol, respectively. Given the selected MCS [150], the SINR must be greater than a given threshold Γ^{th}:

$$\Gamma_{m,u}^{f_i,t_i} \triangleq \frac{\pi_{m,u}^{f_i,t_i} p_{m,u}^{f_i,t_i} g_{m,u}^{f_i,t_i}}{N_0 + I_{m,u}^{f_i,t_i}} \geq \pi_{m,u}^{f_i,t_i} \Gamma^{\mathsf{th}}; \forall f_i, t_i, m, u \in \mathcal{U}^{\mathsf{ur}}. \tag{6.6}$$

To optimize radio resource use, especially under low uRLLC traffic, we propose a slice-aware strategy over the traditional isolated allocation method. This approach allows slices to share resources within set constraints, enabling cross-numerology traffic access. As uRLLC traffic increases, the eMBB slice can adjust by allocating additional RBs, and vice versa, improving eMBB data rate by using underutilized uRLLC resources. While more complex, this method enhances resource efficiency by dynamically adjusting distribution based on traffic demands. To prioritize uRLLC services, we consider $\sum_{t_i=\tau_u^{\mathsf{arv}}/\delta_i}^{(D^{\mathsf{ur}}+\tau_u^{\mathsf{arv}})/\delta_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i} \geq e_u^{\mathsf{ur}}[t], \forall t, u \in \mathcal{U}^{\mathsf{ur}}, i = 1$, where $e_u^{\mathsf{ur}}[t] = \big\lfloor \big(\varepsilon_u^{\mathsf{ur}}[t] - \Omega_u[t]\big)^+ \big\rfloor$, $\Omega_u[t] \triangleq \big(\varepsilon_u^{\mathsf{ur}}[t] / \sum_{u \in \mathcal{U}^{\mathsf{ur}}} \varepsilon_u^{\mathsf{ur}}[t]\big)(F_i \times D^{\mathsf{ur}}/\delta_i)|_{i=2}$, and $(x)^+ \triangleq \max\{x, 0\}$. In contrast, the

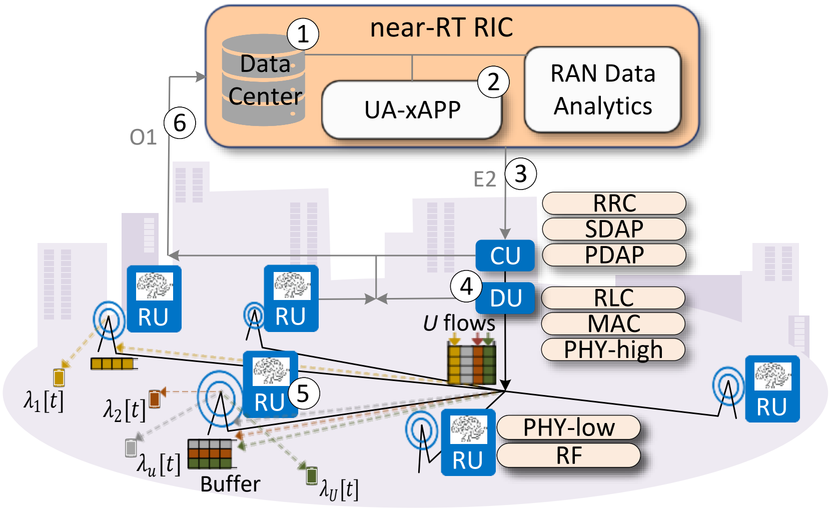


FIGURE 6.2: High-level structure of deploying the proposed intelligent user association, congestion control and resource scheduling considering 7.2x FS in Open RAN architecture.

following constraint allocates all unused RBs of other slices to eMBB users to maximize the data rate as $\sum_{t_i}\sum_{f_i}\pi_{m,u}^{f_i,t_i} \geq e_u^{\mathsf{em}}[t],\ \forall t, u \in \mathcal{U}^{\mathsf{em}}, i=2$, where $e_u^{\mathsf{em}}[t] = \Big(\lfloor (F_i \times T_i - \sum_{u\in\mathcal{U}^{\mathsf{ur}}} \min(\varepsilon_u^{\mathsf{ur}}[t], \Omega_u[t]))/U^{\mathsf{em}}\rfloor\Big)^+|_{i=2}$ is the slice awareness for the mixed numerologies in the frequency domain. Herein, $\min(\varepsilon_u^{\mathsf{ur}}[t], \Omega_u[t])$ represents the number of RBs scheduled for the u-th uRLLC user in the t-th frame. Collecting all mentioned constraints makes the global scheduling vector as

$$\begin{aligned}\mathbf{\Lambda}[t] \triangleq \Big\{ \boldsymbol{\pi}_{m,u}[t],\ \forall m,u \Big| \pi_{m,u}^{f_i,t_i} \in \{0,1\}, \sum_{u=1}^{U} \pi_{m,u}^{f_i,t_i} \leq 1; \forall m, t_i, f_i,\\ \sum_{t_i=\tau_u^{\mathsf{arv}}/\delta_i}^{(D^{\mathsf{ur}}+\tau_u^{\mathsf{arv}})/\delta_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i} \geq \varepsilon_u^{\mathsf{ur}}[t]; \forall t, u \in \mathcal{U}^{\mathsf{ur}}, i=2,\\ \sum_{t_i=\tau_u^{\mathsf{arv}}/\delta_i}^{(D^{\mathsf{ur}}+\tau_u^{\mathsf{arv}})/\delta_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i} \geq e_u^{\mathsf{ur}}[t], \forall t, u \in \mathcal{U}^{\mathsf{ur}}, i=1,\\ \sum_{t_i=1}^{T_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i} \geq e_u^{\mathsf{em}}[t],\ \forall t, u \in \mathcal{U}^{\mathsf{em}}, i=2 \Big\}\end{aligned} \tag{6.7}$$

where $\boldsymbol{\pi}_{m,u}[t] \triangleq \big[\pi_{m,u}^{f_i,t_i}\big]^T$ and $\boldsymbol{\pi}[t] \triangleq \big[\boldsymbol{\pi}_{m,u}[t]\big]^T; \forall m, u$.

Buffering Queue & e2e uRLLC Latency: As depicted in Fig. 6.2.3, U independent data flows in CU are steered to DUs, where the processing queue follows the $M/M/1$ model to serve packets on a first-come-first-serve basis. The queue length [bits] of waiting data flows in the buffer of the m-th RU in the t-th frame is computed as $q^{\mathsf{x}}_m[t] = \sum_u \big(\varphi_{m,u}\lambda^{\mathsf{x}}_u[t]Z^{\mathsf{x}}\Delta + \sum_{t_i} q^{\mathsf{x}}_{m,u}[t_i]\big)$, where $q^{\mathsf{x}}_{m,u}[t_i] = (q^{\mathsf{x}}_{m,u}[t_i - 1] - R^{\mathsf{x}}_{m,u}(\boldsymbol{\varphi}, \boldsymbol{\pi}, \boldsymbol{p})\delta_i)^+$, and $\varphi_{m,u}\lambda^{\mathsf{x}}_u[t]Z^{\mathsf{x}}$ [bits/frame] is the data flow of UE u at RU m.

The e2e latency of the u-th uRLLC UE at time-frame t consists of the propagation (τ^{prog}), processing (τ^{proc}), and transmission (τ^{tx}) delays, which can be written as

$$\tau^{\mathsf{ur}}_u = \tau^{\mathsf{prog}} + \tau^{\mathsf{tx}} + \tau^{\mathsf{proc}} \tag{6.8}$$

where τ^{prog} is the time of a data flow to reach its destination, including the propagation latency in FH, midhaul, and backhaul links. In addition, τ^{proc} is the sum of all processing time at CU, DU, and RU, such as $\tau^{\mathsf{proc}}_{cu} = \sum_u \lambda^{\mathsf{x}}_u[t]/\mu_{cu}$ and $\tau^{\mathsf{proc}}_{du} = \sum_u \lambda^{\mathsf{x}}_u/\mu_{du}$, where μ_{cu} and μ_{du} are the task rates [1/sec] at CU and DU, respectively; and $\tau^{\mathsf{proc}}_{ru}$ is limited by the three OFDM symbols duration, which is commonly very small and refers to the equipment computing capacity [151]. Considering propagation and processing latencies may be redundant because: *i*) The 300 km distance between the core network and UEs makes propagation delay negligible [152], and *ii*) CU and DU's high computing capacity minimize processing latency. Thus, these factors do not affect the optimization. Thus, we focus on transmission latency, which encompasses the time to serve packets across the midhaul, FH, and RU-UE links. Given the high speed of midhaul and FH links, their latency is negligible. This chapter assumes random packet arrivals, with uRLLC latency mainly determined by RU-UE transmission time, calculated as the gap (in TTIs) between packet arrival time τ^{arv}_u and service time $\tau^{\mathsf{tx}}_u = \delta_i.\operatorname{argmax}_{t_i}\{\pi^{f_i,t_i}_{m,u}[t_i]\}; \forall u \in \mathcal{U}^{\mathsf{ur}}$. Since eMBB and uRLLC services are separate, eMBB queues do not affect uRLLC queues. uRLLC UEs are prioritized and served immediately, making the total uRLLC latency roughly equal to the RU-UE link transmission latency: $\tau^{\mathsf{ur}}_u(\boldsymbol{\pi}) \approx \max(\tau^{\mathsf{tx}}_u, \tau^{\mathsf{arv}}_u)$.

To ensure that the u-th uRLLC UE meets its minimum latency requirement, the e2e latency must be bounded by a predefined threshold D^{ur}. Specifically, each packet must be transmitted no later than D^{ur} after arriving at the buffer:

$$\tau^{\mathsf{arv}}_u \le \tau^{\mathsf{ur}}_u(\boldsymbol{\pi}) \le D^{\mathsf{ur}} + \tau^{\mathsf{arv}}_u \tag{6.9}$$

Congestion Control Constraints: Given eMBB's larger packet sizes compared to uRLLC, we focus on eMBB queue length for congestion control. To minimize handovers during congestion and maintain stability, using the average queue length over W lag frames may be an effective approach, *i.e.*, $\bar{q}^{\mathsf{em}}_m \triangleq \frac{1}{W}\sum_{t-W}^{t} q^{\mathsf{em}}_m[t]$. To minimize packet loss due to buffer overflow, the queue length is constrained as

$$\boldsymbol{q}^{\mathsf{em}}[t] \triangleq \big\{q^{\mathsf{em}}_m | \bar{q}^{\mathsf{em}}_m \le q^{\mathsf{max}}_m, \forall m, t\big\}. \tag{6.10}$$

We note that $\bar{q}^{\mathsf{em}}_m \le q^{\mathsf{max}}_m$ is imposed to ensure that the arrival packets in the m-th RU buffer should not exceed the maximum buffer size of q^{max}_m.

6.3 Intelligent User Association, Congestion Control, and Resource Scheduling Problem

6.3.1 Problem Statement

In this section, we present an optimization problem designed to minimize both eMBB queue length and worst-case uRLLC latency by utilizing 7.2x FS within an Open RAN architecture, ensuring compliance with each service's SLAs. The utility function is defined as

$$\omega \sum_{t_i,m,u\in\mathcal{U}^{\mathsf{em}}} \frac{q^{\mathsf{em}}_{m,u}[t_i]}{q_0} + (1-\omega)\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\frac{\tau^{\mathsf{ur}}_u}{\tau_0}\} \tag{6.11}$$

where $\omega \in [0,1]$ is a regulatory factor balancing the influence of queue length and latency. In addition, $q_0 > 0$ and $\tau_0 > 0$ are reference values for eMBB queue length and uRLLC latency, respectively. The intelligent UA decisions, congestion control, and resource scheduling (IUCR) optimization problem is then formulated as follows:

$$\textbf{IUCR:} \min_{\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p}} \quad \omega \sum_{t_i,m,u\in\mathcal{U}^{\mathsf{em}}} \frac{q^{\mathsf{em}}_{m,u}[t_i]}{q_0} + (1-\omega)\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\frac{\tau^{\mathsf{ur}}_u}{\tau_0}\} \tag{6.12a}$$

$$\mathsf{s.t.} \quad (6.3),(6.4),(6.5),(6.6),(6.7),(6.9), \text{and}\,(6.10). \tag{6.12b}$$

6.3.2 Challenges of Solving Problem (6.12)

The main challenge in solving problem (6.12) is that $\boldsymbol{p}$ and $\boldsymbol{\pi}$ cannot be jointly optimized because different units manage their optimization under the 7.2x FS framework [153]. Specifically, higher-layer functions in CUs handle scheduling, modulation, and RAN slicing, while power control is managed by lower-layer functions in RUs. Optimizing the UA variables $\boldsymbol{\varphi}$ requires a network-wide view and historical data to minimize handovers. Since packet arrivals for both services vary dynamically and high traffic demands may necessitate adjustments to $\boldsymbol{\varphi}$ to avoid cell congestion, frequent updates are impractical. Instead, $\boldsymbol{\varphi}$ is updated only when congestion is detected, using a heuristic method to optimize RU-UE connections.

From a mathematical perspective, the binary nature of scheduling and UA decision variables in constraints (6.4), (6.5)-(6.10), and (6.9), combined with the product term $\varphi_{m,u}.\pi^{f_i,t_i}_{m,u}.p^{f_i,t_i}_{m,u}$ in the objective function and some constraints, results in a mixed-integer non-convex program (MINCP). This makes the problem difficult to solve directly. Standard solvers like *Gurobi*, *Mosek*, and *SCA* often fail to provide (near)-optimal solutions due to incomplete queue data, variable network conditions, and RU interference. Additionally, each RU only has local observations without data from neighboring RUs. The 7.2x FS framework in the Open RAN architecture balances centralization and distribution to meet SLAs. Thus, RB allocation is managed in the CU based on general feedback to handle congestion and optimize resource usage, while power control is handled locally at RUs. To ensure high

QoE in each frame, an efficient hierarchical approach combining heuristics, optimization, and DRL is needed to solve problems (6.12).

6.3.3 High-Level IUCR Scheme in 7.2x FS-Based Open RAN Architecture

In this section, we outline the high-level organization and end-to-end flow for deploying the IUCR scheme, aligned with the Open RAN architecture and the 7.2x FS standards recommended by the O-RAN Alliance, as illustrated in Fig. 6.2.3.

① **Data Storage:** Essential data, including queue length, traffic demands, and network conditions, is collected from RAN components and stored in the data center at RIC for preprocessing.

② **User Association:** Utilizing the near-RT control loop (Fig. 6.2.1), the UA-xAPP heuristic algorithm in the near-RT RIC efficiently assigns UEs to RUs, preventing congestion. This centralization enhances awareness of traffic loads, queue lengths, SLAs, and network conditions, ensuring efficient RAN coordination.

③ **Policies/Actions Transfer:** Solutions for binary UA variables and updated queue lengths are transmitted to the CU through the near-RT closed-loop controls.

④ **Resources Scheduling:** In line with 7.2x FS, RB scheduling is managed by the higher PHY layer at the DU, closest to the antennas. The DU uses policies inferred from the near-RT RIC to dynamically adjust RB allocation based on traffic demands, minimizing latency for critical applications like uRLLC.

⑤ **Power Control:** Following 7.2x FS, power control is distributed across RUs using local information. Each RU employs DRL models and trains two DQNs for power control, optimizing UE-centric power allocation and reducing interference. UEs estimate CSI and potential interference, providing feedback to the assigned RU.

⑥ **Continuous Monitoring:** Updated data on traffic demands, network conditions, and queue lengths is sent to the RIC via the E2 interface, facilitating model updates in subsequent frames to maintain SLAs and monitor network performance.

Algorithm 6 outlines the steps for solving problem (6.12).

6.4 Proposed Solutions

This section introduces algorithms to solve problem (6.12). The optimal resource scheduling policy requires accurate UA decision variable estimations based on cloud-stored RAN data. Given random traffic demand per frame, the objective is to use historical system data via open interfaces for more efficient response handling.

Algorithm 6 Proposed Algorithm to Solve ICUR Problem (6.12)

```
Initialization: Set $t = 1$; Initialize $\varphi_{m,u}$ with each UE connecting to the nearest RU; the queue $q^{\mathsf{x}}_{m,u}[1]$ is initialized randomly; and the coordinates of UEs and RUs are denoted by $(\boldsymbol{x}^{\mathsf{UE}}, \boldsymbol{y}^{\mathsf{UE}})$, and $(\boldsymbol{x}^{\mathsf{RU}}, \boldsymbol{y}^{\mathsf{RU}})$, respectively.
for $t = 1, 2, \ldots, T$ do
    UA-xAPP: Given the historical data from previous $W$ frames: The average queue length $\bar{q}^{\mathsf{x}}_{m,u} \triangleq \frac{1}{W}\sum_{t-W}^{t} q^{\mathsf{x}}_{m,u}[t]$, $\varphi_{m,u}[t]$ and the network conditions;
    if $\bar{q}^{\mathsf{x}}_{m} = \sum_u \bar{q}^{\mathsf{x}}_{m,u} \geq q^{\mathsf{th}}$ then
        Run Algorithm 7 to update the UA matrix $\boldsymbol{\varphi}$;
    end if
    RB Scheduler: Given the optimal solution $\boldsymbol{\varphi}^*$, the estimated channel gain feedback from RUs to CU, uniform power allocation, and large-scale interference, $\boldsymbol{\pi}$ is optimized by solving problem 6.13 using Algorithm 8;
    Power Control: The proposed D3QN method presented in Algorithm 9 uses $M_n$ agents - each for one RU to predict the transmission power allocated to each RB based on the optimal solutions $\boldsymbol{\varphi}^*$ and $\boldsymbol{\pi}^*$ as well as the local observation of each agent.
    Update queue length as $q^{\mathsf{x}}_{m,u}[t] = (q^{\mathsf{x}}_{m,u}[t-1] + \varphi^*_{m,u}\lambda^{\mathsf{x}}_u[t] Z^{\mathsf{x}}\Delta - \sum_{t_i}(R^{\mathsf{x}}_{m,u}(\boldsymbol{\varphi}^*, \boldsymbol{p}^*[t_i], \boldsymbol{\pi}^*)\delta_i))^+$;
end for
```

6.4.1 UA-xAPP

Heuristic methods use available data for decision-making based on rules and experience. Although less dynamic than ML algorithms, they are vital for network optimization in environments with limited communication between decentralized and centralized components, aligning with Open RAN principles. Effective for latency-sensitive applications, these methods enable quick decisions despite interface constraints. We propose a heuristic approach for assigning each UE to the appropriate RU, considering traffic demands, queue lengths, and network conditions to meet SLAs. The model is UE-centric with dynamic traffic and fluctuating packet arrivals. To ensure smooth handovers and minimize frequency during high traffic, the comparison threshold is based on average dynamics rather than immediate gains. The SLA metric, defined as the average queue length, facilitates continuous monitoring. Network alarms trigger when congestion nears, prompting the UA-xAPP to update variables. Updates to $\boldsymbol{\varphi}$ are based on average queue length and network conditions, as outlined in Algorithm 7.

In Algorithm 7, UEs are initially linked to the nearest RU within a circular region, assuming a random distribution. The algorithm identifies congested RUs and reassigns UEs to maintain data rates and alleviate congestion. Congestion is detected when the average RU's backlog buffer exceeds the given threshold q^{th}. A scoring mechanism selects candidate UEs for reassignment based on their proximity to other RUs with available buffer capacity.

6.4.2 RB Scheduling

UA-xAPP sends the UA decision variables ($\boldsymbol{\varphi}^*$) to RAN components via the E2 interface for execution. Thanks to the closed-loop controls for providing feedback on channel conditions, efficient RB scheduling enables at CU to meet SLAs. Referring to (6.12), we can optimize RB scheduling ($\boldsymbol{\pi}$) under uniform power allocation and large-scale interference. As higher

Algorithm 7 Heuristic-based UA-xAPP Algorithm

```
Initialization: Set $U' = \emptyset$, $M' = \emptyset$, and $\mathsf{S} = \mathbf{0}_{M_n \times U^{\mathsf{em}}}$ as the sets of reconnected UEs, congested RUs, and scores, respectively.
while $\bar{q}^{\mathsf{em}}_m \geq q^{\mathsf{th}}$ do
    $M' \leftarrow M' \cup \{m\}$: measure score $\mathsf{S}_{m',u}$ of RU $m'$ and all UEs (i.e., $u \in U_m$);
    for $m' = 1, 2, \ldots, M_n \backslash \{m\}$ do
        for $u = 1, 2, \ldots, U_m$ do
            Calculate the normalized distance: $d_{m',u}$ and the normalized average queue length $\bar{q}^{\mathsf{em}}_{m'}$ and $\bar{q}^{\mathsf{em}}_u \triangleq \sum_m \bar{q}^{\mathsf{em}}_{m,u}$;
            Update $\mathsf{S}_{m',u} := \epsilon_1(\frac{1}{1+d_{m',u}}) + \epsilon_2(\frac{1}{1+\bar{q}^{\mathsf{em}}_{m'}}) + \epsilon_3(\frac{1}{1+\frac{1}{\bar{q}^{\mathsf{em}}_u}})$, where $\epsilon_k$ is a priority factor with $\sum_k \epsilon_k = 1$;
        end for
    end for
    Update UA variables with the maximum score $\varphi^*_{\{\max_{m',u} \mathsf{S}\}} = 1$;
    From Algorithm 6, check $\bar{q}^{\mathsf{em}}_m$;
end while
Outputs: $\boldsymbol{\varphi}^*$.
```

eMBB data rates reduce queues and prevent congestion, the utility function focuses on maximizing the eMBB data rate. Thus, we reformulate the optimization to maximize eMBB data rates while minimizing worst-user uRLLC latency, solving for the optimal RB scheduling variables:

$$\max_{\boldsymbol{\pi}} \Big(\omega \sum_{t_i, m, u \in \mathcal{U}^{\mathsf{em}}} \Big\{ \frac{R^{\mathsf{em}}_{m,u}}{R_0} \Big\} - (1-\omega) \max_{u \in \mathcal{U}^{\mathsf{ur}}} \Big\{ \frac{\tau^{\mathsf{ur}}_u}{\tau_0} \Big\} \Big) \tag{6.13a}$$

$$\mathsf{s.t.} \quad (6.5), (6.6), (6.7), (6.9), \text{ and } (6.10) \tag{6.13b}$$

where R_0 is the reference data rate of eMBB. Problem (6.13) is non-convex due to the non-concavity of the first term of the objective function and non-convexity of constraints (6.5), (6.6) and (6.10). The binary nature of the RB allocation variables makes (6.13) even more complex to solve. To address this, we relax the binary variables, *i.e.*, $\pi^{f_i,t_i}_{m,u} \in [0,1[$, forming the new global scheduling vector $\tilde{\mathbf{\Lambda}}[t]$. We then reformulate constraints (6.5), (6.6), and (6.10) into more tractable forms, which can be efficiently solved by an SCA-based iterative algorithm.

Penalty function: To reduce the uncertainty of the relaxed variables, we introduce the penalty function $\mathcal{P}(\boldsymbol{\pi}) = \sum_{m,u,f_i,t_i} \big((\pi^{f_i,t_i}_{m,u})^2 - \pi^{f_i,t_i}_{m,u}\big)$, which is convex in $\boldsymbol{\pi}$. Since $\mathcal{P}(\boldsymbol{\pi}) \leq 0$ for any $\pi^{f_i,t_i}_{m,u} \in [0,1]$, it effectively penalizes relaxed variables, driving them toward near-binary solutions at the optimum (*i.e.*, satisfying 6.7). Furthermore, We introduce slack variables $\boldsymbol{z} = \big[z^{f_i,t_i}_{m,u}\big]^T$ to make the first term of the objective function more tractable. Incorporating the penalty function and auxiliary variables, the parameterized relaxed problem is expressed as

$$\max_{\boldsymbol{\pi},\boldsymbol{z}} \Big(\omega \frac{\sum_{m,u,f_i,t_i \in \mathcal{U}^{\mathsf{em}}} z_{m,u}^{f_i,t_i}}{R_0} - (1-\omega) \max_{u \in \mathcal{U}^{\mathsf{ur}}} \Big\{ \frac{\tau_u^{\mathsf{ur}}}{\tau_0} \Big\} + \nu \mathcal{P}(\boldsymbol{\pi}) \Big) \tag{6.14a}$$

$$\text{s.t.} \quad \boldsymbol{\pi} \in \tilde{\boldsymbol{\Lambda}}[t] \tag{6.14b}$$

$$\log_2 \Big(1 + \frac{\pi_{m,u}^{f_i,t_i} p_{m,u}^{f_i,t_i} g_{m,u}^{f_i,t_i}}{N_0 + I_{m,u}^{f_i,t_i}}\Big) \geq z_{m,u}^{f_i,t_i}, \, \forall u \in \mathcal{U}^{\mathsf{em}} \tag{6.14c}$$

$$(6.5), (6.6), (6.9), \text{ and } (6.10) \tag{6.14d}$$

where $\nu > 0$ is a penalty constant.

Proposition 3. *With an appropriate positive value of* ν*, problems (6.13) and (6.14) share the same optimal solution* $\boldsymbol{\pi}^*$*.*

The detailed proof can be found in [79]. In problem (6.14), the objective function is non-concave due to $\mathcal{P}(\boldsymbol{\pi})$, while constraints (6.14c), (6.5), (6.6), and (6.10) are non-convex. By SCA principles, the function $\mathcal{P}(\boldsymbol{\pi})$ is linearized at iteration j by the first-order Taylor approximation as

$$\mathcal{P}^{(j)}(\boldsymbol{\pi}) \triangleq \sum_{m,u,f_i,t_i} \Big[\pi_{m,u}^{f_i,t_i} \big(2(\pi_{m,u}^{f_i,t_i})^{(j)} - 1\big) - \big((\pi_{m,u}^{f_i,t_i})^{(j)}\big)^2 \Big] \tag{6.15}$$

where $\mathcal{P}(\boldsymbol{\pi}) \geq \mathcal{P}^{(j)}(\boldsymbol{\pi})$ and $\mathcal{P}(\boldsymbol{\pi}^{(j)}) \geq \mathcal{P}^{(j)}(\boldsymbol{\pi}^{(j)})$. To deal with the non-convexity of (6.14c), we first rewritten it as follows:

$$\log_2 \Big(N_0 + I_{m,u}^{f_i,t_i} + \pi_{m,u}^{f_i,t_i} p_{m,u}^{f_i,t_i} g_{m,u}^{f_i,t_i} \Big) \geq z_{m,u}^{f_i,t_i} + f(\pi_{m,u}^{f_i,t_i}) \tag{6.16}$$

where $f(\pi_{m,u}^{f_i,t_i}) \triangleq \log_2 \big(N_0 + I_{m,u}^{f_i,t_i}\big)$ in (6.16). We approximate $f(\pi_{m,u}^{f_i,t_i})$ at iteration j as

$$f^{(j)}(\pi_{m,u}^{f_i,t_i}) \triangleq f((\pi_{m,u}^{f_i,t_i})^{(j)}) + \frac{(\pi_{m,u}^{f_i,t_i} - (\pi_{m,u}^{f_i,t_i})^{(j)})}{\ln 2(N_0 + (I_{m,u}^{f_i,t_i})^{(j)})} \times \Big(\sum_{m' \in \mathcal{M}_n \setminus \{m\}} \sum_{u' \in \mathcal{U}_{m'}} p_{m',u'}^{f_i,t_i} g_{m',u}^{f_i,t_i} \Big) \tag{6.17}$$

where $(I_{m,u}^{f_i,t_i})^{(j)} \triangleq I_{m,u}^{f_i,t_i}((\pi_{m,u}^{f_i,t_i})^{(j)})$. Now, constraint (6.14c) is iteratively replaced by

$$\log_2 \Big(N_0 + I_{m,u}^{f_i,t_i} + \pi_{m,u}^{f_i,t_i} p_{m,u}^{f_i,t_i} g_{m,u}^{f_i,t_i} \Big) \geq z_{m,u}^{f_i,t_i} + f^{(j)}(\pi_{m,u}^{f_i,t_i}), \tag{6.18}$$

which is convex. Similarly, constraints (6.5) and (6.10) are iteratively replaced by

$$\sum_{m,f_i,t_i} \varphi_{m,u}^* \beta_i \log_2 \Big(N_0 + I_{m,u}^{f_i,t_i} + \pi_{m,u}^{f_i,t_i} p_{m,u}^{f_i,t_i} g_{m,u}^{f_i,t_i} \Big) \geq R^{\mathsf{th}} + \sum_{m,f_i,t_i} \varphi_{m,u}^* \beta_i f^{(j)}(\pi_{m,u}^{f_i,t_i}) \tag{6.19}$$

Algorithm 8 The Proposed SCA-based Iterative Algorithm located at DU for Solving (6.13)

```
**Initialization:** Set $j = 0$ and generate initial feasible point for $\boldsymbol{\pi}^{(0)}[t] := \boldsymbol{\pi}[t-1]$ to constraints in (6.22).
**Repeat**
Solve (6.22) to obtain $(\boldsymbol{\pi}^*, \boldsymbol{y}^*, \boldsymbol{z}^*)$ and $\Xi^*$;
Update $\boldsymbol{\pi}^{(j)} := \boldsymbol{\pi}^*$ and $\Xi^{(j)} := \Xi^*$;
Set $j := j + 1$;
**Until** Convergence or $|\Xi^{(j)} - \Xi^{(j-1)}| \leq \varepsilon$ {/**Satisfying a given accuracy level**/}
Recover an exact binary by computing $\boldsymbol{\pi}^* = \lfloor \boldsymbol{\pi}^{(j)} + 0.5 \rfloor$.
**Output:** $\boldsymbol{\pi}^*$.
```

and

$$\sum_u \varphi^*_{m,u} \sum_{f_i,t_i} \beta_i \log_2\Big(N_0 + I^{f_i,t_i}_{m,u} + \pi^{f_i,t_i}_{m,u} p^{f_i,t_i}_{m,u} g^{f_i,t_i}_{m,u}\Big) \geq \phi + \sum_{m,f_i,t_i} \varphi^*_{m,u}\beta_i f^{(j)}(\pi^{f_i,t_i}_{m,u}) \tag{6.20}$$

respectively, where $\phi \triangleq \big(q^{\mathsf{max}}_m - q^{\mathsf{em}}_m[t-1] - \sum_u \varphi^*_{m,u}\lambda^{\mathsf{em}}_u[t] Z^{\mathsf{em}}\Delta\big)^+$. Finally, to handle the non-convexity of constraint (6.6), we introduce the auxiliary variables $\boldsymbol{y} = [y^{f_i,t_i}_{m,u}]^T$, satisfying $y^{f_i,t_i}_{m,u} \leq I^{f_i,t_i}_{m,u}$, to approximate it as

$$\begin{aligned}\big(y^{f_i,t_i}_{m,u} + \pi^{f_i,t_i}_{m,u}\big)^2 \leq 2\pi^{f_i,t_i}_{m,u}\big(\frac{p^{f_i,t_i}_{m,u} g^{f_i,t_i}_{m,u}}{\Gamma^{\mathsf{th}}} - N_0\big) + 2(y^{f_i,t_i}_{m,u})^{(j)} y^{f_i,t_i}_{m,u} + 2\big(\pi^{f_i,t_i}_{m,u}\big)^{(j)}\pi^{f_i,t_i}_{m,u} - \\ \big((y^{f_i,t_i}_{m,u})^{(j)}\big)^2 - \big((\pi^{f_i,t_i}_{m,u})^{(j)}\big)^2.\end{aligned} \tag{6.21}$$

Bearing all the above developments in mind, the convex approximate program (6.14) is solved at iteration j is given as

$$\max_{\boldsymbol{\pi},\boldsymbol{y},\boldsymbol{z}} \quad \Xi^{(j)} \triangleq \omega \frac{\sum_{m,u,f_i,t_i \in \mathcal{U}^{\mathsf{em}}} z^{f_i,t_i}_{m,u}}{R_0} - (1-\omega)\max_{u\in\mathcal{U}^{\mathsf{ur}}}\Big\{\frac{\tau^{\mathsf{ur}}_u}{\tau_0}\Big\} + \nu\mathcal{P}^{(j)}(\boldsymbol{\pi}) \tag{6.22a}$$

$$\mathsf{s.t.} \quad \boldsymbol{\pi} \in \tilde{\boldsymbol{\Lambda}}[t] \tag{6.22b}$$

$$y^{f_i,t_i}_{m,u} \leq I^{f_i,t_i}_{m,u},\ \forall m, f_i, t_i, u \in \mathcal{U}^{\mathsf{ur}} \tag{6.22c}$$

$$(6.9), (6.18), (6.19), (6.20), \text{and}\, (6.21). \tag{6.22d}$$

The proposed SCA-based iterative algorithm to solve (6.13) assuming large-scale interference channel gains is summarized in Algorithm 8.

6.4.3 Power Allocation

The next step is to allocate power to each RB assigned to eMBB and/or uRLLC services. Due to incomplete information on channel conditions and interference, each RU relies only on local observations, *i.e.*, the channel conditions of its served UEs based on the estimated

$\boldsymbol{\varphi}^*$ without knowledge of interference from adjacent RUs. This scenario requires a decentralized approach for power allocation, where each RU aims to maximize its utility function in a distributed optimization process. However, channel interference and resource limitations may hinder optimal performance due to decisions made by other RUs. To address this, the paper proposes two methods for power control: 1) Distributed power optimization and 2) DRL-based power control.

1) Distributed Interference-Aware Power Optimization (DIPO): This approach aims to determine the power allocation for each RU in a distributed manner using optimization solvers. It assumes uniform power allocation $\left(p_{m,u}^{f_i,t_i} \triangleq P_m^{\mathsf{max}} / \sum_i F_i\right)$ as the interference power from adjacent RUs and accounts for large-scale interference channel gains. Each RU m optimizes its power allocation by solving the following optimization problem,

$$\max_{\boldsymbol{p_m}} \sum_{t_i, u \in \mathcal{U}_m^{\mathsf{em}}} R_u^{\mathsf{em}}, \quad \forall m \in \mathcal{M}_n \tag{6.23a}$$

$$\mathsf{s.t.} \quad (6.3), (6.5), (6.6), \text{and} \, (6.10) \tag{6.23b}$$

which relies only on its channel gains $\boldsymbol{G}_m$ and the optimal RB scheduling of its served UEs $\boldsymbol{\pi}_m^*$.

2) DRL-based Power Control (DRL-PC): Since each RU operates with limited network information, it must infer other RUs when making decisions. This challenge highlights the need for an efficient mechanism on the RU side to assess the mutual impacts of decisions across RUs. An intelligent power control scheme based on DRL is crucial for tackling these issues, as DRL excels in sequential decision-making. By employing DRL, each RU can iteratively interact with the dynamic environment to derive an optimal power policy that maximizes its utility.

In operational Open RAN environments, the state transition probabilities of the Markov Decision Process (MDP) are often unavailable, requiring predictive estimation through environmental interactions. Our approach leverages DRL across a set of $\mathcal{M}_n$ agents, with each agent assigned to an RU. This enables each RU to detect hidden patterns based on local observations. To improve overall performance and reduce signaling overhead, RUs make decisions using an efficient reward function integrated with a flag system. Given the cooperative nature of the system, the flag mechanism allows distributed agents to collaborate effectively with minimal communication, as they share only their flags. Here, $\mathtt{F}_m[t] \in \{0, 1\}$ represents the flags of the agents in frame t.

Proposed D3QN-based Power Control at Each RU: The problem is commonly modeled as an MDP, where each agent $m \in \mathcal{M}_n$ learns through a transition tuple $(\mathbf{s}_m[t], \boldsymbol{a}_m[t], \mathbf{r}_m[t], \mathbf{s}_m[t+1])$. In this framework, agent m selects an action $\boldsymbol{a}_m[t]$ from its action space $\mathcal{A}_m[t]$ based on its current state $\mathbf{s}_m[t] \in \mathcal{S}_m$, with the joint action of all agents denoted as $\boldsymbol{a} = \{\boldsymbol{a}_1[t], \ldots, \boldsymbol{a}_{M_n}[t]\}$. Each agent then transitions to the next state $\mathbf{s}_m[t+1]$ and receives a reward $\mathbf{r}_m[t]$. It's important to note that if agents only receive distinct rewards, this may lead to selfish behavior, reducing overall network performance [129]. To enhance the overall system performance, we assume agents receive a common reward in addition to their individual rewards. Consequently, the state, action spaces, and reward functions are

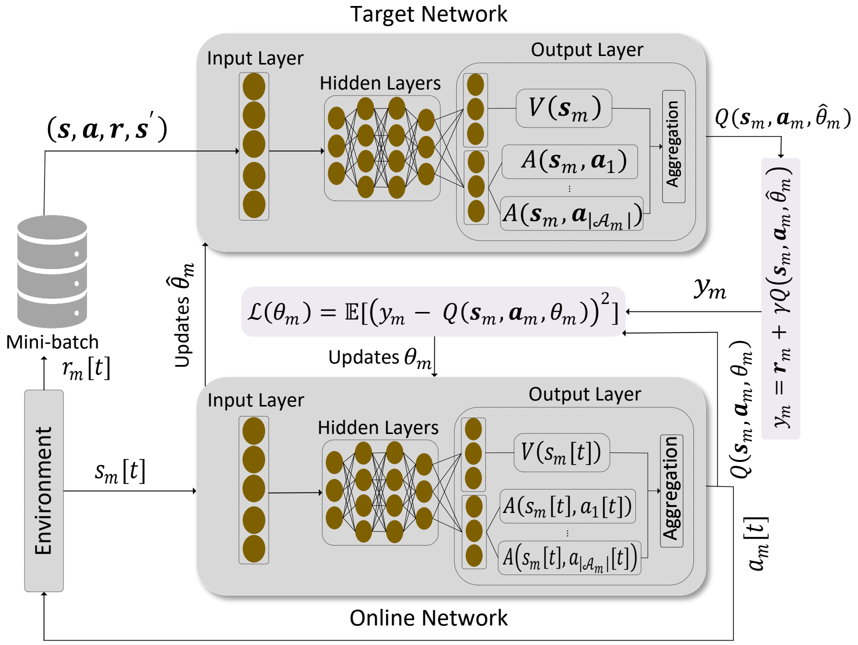

Target Network
Input Layer
Hidden Layers
Output Layer
$V(\boldsymbol{s}_m)$
$A(\boldsymbol{s}_m, \boldsymbol{a}_1)$
$A(\boldsymbol{s}_m, \boldsymbol{a}_{|\mathcal{A}_m|})$
Aggregation
$Q(\boldsymbol{s}_m, \boldsymbol{a}_m, \hat{\theta}_m)$
$(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{s}')$
Mini-batch
$r_m[t]$
Updates $\hat{\boldsymbol{\theta}}_m$
$\mathcal{L}(\theta_m) = \mathbb{E}[(y_m - Q(\boldsymbol{s}_m, \boldsymbol{a}_m, \theta_m))^2]$
y_m
Updates θ_m
$y_m = \boldsymbol{r}_m + \gamma Q(\boldsymbol{s}_m, \boldsymbol{a}_m, \hat{\theta}_m)$
$Q(\boldsymbol{s}_m, \boldsymbol{a}_m, \theta_m)$
Environment
$s_m[t]$
Input Layer
Hidden Layers
Output Layer
$V(s_m[t])$
$A(s_m[t], a_1[t])$
$A(s_m[t], a_{|\mathcal{A}_m|}[t])$
Aggregation
$a_m[t]$
Online Network

FIGURE 6.3: The proposed dueling double DQN (D3QN) agent.

defined as follows:

- *State*: The state of each agent m is designed based on its local observation and flag system. To further minimize the signalling overheads in information exchange between the agent and environment, the state of agent m in time-frame t can be defined as the combination of RBs scheduled to UEs served by RU m ($u \in \mathcal{U}_m$) and their channel gains, the action and flags in the previous time frame, which is expressed as

$$\mathbf{s}_m[t] = \Big\{\boldsymbol{\pi}_m[t] \times \boldsymbol{G}_m[t], \boldsymbol{a}_m[t-1], \mathtt{F}[t-1]\Big\}. \tag{6.24}$$

- *Action*: The discrete power domain has become a viable approach for learning-based dynamic scheduling in multi-traffic scenarios [154]. This method ensures stable convergence and reduces computational complexity for RUs with limited computational resources. In this context, we use a discrete action space where power is quantized into L levels, defined as $\hat{p}_l = lP_m^{\mathsf{max}}/L$ for $l = 1, \dots, L$, where $\hat{p}_l$ represents the l-th transmission power level. Assuming $K_m = \sum_i (T_i \times F_i)$ is the total number of RBs occupied by the m-th RU to serve its UEs at time frame t, the action for agent m at time frame t from its action space $\mathcal{A}_m$ is defined as

$$\boldsymbol{a}_m[t] = \Big\{\boldsymbol{a}^{1,1}, \dots, \boldsymbol{a}^{k_m,l}, \dots, \boldsymbol{a}^{K_m,L}\Big\} \tag{6.25}$$

where $\boldsymbol{a}^{k_m,l}$ indicates that agent m selects l-th transmission power level for RB index

Algorithm 9 The proposed D3QN-based Power Control

```
**Initialization:** Each RU builds two D3QNs, i.e., $Q(\mathbf{s}_m[t], \boldsymbol{a}_m[t]; \theta_m)$, and $Q(\mathbf{s}_m[t], \boldsymbol{a}_m[t]; \hat{\theta}_m)$ with random weights $\theta_m = \hat{\theta}_m$; set replay buffer $\mathcal{D}_m$ with capacity $C_m^{\mathsf{max}}, \forall m$, and reward values to $\mathbf{r}[t] = \mathbf{0}_{\boldsymbol{M_n}\times\mathbf{1}}$, and $\mathtt{F}[1] = \mathbf{0}_{\boldsymbol{M_n}\times\mathbf{1}}$.
**for** $e = 1, 2, \ldots, E$ **do**
    Initialize the network state $\mathbf{s}_m[t], \forall m$;
    **for** $t = 1, 2, \ldots, T$ **do**
        **for** $m = 1, 2, \ldots, M_n$ **do**
            All agents select their actions $\boldsymbol{a}_m[t] \in \mathcal{A}_m, \forall m$ based on $\epsilon$-greedy policy as
            Generate a random number rand();
            **if** rand() $< \epsilon$ **then**
                Random generating actions $\boldsymbol{a}_m[t]$;
            **else**
                Select actions $\boldsymbol{a}_m[t]$ predicted by both Q-networks;
            **end if**
            Agent $m$ takes its action and $\mathtt{F}_m[t] = 1$ if it meets constraint (6.3); Otherwise $\mathtt{F}_m[t] = 0$, then receive the reward $\mathbf{r}_m[t]$ as
            **if** $\prod_m \mathtt{F}_m[t] = 1$ **then**
                $\boldsymbol{r}_m[t] += r^{\mathsf{com}}$;
            **else if** $\mathtt{F}_m[t] = 1$ **then**
                $\boldsymbol{r}_m[t] += r^{\mathsf{indv}}$;
            **else**
                $\boldsymbol{r}_m[t] += r^{\mathsf{neg}}$;
            **end if**
            Update and store the transition of $(\mathbf{s}_m[t], \boldsymbol{a}_m[t], \mathbf{r}[t], \mathbf{s}_m[t+1])$ to the $m$-th replay memory.
        **end for**
        Sample random mini-batches from replay memory $m$;
        Update $\theta_m$ by minimizing the loss function $\mathcal{L}(\theta_m)$ (6.28); while update $\hat{\theta}_m$ every $C$ steps by resetting $\hat{\theta}_m = \theta_m$.
    **end for**
**end for**
```

k_m. Thus, the action space size of agent m is LK_m and the overall action space size of all agents is determined as $(LK_m)^{M_n}$.

- *Reward*: To design an efficient reward function, this chapter adopts a penalty-based approach that integrates constraints (6.5) and (6.6) while ensuring the sum of all distributed powers stays within the bound specified by constraint (6.3). A negative reward r^{neg} is given to any agent that fails to meet constraint (6.3) indicated by $\mathtt{F}_m[t] = 0$; Otherwise $\mathtt{F}_m[t] = 1$, and the agent earns an individual reward r^{indv} based on its action $\boldsymbol{a}_m[t]$ as $r^{\mathsf{indv}} \triangleq r_m[t] = \upsilon \frac{\sum_{u\in\mathcal{U}^{\mathsf{em}}}(R^{\mathsf{em}}_{m,u} - R^{\mathsf{min}})}{R_0} + (1-\upsilon)\frac{\sum_{u\in\mathcal{U}^{\mathsf{ur}}, f_i, t_i}(\Gamma^{f_i,t_i}_{m,u} - \pi^{f_i,t_i}_{m,u}\Gamma^{\mathsf{th}})}{\Gamma_0}$, where υ is a priority factor to balance both services and Γ_0 denotes the reference of uRLLC SINR. To boost overall network performance and prevent selfish behavior, a common reward $r^{\mathsf{com}} \triangleq \sum_m r_m[t]$ is shared among all agents if $\prod_m \mathtt{F}_m[t] = 1$. Throughout learning, agents (RUs) explore the environment to find optimal policies, despite limited observations like interference, ultimately leading to enhanced network performance.

Each agent aims to learn a policy $\Pi(\boldsymbol{a}[t]|\mathbf{s}[t])$ that maps observations to actions in order to maximize the expected total reward, represented by the action-value function:

$Q_{\Pi}(\mathbf{s}[t], \boldsymbol{a}[t]) \triangleq \mathbb{E}_{\Pi}(\sum_{l=t}^{T} \gamma^{(l-t)}\mathbf{r}[l]|\mathbf{s}, \boldsymbol{a})$, where $\sum_{l=t}^{T} \gamma^{(l-t)}\mathbf{r}[l]$ represents the accumulated discounted reward and $\gamma \in [0, 1]$ is the discount factor that balances future and immediate rewards. Once the optimal action-value function is determined: $Q^*(\mathbf{s}[t], \boldsymbol{a}[t]) = \max_{\Pi} Q_{\Pi}(\mathbf{s}[t], \boldsymbol{a}[t])$, the optimal policy $\Pi^*(\boldsymbol{a}[t]|\mathbf{s}[t]) = \text{argmax}_{\boldsymbol{a}[t]\in\mathcal{A}} Q^*(\mathbf{s}[t], \boldsymbol{a}[t])$ can be derived by acting greedily at each time step t.

Traditional DQN models use the same Q-value network for both action selection and evaluation, leading to an unstable learning process due to overly optimistic Q-value estimates. To address this and enhance convergence and learning efficiency, a novel multi-agent dueling double DQN (MAD3QN) is introduced in this work. As depicted in Fig. 6.4.3, each agent in MAD3QN builds its own D3QN model with two Q-networks: an online network with weights θ_m and a target network with $\hat{\theta}_m$. The last layer of each network is divided into two parts to evaluate the state value function $V(\mathbf{s}_m[t])$ and the advantage function $A(\mathbf{s}_m[t], \boldsymbol{a}_m[t])$. $V(\mathbf{s}_m[t])$ assesses the value of a state, helping the agent evaluate the potential long-term benefits of staying in that state. $A(\mathbf{s}_m[t], \boldsymbol{a}_m[t])$[t]) measures the relative advantage of an action within a state, guiding the agent to select the most favorable action. These two components are combined to produce the final Q-value function, which directs action selection in the environment:

$$Q(\mathbf{s}_m[t], \boldsymbol{a}_m[t]; \theta_m, \theta_m^V, \theta_m^A) = V(\mathbf{s}_m[t]) + A(\mathbf{s}_m[t], \boldsymbol{a}_m[t]) - \frac{1}{|\mathcal{A}_m|} \sum_{a\in\mathcal{A}_m} A(\mathbf{s}_m[t], \boldsymbol{a}_m[t]) \tag{6.26}$$

where θ_m^V and θ_m^A are the parameters according to state value function and advantage function parts, respectively. Note that the third term on the right-hand side of (6.26), representing the mean of the advantage function over all actions, ensures that the advantage function is centered around zero, facilitating easier network training. With these considerations, the updated equation for the Q-value function of agent m can be expressed as

$$Q(\mathbf{s}_m[t], \boldsymbol{a}_m[t]; \theta_m, \theta_m^V, \theta_m^A) \leftarrow (1-\eta)Q(\mathbf{s}_m[t], \boldsymbol{a}_m[t]; \theta_m, \theta_m^V, \theta_m^A) + \eta(y_m[t]) \tag{6.27}$$

where $\eta \in [0, 1]$ is the learning rate, and $Q(\mathbf{s}_m[t], \boldsymbol{a}_m[t]; \theta_m, \theta_m^V, \theta_m^A)$ and $Q(\mathbf{s}_m[t], \boldsymbol{a}_m[t]; \hat{\theta}_m, \hat{\theta}_m^V, \hat{\theta}_m^A)$ are the online Q-network and the target Q-network that are used to select actions and evaluate actions, respectively. In addition, $y_m[t] \triangleq \mathbf{r}_m[t] + \gamma Q(\mathbf{s}_m[t+1], \text{argmax}_{\boldsymbol{a}_m[t]\in\mathcal{A}_m} Q(\mathbf{s}_m[t+1], \boldsymbol{a}_m[t]; \theta_m, \theta_m^V, \theta_m^A); \hat{\theta}_m, \hat{\theta}_m^V, \hat{\theta}_m^A)$ is known as the TD-target (or the temporal distance of Q). Clearly, it is necessary to optimize θ_m. To do this, we minimize the distance between $Q(\mathbf{s}_m[t], \boldsymbol{a}_m[t]; \theta_m, \theta_m^V, \theta_m^A)$ and the TD-target y_m, which is expressed as the loss function $\mathcal{L}(\theta_m)$:

$$\mathcal{L}(\theta_m) = \mathbb{E}\big(y_m[t] - Q(\mathbf{s}_m[t], \boldsymbol{a}_m[t]; \theta_m, \theta_m^V, \theta_m^A)\big)^2. \tag{6.28}$$

Meanwhile, the target network is used to stabilize the learning process by updating its parameters $\hat{\theta}_m$ by copying the parameters θ_m from the online network after a specified number of frames, a process known as the parameter update frequency C, as illustrated in

Table 6.2: Simulation Parameters

Parameter	**Value**	**Parameter**	**Value**
No. RUs	3	Pre. uRLLC latency (D^{ur})	0.5 ms
No. eMBB users	14	Pre. eMBB data rate (R^{min})	10 Mbps
No. uRLLC users	6	Noise power (N_0)	-137 dBm
BW of RU	5 MHz	uRLLC SNR threshold (Γ^{th})	17 dB
No. RBs per sub-frame	52	Pre. RU's queue-length (q_m^{max})	10 KB
BW split variable	0.3	Discount factor (γ)	0.99
No. of antennas (K_{tx})	8	Learning rate (η)	0.0001
Packet size (Z^{ur})	32 B	Buffer capacity (C^{max})	1e+05
Packet size (Z^{em})	64 KB	Batch size	64
Time-frame (Δ)	10 ms	r^{neg}	-2

Fig. 6.4.3. The D3QN-based algorithm is summarized in Algorithm 9.

6.5 Performance Evaluations

6.5.1 Simulation Environment and Benchmark Schemes

We consider a network topology with three RUs and UEs randomly distributed according to a Poisson process within a 500 m radius. The quasi-static channel vector between RU m and UE u for all TTIs within each frame is modeled as $\boldsymbol{h}_{m,u}^{f_i} = \sqrt{10^{-\mathsf{PL}_{m,u}/10}}\bar{\boldsymbol{h}}_{m,u}^{f_i}$, where $\mathsf{PL}_{m,u}$ is the log-distance path-loss. The traffic flows for eMBB and uRLLC follow the FTP3 model, with mean arrival rates of 21.12 and 2.12 packets/frame, respectively. Assuming $P_e = 10^{-3}$, MCS12 is chosen with $\Gamma^{\mathsf{th}} \approx 17$ dB, $\kappa = 3.90$ bits/symbol, and $\rho = 60$ symbols/RB, accounting for reference signal overhead. The proposed Algorithm 6 runs for $T = 100$ sub-frames based on the 5G NR frame structure. The DRL method is implemented using TensorFlow 2.12.0 with Keras, and simulations are performed on a Dell desktop with an Intel Core i7-10610U CPU, 16 GB RAM, and Windows 11. The rest of the simulation parameters are given in Table 6.2.

Benchmark Schemes: To evaluate the effectiveness of the proposed methods, we compare them with four benchmark schemes:

1. *Centralized*: In this scheme, RB scheduling and power allocation are centrally optimized at a server within the RICs of the Open RAN architecture, with full knowledge of RAN components, channel conditions, and interference. The problem is solved using the SCA method.

2. *Uniform Power*: RB scheduling is determined by solving an optimization problem via the SCA method, assuming equal power allocation for each RU as $p_{m,u}^{f_i,t_i} \triangleq P_m^{\mathsf{max}}/\sum_i F_i$.

3. *Slice Isolation*: This scheme explores slice isolation within the *Centralized* scheme to demonstrate the effectiveness of slice awareness, particularly in enhancing eMBB data rate.

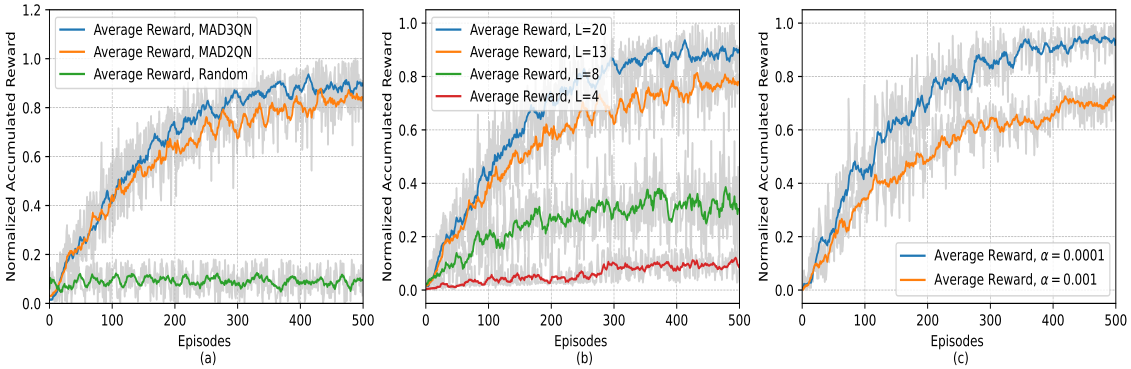


FIGURE 6.4: The convergence analysis of Algorithm 9: (a) with different approaches, (b) with different transmission power levels, and (c) with different learning rates.

4. *OFDMA*: Similar to *Centralized* scheme, this method optimizes RB scheduling and power allocation but ignores interference effects [149].

These benchmark schemes are used to thoroughly evaluate the proposed methods for solving the optimization problem 6.12.

6.5.2 Numerical Results

Fig. 6.4(a) shows the convergence behavior during the training phase for the proposed MAD3QN, multi-agent deep double Q-network (MAD2QN), and random schemes with normalized accumulated rewards over 500 episodes. The Random scheme consistently achieves the lowest rewards, highlighting its inefficiency in optimizing network performance due to random power level selection. Among the structured approaches, MAD3QN outperforms both MAD2QN and Random, with a steady increase in rewards peaking around episode 400 and maintaining a higher average after that. This indicates MAD3QN's superior ability to navigate the complex state-action space and dynamic conditions. While MAD2QN performs better than Random, it still falls short of MAD3QN, underscoring the benefits of the D3QN approach in providing more accurate value estimations and policy evaluations. Overall, these results validate the effectiveness of MAD3QN in achieving higher performance and more stable learning.

Fig. 6.4(b) compares the normalized accumulated rewards of MAD3QN across different transmission power levels, L. The results show that the higher the power levels, the better the performance can be achieved. The reason is attributed to the fact that they enable more nuanced and optimal decisions to enhance network performance. The proposed method with $L = 20$ consistently achieves the highest rewards, highlighting the advantage of a larger state-action space in maximizing the capabilities of DRL algorithms like MAD3QN. In contrast, $L = 4$ yields the lowest rewards due to the limitation of a smaller set of power levels in effectively optimizing network performance in a dynamic environment.

Fig. 6.4(c) shows the normalized accumulated rewards of the MAD3QN algorithm with two learning rates, η, over 500 episodes. The lower learning rate (*e.g.*, $\eta = 0.0001$) consistently outperforms the higher rate (*e.g.*, $\eta = 0.001$) and achieves higher and more

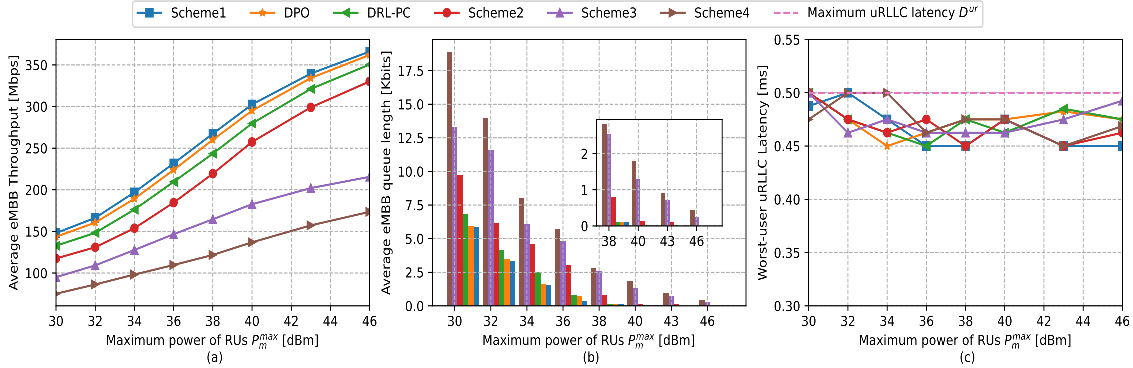


FIGURE 6.5: The performance comparison between Algorithm 6 and existing benchmark schemes to solve ICUR Problem (6.12) versus the maximum power budget of RU $P_m^{\max}$ in terms of (a) average overall eMBB data rate, (b) average backlog eMBB queue length, and (c) worst uRLLC latency.

stable rewards. Both learning rates initially show rapid learning, but the lower rate continues to improve, reaching a higher peak and maintaining stability in later episodes. This suggests that the lower learning rate enables more precise fine-tuning, leading to better long-term performance. In contrast, the higher learning rate, while initially competitive, plateaus at a lower reward level with greater fluctuations. This phenomenon indicates that the higher learning rate may cause the algorithm to overshoot optimal solutions and struggle to converge to the best possible policy.

We compare the performance of Algorithm 6 with the benchmark schemes indicated in Fig. 6.5, employing different techniques. The average eMBB data rate over the maximum power budget for RUs $P_m^{\max}$, which spans from 30 to 46 dBm, is shown in Fig. 6.5(a) to analyze the eMBB data rate under various resource allocation schemes. As anticipated, the *Centralized* scheme performs the best out of all the schemes and provides a benchmark for comparison. This is explained by the fact that the *Centralized* scheme can jointly optimize RB scheduling and power allocation via the SCA method since, being a centralized technique, it has access to perfect CSI, interference channel gains, queue length, and comprehensive RAN information. Conversely, the proposed methods (*i.e.*, DIPO and DRL-PC), being decentralized methods, depend on the local data of each RU and do not have access to comprehensive information, such as interference from neighboring RUs and queue length. However, the performance gap is smaller than 1.1% and 4.8% at $P_m^{\max} = 43$ dBm, demonstrating the effectiveness of the proprietary DIPO and DRL-PC, respectively, in precisely optimizing power allocation over time in such a dynamic environment. Furthermore, in comparison with the other schemes, the proposed schemes achieve a better eMBB data rate. Specifically, at the usual power value of $P_m^{\max} = 38$ dBm, DRL-PC provides a performance boost of 11%, 45%, and 92% over *Uniform Power*, *Slice Isolation*, and *OFDMA* schemes, respectively. Notably, the DRL-PC method performs similarly to the DIPO method, highlighting the effectiveness of the MAD3QN model. As the power budget increases, all methods for optimizing RB and power allocation improve. However, the *OFDMA* scheme performs the worst compared to the other schemes, emphasizing the importance of RUs sharing RBs and managing interference power effectively.

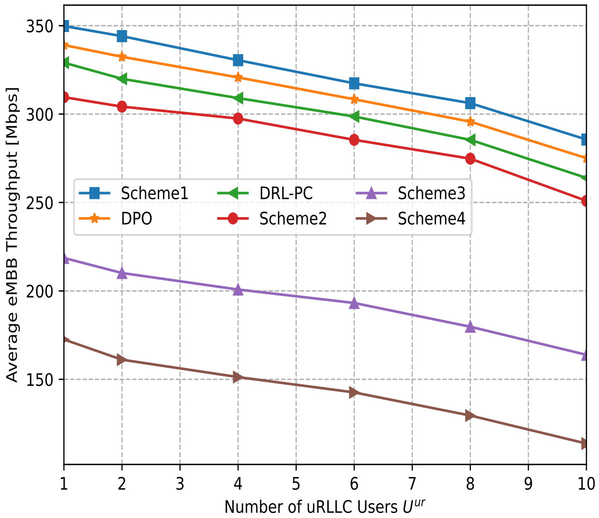


FIGURE 6.6: Effect of number of uRLLC UEs on the eMBB performance with different schemes.

Fig. 6.5(b) shows the average queue length at different maximum power budgets for RUs. As previously noted, higher data rates in eMBB services correspond to shorter queue lengths. Since Fig. 6.5(a) indicates that higher P_m^{max} results in higher eMBB data rate, it is expected that higher P_m^{max} would also lead to shorter eMBB queue lengths. This is confirmed in Fig. 6.5(b), where the average queue length decreases as P_m^{max} increases. Interestingly, there is a small performance gap between the *Centralized* scheme, the upper-bound scheme, and the proposed methods (DIPO and DRL-PC). Both proposed methods effectively manage queue lengths, adjusting to different power budgets while maintaining a low average backlog. For $P_m^{\mathsf{max}} \geq 38$ dBm, RUs consistently service more packets than arrive, preventing buffer backlogs. In contrast, the *OFDMA* scheme performs the worst, while the *Uniform Power* and *Slice Isolation* schemes show significant improvement over the *OFDMA* one.

In Fig. 6.5(c), the worst-user uRLLC latency is analyzed, where every scheme meets the necessary uRLLC latency requirement of 0.5 ms. It should be noted that the RB scheduling phase is responsible for maintaining uRLLC latency within the uRLLC requirement constraint immediately after arriving in the buffer, whereas the calculated uRLLC latency is not significantly affected by changing the power budget, according to the formulated latency for uRLLC service in (6.9).

Fig. 6.6 depicts the variation in average eMBB data rate as the number of uRLLC UEs (U^{ur}) increases across different methods. The figure demonstrates that as U^{ur} rises, eMBB performance declines due to increased interference from more uRLLC UEs sharing the same RBs. Additionally, the number of unused RBs in the uRLLC slice available for eMBB allocation decreases as U^{ur} grows. Despite this, *Centralized*, DIPO, and DRL-PC consistently deliver better eMBB performance, with small performance gaps between them, compared to *Uniform Power*, *Slice Isolation*, and *OFDMA* schemes. As seen in previous results, the *Uniform Power* scheme outperforms *Slice Isolation* and *OFDMA*.

We now evaluate the impact of Algorithm 7 in managing buffers under congestion with

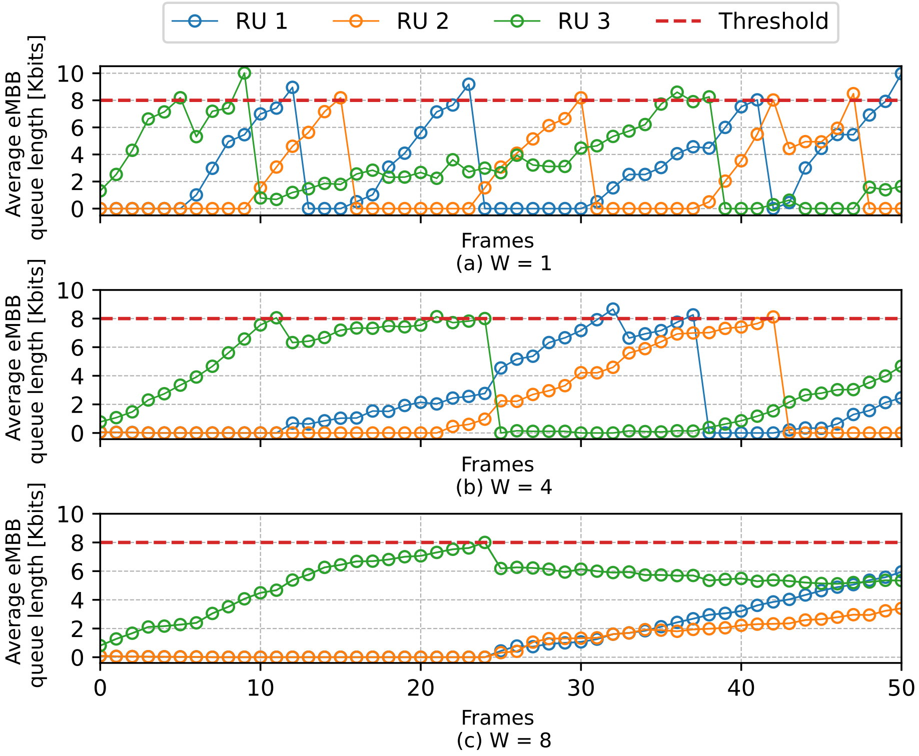


Figure 6.7: Impact of the UA Algorithm 7 on the average eMBB queue length with different window sizes W for $P_m^{\mathsf{max}} = 38$ [dBm] and $q^{\mathsf{th}} = 0.8q_m^{\mathsf{max}}$.

different window sizes W. Fig. 6.7 shows the average eMBB queue length per RU over 50 frames. RU 3 is initially overloaded by serving more UEs than other RUs. The red dashed line marks the queue threshold at 80% of the maximum buffer capacity, triggering a congestion alarm if reached. When this happens, Algorithm 7 activates to prevent network outages and packet loss by adjusting UA variables and initiating handovers. The figure also illustrates how Algorithm 7 identifies congested RUs and reassigns UEs to other RUs, sharply reducing the congested RU's buffer below the threshold. However, this causes queue lengths at other RUs to increase, potentially triggering further congestion alarms. To avoid excessive handovers, the UA parameter is based on average dynamics rather than instantaneous queues, making window size crucial for balancing accuracy and responsiveness. Fig. 6.7(c) shows that a larger window size offers a more stable estimate but reacts slowly to changes while a smaller window size, as seen in Figure 6.7(a), responds quickly but is more susceptible to noise, leading to more handovers. Fig. 6.7(b) demonstrates that $W = 4$ achieves a balance, smoothing queue lengths compared to $W = 1$ while still responding adequately to network changes. Given the dynamic nature of network traffic, periodically reassessing and adjusting the window size is essential, a potential topic for future work.

Fig. 6.8 illustrates the impact of window size W on the average percentage of handovers during the simulation for the DIPO method. The graph reveals a clear inverse relationship between window size and handover frequency. A smaller window size leads to a higher

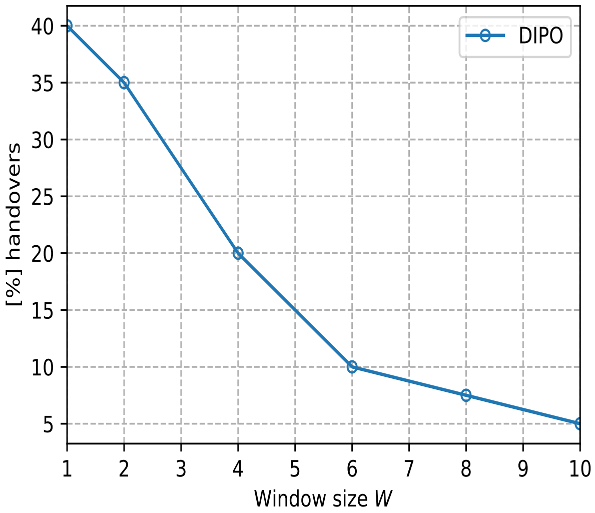


FIGURE 6.8: Effect of window size W on average percent of handovers within simulation runtime.

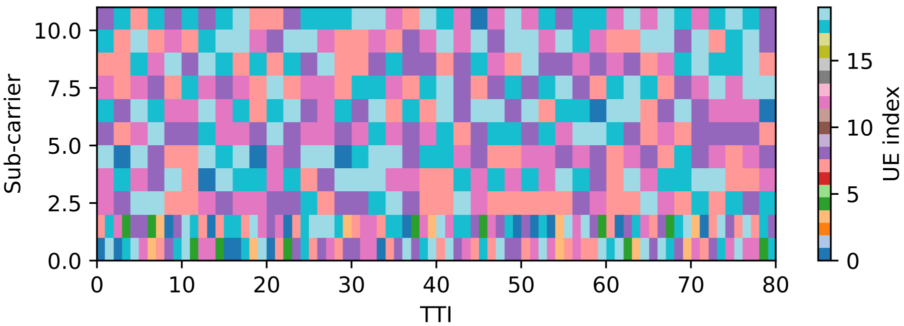


FIGURE 6.9: Visualization of RB scheduling for both services based on Algorithm 8 considering mixed numerology in the frequency domain.

percentage of handovers due to the algorithm's rapid response to instantaneous queue length changes. Conversely, a larger window size stabilizes the queue length estimates, reducing the number of handovers but at the cost of slower reactions to network changes.

Lastly, we provide a visualization of RB scheduling for both services based on Algorithm 8, as illustrated in Fig. 6.9. This also provides a detailed illustration of the slice-awareness method in uRLLC preemption and its impact on enhancing eMBB data rate for a single RU. We assume that uRLLC packets arrive randomly within the frame and must be served immediately upon arrival. If uRLLC UEs cannot transmit all packets within the frame, they request additional RBs from another slice considering mixed numerologies in the frequency domain [149]. Conversely, eMBB UEs can utilize the unused RBs from the uRLLC slice, significantly boosting data rate. The figure also shows that UEs with lower indexes represent uRLLC UEs, which occupy fewer RBs compared to the higher-indexed eMBB UEs. This method not only improves eMBB data rate but also ensures that uRLLC users can access additional resources as needed. As can be seen, Algorithm 8 effectively meets the stringent requirements of both uRLLC and eMBB services by dynamically adjusting RB allocation according to varying demands, optimizing resource utilization in the process.

6.6 Summary

We have proposed an intelligent UA, resource scheduling, and power control scheme (IUCR) designed to minimize the overall queue lengths and uRLLC latency while avoiding frequent handovers within the Open RAN architecture, leveraging the centralized intelligence of RICs. The IUCR scheme is specifically designed to navigate the dynamic and complex nature of modern networks, accommodating multiple traffic types like eMBB and uRLLC. The optimization framework is tailored for the 7.2x FS within the Open RAN architecture, incorporating advanced techniques such as RAN slicing, mixed numerology multiplexing, and a mini-slot-based frame structure. These innovations ensure efficient resource utilization and adherence to SLAs by reducing eMBB queue lengths and minimizing uRLLC latency. Additionally, we have proposed a hierarchical optimization strategy that integrates heuristic methods, iterative SCA, and distributed MAD3QN-based algorithms. This multi-faceted approach addresses challenges like incomplete queue length information, frequent handovers, channel interference, and partial environmental observations. By leveraging closed-control loops between RAN components and RICs, our scheme ensures effective resource management even in large-scale networks with limited data. Comprehensive simulations validate the effectiveness of the IUCR scheme, especially in congested scenarios. Future research will aim to further enhance the framework's adaptability and efficiency by integrating advanced ML techniques such as federated learning, improving decision-making processes in distributed environments, and bolstering the system's ability to adapt to evolving network conditions.

Chapter 7

Conclusions and Future Research

In Section 7.1, we provide a summary of the work conducted in this thesis, highlighting key conclusions. Furthermore, Section 7.2 discusses potential directions for future research.

7.1 Main Conclusions

Open radio access network (RAN) represents a groundbreaking shift in wireless telecommunications, aiming to transform traditional proprietary RANs into open, intelligent, and interoperable systems. Recently, machine learning (ML) applications have gained significant attention in Open RAN for their potential to revolutionize network automation and enable data-driven decision-making, such as intelligent traffic steering (TS). This streamlines network management and allows mobile network operators (MNOs) to optimize performance. ML algorithms are crucial for TS research, offering real-time intelligence and decision-making to enhance traffic flow and overall network efficiency. However, challenges remain, including handling dynamic environments and unpredictable traffic demands and patterns, frequent handovers, load balancing, and resource constraints. This dissertation addresses these issues by focusing on intelligent TS and user association (UA) within the Open RAN architecture. The core structure of this thesis is summarized as follows:

- Chapter 1 provided a comprehensive introduction and literature review, structured into three main sections: related works; motivation and limitations of existing works; and the thesis outline and contributions. The first section investigated the literature review in three subsections. The first subsection explored radio resource management (RRM) for multi-traffic scenarios, detailing the strategies and challenges in balancing these diverse service requirements. The second subsection reviewed UA and TS approaches in traditional RAN, highlighting established methods and their limitations. The final subsection shifted focus to intelligent TS and UA within Open RAN, discussing recent advancements and the potential for machine learning to enhance network performance and management. After investigating the limitations of the mentioned studies in the second section that motivated this thesis to address them, the last section highlighted its contributions and achievements. This chapter established a foundation for understanding the evolution and current state of traffic management and resource optimization, setting the stage for further exploration and innovation in subsequent chapters.

- In Chapter 2, we conducted a thorough examination of the foundational elements pertinent to the dissertation. We began by discussing the traditional RAN architecture, its limitations, and the evolution towards new network paradigms. Next, we detailed the Open RAN architecture, describing its key components, principles, interfaces, functional split (FS) options, and how it diverges from traditional RAN systems. Finally, we explored the role of ML within the Open RAN framework, emphasizing how ML techniques can improve network efficiency, automation, and decision-making. This chapter set the stage by providing the context and technical background essential for understanding the advancements and implications of integrating ML applications into Open RAN.

- In Chapter 3, we developed a traffic steering framework within the Open RAN architecture that optimizes centrally resource allocation for coexistence of enhanced mobile broadband (eMBB) and ultra-reliable low-latency communication (uRLLC) services. This framework utilizes macro and small cells as types of radio units (RUs) and leverages isolation numerology scalability techniques to maximize eMBB throughput while minimizing uRLLC latency under known traffic demands. To address the optimization problem, we introduced an efficient iterative algorithm based on the successive convex approximation (SCA) method, ensuring locally optimal solutions. Numerical results confirmed the rapid convergence of the algorithm and its effectiveness in enhancing eMBB throughput and reducing uRLLC latency, outperforming traditional resource allocation methods.

- In Chapter 4, we presented a comprehensive framework for intelligent traffic prediction, dynamic RAN slicing, flow-split distribution, and RRM within the orthogonal frequency division multiple access (OFDMA)-based Open RAN architecture, addressing the challenge of unknown dynamic traffic demands. To maximize eMBB throughput while ensuring minimal uRLLC latency and vice versa, we formulated two optimization problems tailored to the specific service requirements, subject to constraints such as quality of service (QoS), slice isolation, power budget, and fronthaul (FH) capacity. These problems were decomposed into long-term and short-term subproblems based on different time scales, with the long-term subproblems managed by the non-real-time (non-RT) RAN intelligent controller (RIC) through traffic prediction, RAN slicing, and flow-split decisions, while the short-term subproblems were addressed using the SCA method at the near-real-time (near-RT) RIC. Numerical results demonstrated the fast convergence of the proposed algorithm and its superior performance compared to benchmark schemes, including fixed numerology, equal flow-split distribution, single connectivity, and the proposed framework with known traffic demands.

- In Chapter 5, we introduced a sophisticated multi-layer intelligent TS framework designed for slice-aware operations within the OFDMA-based Open RAN architecture.

This framework centrally tackles challenges such as high complexity, varying channel conditions, and dynamic traffic demands. By employing dynamic multi-connectivity (MC) management, slice-aware RAN slicing, and mixed numerologies multiplexing across both frequency and time domains, we aimed to minimize the long-term average queue length for eMBB UEs and reduce the long-term average latency for uRLLC, all while adhering to QoS requirements, slice awareness, power budget, and flow-split constraints. To address the lack of complete information—such as time-varying channel state information (CSI) and queue lengths—we introduced a centralized ML approach that reduces computational complexity by making decisions per frame rather than at each time slot. This framework implements a two-stage optimization strategy: long-term decisions are managed by customized xAPPs in the near-RT RIC, and short-term decisions are handled by distributed units (DUs). Extensive simulations demonstrate the effectiveness of our method compared to benchmark schemes like slice isolation and SCA, further validating its superiority in dynamic environments.

- In Chapter 6, we tackled the critical challenges of frequent handovers and load balancing within the Open RAN architecture by introducing an innovative joint intelligent UA, congestion control, and resource scheduling scheme aligned with the 7.2x FS. Building on the non-orthogonality between RUs and guided by O-RAN Alliance recommendations, our approach seeks to minimize eMBB queue lengths and uRLLC latency while addressing congestion control, power budgets, and other real-world constraints. We employed a hierarchical optimization strategy, combining heuristic algorithms for UA, iterative SCA for resource scheduling, and distributed deep reinforcement learning (DRL) for power optimization. These algorithms were strategically deployed across different Open RAN nodes, with the near-RT RIC handling UA updates via a heuristic algorithm, the central unit (CU) optimizing physical resources through the SCA method, and RUs managing power allocation distributively via DRL. The chapter concludes with extensive numerical results showcasing the superior performance of our proposed scheme in comparison to centralized OFDMA and other existing methods, highlighting its effectiveness in dynamic environments.

7.2 Future Works

Within this dissertation, we addressed several key challenges within the Open RAN architecture, focusing on intelligent TS, UA, RRM, congestion control, and load balancing for optimization of the coexistence of eMBB and uRLLC services. Through a series of novel frameworks, algorithms, and simulation-based validation, we demonstrated how advanced techniques, such as centralized and decentralized ML algorithms, time-series algorithms, optimization methods such as SCA, and heuristic methods, can effectively manage dynamic known/unknown traffic demands, optimize resource allocation, and improve overall network performance. Our work considered both centralized and decentralized strategies, with particular attention to Open RAN's 7.2x FS, ensuring that the solutions were adaptable to real-world constraints like power budgets, fronthaul capacity, and service-specific QoS requirements. However, despite the advancements presented in this thesis, there remain numerous opportunities for future research. Addressing these areas could further enhance the performance, scalability, and robustness of Open RAN systems. Some potential avenues for future research include:

- **Scalability and Adaptability in Large-Scale Networks**: The Open RAN, with its disaggregated and software-centric design, offers unique opportunities to address accommodating billions of devices but also introduces new complexities when deployed at large scales. Future research must focus on enhancing the scalability and adaptability of Open RAN systems to support the seamless growth and evolution of next-generation networks. One of the key challenges in scaling Open RAN is the efficient management of resources across a vast number of DUs, CUs, and RUs. As network size increases, so does the complexity of orchestrating these components, especially when managing heterogeneous traffic demands across different regions. Future work could explore the development of advanced orchestration platforms that leverage AI and ML to dynamically allocate resources, balance loads, and optimize performance in real-time, even as network size and traffic volumes grow exponentially. These platforms could enable Open RAN to automatically adjust to changing network conditions, ensuring that performance remains consistent as the network scales. Another important aspect of scalability in Open RAN is the ability to efficiently handle a massive number of devices and connections. As networks expand, they must support not only a growing number of user devices but also an increasing variety of IoT devices, each with different connectivity requirements. By dynamically creating and adjusting slices in response to network demands, Open RAN can maintain high levels of performance and QoS even as the network scales. Adaptability is equally critical in large-scale Open RAN deployments. Networks must be able to adapt to a wide range of scenarios, from sudden surges in traffic due to large events to the rapid deployment of new services and technologies. To overcome these, the real-time analytics and predictive algorithms could enable the network to anticipate and respond to changes before they impact performance, such as by reallocating resources, adjusting power levels, or re-configuring network slices. This proactive approach would allow Open RAN to maintain optimal performance even in the face of unpredictable and rapidly changing conditions. Additionally, in large-scale networks, it is essential to ensure that resources are distributed efficiently to avoid bottlenecks and ensure consistent performance across the entire network. Hence, future research could investigate the development of decentralized resource management algorithms that distribute decision-making closer to the network edge, allowing for more responsive and localized control. This approach could enhance the scalability of Open RAN by reducing the burden on central control units and enabling more efficient use of network resources.

- **Security and Privacy Enhancements for Open RAN**: The open and disaggregated nature of Open RAN, while offering significant advantages in terms of innovation and interoperability, also introduces new vulnerabilities and attack surfaces that could be exploited by malicious actors. Therefore, one of the most promising directions for future research involves developing advanced security and privacy frameworks tailored specifically to the Open RAN environment. In traditional RAN architectures, security mechanisms are often tightly integrated within proprietary systems, making them less susceptible to certain types of attacks. However, the shift towards an open and multi-vendor ecosystem in Open RAN necessitates a rethinking of security strategies. Future research could focus on designing security protocols that are adaptable to the diverse range of components within Open RAN, including the CU, DU, and RU, as well as the RICs. These protocols must ensure secure communication, data integrity, and authentication across all layers of the network, from the physical layer up to the application layer. Another key area of focus could be the development of privacy-preserving algorithms for managing sensitive user data within Open RAN. As the network handles an increasing amount of personal and service-related data, it is essential to implement techniques that safeguard user privacy while maintaining network performance. This could involve the use of advanced encryption methods, secure multi-party computation, and federated learning approaches that allow for distributed data processing without exposing individual data points. Additionally, the integration of ML into Open RAN, as explored in this thesis, presents both opportunities and challenges in terms of security. On one hand, AI and ML can enhance threat detection and response capabilities by analyzing network patterns and identifying anomalies in real-time. On the other hand, these technologies can also be targeted by adversarial attacks, where malicious inputs are crafted to deceive the learning models. Future work could explore the development of robust and resilient AI/ML models that can withstand such attacks, ensuring the integrity of the decision-making processes within the RAN. The implementation of a zero trust architecture (ZTA) within Open RAN could also be a pivotal step towards enhancing security. In a ZTA model, no entity, whether inside or outside the network, is trusted by default. Every access request is thoroughly authenticated, authorized, and encrypted, thereby minimizing the risk of breaches. Research could explore how ZTA principles can be applied to the various components of Open RAN, creating a secure environment that is resilient to both internal and external threats. Lastly, the collaborative and multi-vendor nature of Open RAN requires a coordinated approach to security management. Future research could investigate the establishment of standardized security frameworks and protocols that can be universally adopted across different vendors and operators, ensuring a cohesive security posture throughout the network. This would not only protect the network from emerging threats but also foster trust and collaboration within the Open RAN ecosystem. By addressing these challenges through innovative research and development, we can build a secure, resilient, and trustworthy Open RAN architecture that meets the demands of future wireless networks.

- **Integration of Non-Terrestrial Networks in Open RAN Architecture**: Another exciting avenue for future research involves the integration of non-terrestrial networks (NTN), particularly unmanned aerial vehicles (UAVs), into the Open RAN architecture. As the demand for ubiquitous connectivity and network flexibility continues to grow, NTNs offer a promising solution by extending network coverage and enhancing capacity, especially in hard-to-reach or underserved areas. UAVs, functioning as aerial base stations (BSs) or relay nodes, can dynamically adjust their position to optimize signal strength and network coverage, making them ideal for complementing terrestrial RAN deployments. Integrating UAVs into the Open RAN framework presents unique opportunities to leverage the existing flexibility and modularity of Open RAN components. This includes exploring new functional splits that allow for efficient coordination between ground-based and airborne network elements, optimizing the use of radio resources across different layers of the network, and ensuring seamless connectivity in both urban and remote environments. However, the integration of UAVs and other NTNs into Open RAN also introduces challenges that require innovative solutions. These include managing the high mobility of UAVs, dealing with the varying quality of air-to-ground and air-to-air communication links, and addressing the stringent power constraints of UAVs. Future research could focus on developing specialized algorithms for policy-based traffic management, TS, UA, and RRM that take into account the unique characteristics of NTNs. For instance, UAVs could be used to dynamically offload traffic from congested ground networks, provide temporary coverage during network outages, or enhance capacity during high-demand events. Additionally, incorporating UAVs into the Open RAN ecosystem necessitates advancements in network orchestration and management. The near-RT and non-RT RICs could be extended to include capabilities for managing the interaction between terrestrial and non-terrestrial components, ensuring that network policies are efficiently applied across different domains. The use of advanced ML algorithms could further enhance decision-making processes, allowing the network to autonomously adapt to changing conditions and optimize performance in real-time. This approach not only promises to improve connectivity in challenging environments but also opens up new possibilities for innovative services and applications ranging from disaster recovery to smart agriculture and beyond.

- **Online Learning for Adaptive TS in Open RAN**: Future research could explore the integration of online learning algorithms to enable adaptive and real-time TS in Open RAN. Possible solutions include employing multi-armed bandit algorithms to dynamically balance exploration and exploitation, ensuring optimal resource allocation even under uncertain and changing network conditions. Another approach could involve online gradient descent methods, allowing for continuous updates to TS models as new data is received. Additionally, reinforcement learning with online updates could be used to adjust traffic flow and UA in real-time, improving the network's ability to respond to fluctuating traffic demands, user mobility, and varying channel quality. These online learning solutions would enhance the responsiveness and accuracy of TS mechanisms, particularly in highly dynamic and unpredictable network environments.

- **Validation and Refinement of Intelligent TS Using the OpenRAN Gym Testbed**: A critical next step in advancing our research involves the rigorous testing and validation of the proposed methods within the OpenRAN Gym testbed. This platform serves as a sophisticated and highly realistic simulation environment tailored to Open RAN architecture, enabling the emulation of complex, real-world network conditions and scenarios. By integrating our intelligent TS, UA, and RRM strategies within this testbed, we can conduct comprehensive evaluations of their performance under varying traffic loads, service requirements, and environmental dynamics. The OpenRAN Gym testbed offers the ability to test across a range of configurations, such as different functional split options, varied fronthaul capacities, and heterogeneous service demands. This flexibility will allow us to critically assess the robustness of our algorithms, particularly in scenarios where service-specific QoS requirements, such as those for eMBB and uRLLC, must be maintained amidst fluctuating network conditions. Furthermore, it provides a platform to evaluate the impact of our proposed optimizations on key performance indicators (KPIs) like latency, throughput, and reliability in a controlled yet realistic setting. Through iterative testing within this environment, we can refine our models and algorithms, addressing any limitations identified during simulation. This step is crucial in bridging the gap between theoretical research and practical application, ensuring that the proposed solutions are not only innovative but also implementable in the context of future wireless networks.

- **Ultra-Dense Networks and Small Cell Optimization in Open RAN**: As the demand for higher data rates, increased capacity, and seamless connectivity continues to grow, ultra-dense networks (UDNs) have emerged as a critical component of future wireless networks. UDNs involve the deployment of a large number of small cells in a given area, significantly increasing the network's density and, consequently, its ability to handle massive amounts of traffic. However, the integration of UDNs into Open RAN requires careful consideration of several factors, including interference management, resource allocation, load balancing, and mobility management. The dense deployment of small cells can lead to increased interference, particularly in environments where multiple cells overlap. While small cells in UDNs are typically characterized by limited coverage areas and the need for efficient backhaul connectivity, future research should focus on developing advanced interference mitigation techniques that are tailored to the Open RAN framework. This could involve the design of intelligent, context-aware algorithms that dynamically adjust transmission power, frequency allocation, and beamforming strategies to minimize interference while maximizing throughput. Besides, load balancing in UDNs is another critical area for future research. The high density of small cells means that users may frequently move between different cells, leading to challenges in maintaining consistent QoS. Open RAN's disaggregated architecture, with its near-RT and non-RT RICs, offers a promising platform for implementing dynamic load balancing algorithms. These algorithms could leverage real-time data on user location, traffic patterns, and network conditions to make intelligent decisions about handovers, resource allocation, and traffic steering. The goal would be to ensure that the load is evenly distributed across the network, minimizing congestion and optimizing the user experience. By addressing challenges such as interference management, load balancing, and mobility management and by leveraging the flexibility and intelligence provided by Open RAN's disaggregated architecture, it is possible to realize the full potential of UDNs.

- **End-to-End Latency Optimization for Critical Applications**: In the evolving landscape of wireless networks, the need for ultra-low latency communication has become paramount, particularly for critical applications such as autonomous vehicles, industrial automation, remote surgery, and immersive real-time experiences. End-to-end latency in an Open RAN environment is influenced by multiple factors, including the protocols and policies implemented at different layers of the network, the impact of information transfer latency across open interfaces, and the processing delays made by network functions distributed between the CU, DU, and RU. These components, connected via open interfaces such as F1 and E1, must operate with minimal delay to ensure that time-sensitive data is processed and transmitted swiftly. Researchers could explore the optimization of these protocols, possibly through the development of streamlined signaling procedures or the introduction of more efficient compression techniques that reduce the amount of data exchanged without compromising the integrity of the information. By addressing the various sources of latency, from protocol inefficiencies to open interface delays, and by integrating these optimizations into a cohesive, cross-layer framework, it is possible to meet the stringent QoS requirements of next-generation applications. This work will be crucial in ensuring that Open RAN can support the demanding needs of future wireless networks, delivering the ultra-reliable and low-latency performance required for the most critical use cases.

- **Cross-Layer Optimization and QoS-Aware Network Slicing**: As 5G networks evolve towards 6G and beyond, one promising direction for future research is the development of cross-layer optimization strategies within the Open RAN architecture, aimed at enhancing network slicing capabilities to meet the diverse QoS demands. Network slicing (NS) allows operators to create multiple virtual networks over a common physical infrastructure, each tailored to specific use cases such as eMBB, uRLLC, and mMTC. However, the traditional approach to network slicing, which primarily operates at a single layer of the network stack, often falls short in efficiently managing the dynamic and heterogeneous nature of modern wireless services. Future research could focus on cross-layer optimization techniques that harmonize resource allocation across different layers of the Open RAN architecture, from the physical and MAC layers to the network and application layers. This approach would enable a more holistic view of the network, where decisions made at one layer are informed by the conditions and requirements of other layers. For instance, the integration of RRM with higher-layer functions like TS and UA could lead to more intelligent and adaptive network behavior, improving the overall efficiency and performance of the system. Moreover, QoS-aware NS requires sophisticated algorithms that can dynamically adjust to varying traffic loads, user mobility, and service-level agreements (SLAs). In this context, the application of ML techniques, as explored in this thesis, could play a pivotal role. By leveraging real-time data and predictive analytics, ML algorithms can enable the network to anticipate changes in traffic patterns and adapt resource allocation accordingly, ensuring that each network slice maintains the required QoS levels even under fluctuating network conditions. Future research could investigate how to optimize the interactions between RAN nodes and RICs to support end-to-end QoS across the entire network. This could involve the development of advanced coordination mechanisms between non-RT and near-RT RICs, ensuring that the network can respond swiftly to real-time demands while maintaining overall stability and performance. By integrating insights from multiple layers and leveraging advanced

ML techniques, these innovations could pave the way for next-generation wireless networks capable of meeting the diverse and dynamic demands of future services and applications.

In summary, the future of Open RAN research offers a wealth of possibilities for further enhancing network intelligence, adaptability, and performance. By exploring these potential avenues, future work can contribute to the evolution of Open RAN as a cornerstone technology in the next generation of wireless networks.

Bibliography

[1] A. Damnjanovic, J. Montojo, Y. Wei, T. Ji, T. Luo, M. Vajapeyam, T. Yoo, O. Song, and D. Malladi, "A survey on 3GPP heterogeneous networks," *IEEE Wireless communications*, vol. 18, no. 3, pp. 10–21, 2011.

[2] M. Series, "IMT Vision–Framework and overall objectives of the future development of IMT for 2020 and beyond," *Recommendation ITU*, vol. 2083, no. 0, pp. 1–21, 2015.

[3] B. S. Khan, S. Jangsher, A. Ahmed, and A. Al-Dweik, "uRLLC and eMBB in 5G industrial IoT: A survey," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1134–1163, 2022.

[4] J. Navarro-Ortiz, P. Romero-Diaz, S. Sendra, P. Ameigeiras, J. J. Ramos-Munoz, and J. M. Lopez-Soler, "A survey on 5G usage scenarios and traffic models," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 905–929, 2020.

[5] A. Osseiran, J. F. Monserrat, and P. Marsch, *5G mobile and wireless communications technology.* Cambridge University Press, 2016.

[6] A. Checko, H. L. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. S. Berger, and L. Dittmann, "Cloud RAN for mobile networks—A technology overview," *IEEE Communications surveys & tutorials*, vol. 17, no. 1, pp. 405–426, 2014.

[7] M. Polese, L. Bonati, S. D'oro, S. Basagni, and T. Melodia, "Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 1376–1411, 2023.

[8] D. Johnson, D. Maas, and J. Van Der Merwe, "NexRAN: Closed-loop RAN slicing in POWDER-A top-to-bottom open-source open-RAN use case," in *Proceedings of the 15th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & CHaracterization*, 2022, pp. 17–23.

[9] M. Dryjański, Ł. Kułacz, and A. Kliks, "Toward modular and flexible open RAN implementations in 6G networks: Traffic steering use case and O-RAN xapps," *Sensors*, vol. 21, no. 24, p. 8173, 2021.

[10] Y. Liu, B. Clerckx, and P. Popovski, "Network slicing for eMBB, uRLLC, and mMTC: An uplink rate-splitting multiple access approach," *IEEE Transactions on Wireless Communications*, 2023.

[11] P. Popovski *et al.*, "5G wireless network slicing for eMBB, uRLLC, and mMTC: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55 765–55 779, 2018.

[12] T. T. Nguyen, V. N. Ha, and L. B. Le, "Wireless scheduling for heterogeneous services with mixed numerology in 5G wireless networks," *IEEE Communications Letters*, vol. 24, no. 2, pp. 410–413, 2019.

[13] Y. L. Lee, T. C. Chuah, J. Loo, and F. Ke, "Proportional-fair uplink resource allocation with statistical QoS provisioning for RAN slicing," *Physical Communication*, vol. 65, p. 102389, 2024.

[14] A. Gharehgoli, A. Nouruzi, N. Mokari, P. Azmi, M. R. Javan, and E. A. Jorswieck, "AI-based resource allocation in end-to-end network slicing under demand and CSI uncertainties," *IEEE Transactions on Network and Service Management*, vol. 20, no. 3, pp. 3630–3651, 2023.

[15] A. K. Bairagi, M. S. Munir, M. Alsenwi, N. H. Tran, S. S. Alshamrani, M. Masud, Z. Han, and C. S. Hong, "Coexistence mechanism between eMBB and uRLLC in 5G wireless networks," *IEEE transactions on communications*, vol. 69, no. 3, pp. 1736–1749, 2020.

[16] R. Kassab, O. Simeone, and P. Popovski, "Coexistence of uRLLC and eMBB services in the C-RAN uplink: An information-theoretic study," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[17] X. Han, K. Xiao, R. Liu, X. Liu, G. C. Alexandropoulos, and S. Jin, "Dynamic resource allocation schemes for eMBB and uRLLC services in 5G wireless networks," *Intelligent and Converged Networks*, vol. 3, no. 2, pp. 145–160, 2022.

[18] A. K. Bairagi, M. S. Munir, M. Alsenwi, N. H. Tran, and C. S. Hong, "A matching based coexistence mechanism between eMBB and uRLLC in 5G wireless networks," in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, 2019, pp. 2377–2384.

[19] M. Alsenwi, N. H. Tran, M. Bennis, A. K. Bairagi, and C. S. Hong, "eMBB-uRLLC resource slicing: A risk-sensitive approach," *IEEE communications letters*, vol. 23, no. 4, pp. 740–743, 2019.

[20] Q. Chen, J. Wu, J. Wang, and H. Jiang, "Coexistence of uRLLC and eMBB Services in MIMO-NOMA Systems," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 1, pp. 839–851, 2022.

[21] E. Markoval, D. Moltchanov, R. Pirmagomedov, D. Ivanova, Y. Koucheryavy, and K. Samouylov, "Priority-based coexistence of eMBB and uRLLC traffic in industrial

5G NR deployments," in *2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*. IEEE, 2020, pp. 1–6.

[22] A. A. Esswie and K. I. Pedersen, "Opportunistic spatial preemptive scheduling for uRLLC and eMBB coexistence in multi-user 5G networks," *Ieee Access*, vol. 6, pp. 38 451–38 463, 2018.

[23] M. Setayesh, S. Bahrami, and V. W. Wong, "Resource slicing for eMBB and uRLLC services in radio access network using hierarchical deep learning," *IEEE Transactions on Wireless Communications*, vol. 21, no. 11, pp. 8950–8966, 2022.

[24] E. N. Tominaga, H. Alves, R. D. Souza, J. L. Rebelatto, and M. Latva-Aho, "Non-orthogonal multiple access and network slicing: Scalable coexistence of eMBB and uRLLC," in *2021 IEEE 93rd vehicular technology conference (VTC2021-Spring)*. IEEE, 2021, pp. 1–6.

[25] J. Tang, B. Shim, and T. Q. Quek, "Service multiplexing and revenue maximization in sliced C-RAN incorporated with uRLLC and multicast eMBB," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 4, pp. 881–895, 2019.

[26] R. M. Sohaib, O. Onireti, Y. Sambo, R. Swash, S. Ansari, and M. A. Imran, "Intelligent Resource Management for eMBB and uRLLC in 5G and beyond Wireless Networks," *IEEE Access*, vol. 11, pp. 65 205–65 221, 2023.

[27] M. Alsenwi, N. H. Tran, M. Bennis, S. R. Pandey, A. K. Bairagi, and C. S. Hong, "Intelligent resource slicing for eMBB and uRLLC coexistence in 5G and beyond: A deep reinforcement learning based approach," *IEEE Transactions on Wireless Communications*, vol. 20, no. 7, pp. 4585–4600, 2021.

[28] G. S. Kesava and N. B. Mehta, "Multi-connectivity for uRLLC and coexistence with eMBB in time-varying and frequency-selective fading channels," *IEEE Transactions on Wireless Communications*, vol. 22, no. 6, pp. 3599–3611, 2022.

[29] K. Zhang, X. Xu, J. Zhang, B. Zhang, X. Tao, and Y. Zhang, "Dynamic multiconnectivity based joint scheduling of eMBB and uRLLC in 5G networks," *IEEE Systems Journal*, vol. 15, no. 1, pp. 1333–1343, 2020.

[30] R. Liu, G. Yu, J. Yuan, and G. Y. Li, "Resource management for millimeter-wave ultra-reliable and low-latency communications," *IEEE Transactions on Communications*, vol. 69, no. 2, pp. 1094–1108, 2020.

[31] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. G. Andrews, "User association for load balancing in heterogeneous cellular networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2706–2716, 2013.

[32] S. F. Abedin, M. G. R. Alam, S. A. Kazmi, N. H. Tran, D. Niyato, and C. S. Hong, "Resource allocation for ultra-reliable and enhanced mobile broadband IoT applications in fog network," *IEEE Transactions on Communications*, vol. 67, no. 1, pp. 489–502, 2018.

[33] N. Namvar, W. Saad, B. Maham, and S. Valentin, "A context-aware matching game for user association in wireless small cell networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 439–443.

[34] K. Khawam, S. Lahoud, M. El Helou, S. Martin, and F. Gang, "Coordinated framework for spectrum allocation and user association in 5G HetNets with mmWave," *IEEE Transactions on Mobile Computing*, vol. 21, no. 4, pp. 1226–1243, 2020.

[35] M. Ali, Q. Rabbani, M. Naeem, S. Qaisar, and F. Qamar, "Joint user association, power allocation, and throughput maximization in 5G H-CRAN networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 10, pp. 9254–9262, 2017.

[36] A. Mesodiakaki, F. Adelantado, A. Antonopoulos, L. Alonso, and C. Verikoukis, "Energy and spectrum efficient user association in 5G heterogeneous networks," in *2016 IEEE 27th annual international symposium on personal, indoor, and mobile radio communications (PIMRC)*. IEEE, 2016, pp. 1–6.

[37] J.-S. Liu, C.-H. R. Lin, and Y.-C. Hu, "Joint resource allocation, user association, and power control for 5G LTE-based heterogeneous networks," *IEEE Access*, vol. 8, pp. 122 654–122 672, 2020.

[38] Z. Zhu, H. Li, Y. Chen, Z. Lu, and X. Wen, "Joint Optimization of Functional Split, Base Station Sleeping, and User Association in Crosshaul Based V-RAN," *IEEE Internet of Things Journal*, 2024.

[39] A. Prasad, F. S. Moya, M. Ericson, R. Fantini, and O. Bulakci, "Enabling RAN moderation and dynamic traffic steering in 5G," in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*. IEEE, 2016, pp. 1–6.

[40] S. Zhang, N. Zhang, S. Zhou, J. Gong, Z. Niu, and X. Shen, "Energy-sustainable traffic steering for 5G mobile networks," *IEEE Communications Magazine*, vol. 55, no. 11, pp. 54–60, 2017.

[41] D.-Y. Kim and S. Kim, "Network-Aided Intelligent Traffic Steering in 5G Mobile Networks," *Computers, Materials & Continua*, vol. 65, no. 1, 2020.

[42] X. Ba, L. Jin, Z. Li, J. Du, and S. Li, "Multiservice-based traffic scheduling for 5G access traffic steering, switching and splitting," *Sensors*, vol. 22, no. 9, p. 3285, 2022.

[43] H. Khaled, I. Ahmad, D. Habibi, and Q. V. Phung, "A green traffic steering solution for next generation communication networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 1, pp. 222–238, 2020.

[44] J. Burgueño, I. de-la Bandera, D. Palacios, and R. Barco, "Traffic Steering for eMBB in Multi-Connectivity Scenarios," *Electronics*, vol. 9, no. 12, p. 2063, 2020.

[45] H. Hojeij, G. I. Ricardo, M. Sharara, S. Hoteit, V. Vèque, and S. Secci, "Flexible Association and Placement for Open-RAN," in *IEEE INFOCOM 2024-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS).* IEEE, 2024, pp. 1–6.

[46] K. Adachi, M. Li, P. H. Tan, Y. Zhou, and S. Sun, "Q-Learning based intelligent traffic steering in heterogeneous network," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring).* IEEE, 2016, pp. 1–5.

[47] S. Math, P. Tam, and S. Kim, "Intelligent Real-Time IoT Traffic Steering in 5G Edge Networks," *Computers, Materials & Continua*, vol. 67, no. 3, 2021.

[48] Y. Hou, K. Zhang, X. Liu, G. Chuai, W. Gao, and X. Chen, "An Inter-Slice RB Leasing and Association Adjustment Scheme in O-RAN," *IEEE Transactions on Network and Service Management*, 2023.

[49] V.-D. Nguyen, T. X. Vu, N. T. Nguyen, D. C. Nguyen, M. Juntti, N. C. Luong, D. T. Hoang, D. N. Nguyen, and S. Chatzinotas, "Network-aided intelligent traffic steering in 6G O-RAN: A multi-layer optimization framework," *IEEE Journal on Selected Areas in Communications*, 2023.

[50] A. Lacava, M. Polese, R. Sivaraj, R. Soundrarajan, B. S. Bhati, T. Singh, T. Zugno, F. Cuomo, and T. Melodia, "Programmable and customized intelligence for traffic steering in 5G networks using open RAN architectures," *IEEE Transactions on Mobile Computing*, vol. 23, no. 4, pp. 2882–2897, 2023.

[51] I. Tamim, S. Aleyadeh, and A. Shami, "Intelligent O-RAN traffic steering for uRLLC through deep reinforcement learning," in *ICC 2023-IEEE International Conference on Communications.* IEEE, 2023, pp. 112–118.

[52] R. Ntassah, G. M. Dell'area, and F. Granelli, "User Classification and Traffic Steering in O-RAN," *IEEE Open Journal of the Communications Society*, 2024.

[53] H. Erdol, X. Wang, P. Li, J. D. Thomas, R. Piechocki, G. Oikonomou, R. Inacio, A. Ahmad, K. Briggs, and S. Kapoor, "Federated meta-learning for traffic steering in O-RAN," in *2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall).* IEEE, 2022, pp. 1–7.

[54] M. A. Habib, H. Zhou, P. E. Iturria-Rivera, M. Elsayed, M. Bavand, R. Gaigalas, Y. Ozcan, and M. Erol-Kantarci, "Hierarchical reinforcement learning based traffic steering in multi-RAT 5G deployments," in *ICC 2023-IEEE International Conference on Communications.* IEEE, 2023, pp. 100–105.

[55] P. Rost, A. Banchs, I. Berberana, M. Breitbach, M. Doll, H. Droste, C. Mannweiler, M. A. Puente, K. Samdanis, and B. Sayadi, "Mobile network architecture evolution toward 5G," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 84–91, 2016.

[56] J. Rodriguez, *Fundamentals of 5G mobile networks.* John Wiley & Sons, 2015.

[57] M. S. Wani, M. Kretschmer, B. Schröder, A. Grebe, and M. Rademacher, "Open RAN: A concise overview," *IEEE Open Journal of the Communications Society*, 2024.

[58] M. A. Habibi, M. Nasimi, B. Han, and H. D. Schotten, "A comprehensive survey of RAN architectures toward 5G mobile communication system," *Ieee Access*, vol. 7, pp. 70 371–70 421, 2019.

[59] B. Khan, N. Nidhi, H. OdetAlla, A. Flizikowski, A. Mihovska, J.-F. Wagen, and F. Velez, "Survey on 5G second phase RAN architectures and functional splits," *Authorea Preprints*, 2023.

[60] X. Lin and N. Lee, "5G and Beyond," *Cham, Switzerland: Springer Nature Switzerland AG*, 2021.

[61] M. Kassi and S. Hamouda, "RAN Virtualization: How Hard Is It to Fully Achieve?" *IEEE Access*, 2024.

[62] L. Bonati, M. Polese, S. D'Oro, S. Basagni, and T. Melodia, "Open, programmable, and virtualized 5G networks: State-of-the-art and the road ahead," *Computer Networks*, vol. 182, p. 107516, 2020.

[63] M. Q. Hamdan, H. Lee, D. Triantafyllopoulou, R. Borralho, A. Kose, E. Amiri, D. Mulvey, W. Yu, R. Zitouni, R. Pozza *et al.*, "Recent advances in machine learning for network automation in the O-RAN," *Sensors*, vol. 23, no. 21, p. 8792, 2023.

[64] C. Fiandrino, L. Bonati, S. D'Oro, M. Polese, T. Melodia, and J. Widmer, "Explora: AI/ML explainability for the Open RAN," *Proceedings of the ACM on Networking*, vol. 1, no. CoNEXT3, pp. 1–26, 2023.

[65] J. Kaur, M. A. Khan, M. Iftikhar, M. Imran, and Q. E. U. Haq, "Machine learning techniques for 5G and beyond," *IEEE Access*, vol. 9, pp. 23 472–23 488, 2021.

[66] L. Gavrilovska, V. Rakovic, and D. Denkovski, "From Cloud RAN to Open RAN," *Wirel. Pers. Commun.*, vol. 113, no. 3, pp. 1523–1539, 2020.

[67] L. Bonati *et al.*, "Intelligence and learning in O-RAN for data-driven NextG cellular networks," *IEEE Commun. Mag.*, vol. 59, no. 10, pp. 21–27, 2021.

[68] S. Niknam, A. Roy, H. S. Dhillon, S. Singh, R. Banerji, J. H. Reed, N. Saxena, and S. Yoon, "Intelligent O-RAN for beyond 5G and 6G wireless networks," in *2022 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2022, pp. 215–220.

[69] M. Dryjanski and M. Szydelko, "A unified traffic steering framework for LTE radio access network coordination," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 84–92, 2016.

[70] S. Vassilaras *et al.*, "The algorithmic aspects of network slicing," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 112–119, 2017.

[71] M. Iwabuchi *et al.*, "5G field experimental trials on uRLLC using new frame structure," in *IEEE Globecom Workshops (GC Wkshps)*, 2017, pp. 1–6.

[72] M. Simsek *et al.*, "Multiconnectivity in multicellular, multiuser systems: A matching-based approach," *Proc. IEEE*, vol. 107, no. 2, pp. 394–413, 2019.

[73] N. Zhang, S. Zhang, S. Wu, J. Ren, J. W. Mark, and X. Shen, "Beyond coexistence: Traffic steering in LTE networks with unlicensed bands," *IEEE Wireless Commun.*, vol. 23, no. 6, pp. 40–46, 2016.

[74] M. Dryjański, Kułacz, and A. Kliks, "Toward Modular and Flexible Open RAN Implementations in 6G Networks: Traffic Steering Use Case and O-RAN xApps," *Sensors*, vol. 21, no. 24, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/24/8173

[75] J. Choi and R. W. Heath, "Interpolation based transmit beamforming for MIMO-OFDM with limited feedback," *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4125–4135, 2005.

[76] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, 2010.

[77] S. Schiessl *et al.*, "Delay analysis for wireless fading channels with finite blocklength channel coding," in *Proc. 18th ACM Inter. Conf. Model. Anal. and Simul. Wire. and Mob. Sys.*, 2015, pp. 13–22.

[78] P. J. Burke, "The output of a queuing system," *Oper. Res.*, vol. 4, no. 6, pp. 699–704, 1956.

[79] E. Che *et al.*, "Joint Optimization of Cooperative Beamforming and Relay Assignment in Multi-User Wireless Relay Networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 10, p. 5481–5495, 2014.

[80] A. Beck, A. Ben-Tal, and L. Tetruashvili, "A sequential parametric convex approximation method with applications to nonconvex truss topology design problems," *J. Global Optim.*, vol. 47, no. 1, pp. 29–51, May 2010.

[81] C.-X. Wang, M. Di Renzo, S. Stanczak, S. Wang, and E. G. Larsson, "Artificial intelligence enabled wireless networking for 5G and beyond: Recent advances and future challenges," *IEEE Wireless Communications*, vol. 27, no. 1, pp. 16–23, 2020.

[82] S. Niknam, H. S. Dhillon, and J. H. Reed, "Federated learning for wireless communications: Motivation, opportunities, and challenges," *IEEE Communications Magazine*, vol. 58, no. 6, pp. 46–51, 2020.

[83] M.-T. Suer, C. Thein, H. Tchouankem, and L. Wolf, "Multi-connectivity as an enabler for reliable low latency communications—An overview," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 156–169, 2019.

[84] H. Arslan *et al.*, "Flexible multi-numerology systems for 5G new radio," *Journal of Mobile Multimedia*, 2018.

[85] F. Kavehmadavani, V.-D. Nguyen, T. X. Vu, and S. Chatzinotas, "Traffic Steering for eMBB and uRLLC Coexistence in Open Radio Access Networks," in *2022 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2022, pp. 242–247.

[86] S. Niknam, A. Roy, H. S. Dhillon, S. Singh, R. Banerji, J. H. Reed, N. Saxena, and S. Yoon, "Intelligent O-RAN for beyond 5G and 6G wireless networks," in *2022 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2022, pp. 215–220.

[87] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural computation*, vol. 31, no. 7, pp. 1235–1270, 2019.

[88] M. I. Kamel, L. B. Le, and A. Girard, "LTE wireless network virtualization: Dynamic slicing via flexible scheduling," in *2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall)*. IEEE, 2014, pp. 1–5.

[89] G. Pocovi, K. I. Pedersen, and P. Mogensen, "Joint link adaptation and scheduling for 5G ultra-reliable low-latency communications," *Ieee Access*, vol. 6, pp. 28 912–28 922, 2018.

[90] A. Karimi, K. I. Pedersen, N. H. Mahmood, G. Pocovi, and P. Mogensen, "Efficient low complexity packet scheduling algorithm for mixed uRLLC and eMBB traffic in 5G," in *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*. IEEE, 2019, pp. 1–6.

[91] Z. Wu, F. Zhao, and X. Liu, "Signal space diversity aided dynamic multiplexing for eMBB and uRLLC traffics," in *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*. IEEE, 2017, pp. 1396–1400.

[92] A. Anand, G. De Veciana, and S. Shakkottai, "Joint scheduling of uRLLC and eMBB traffic in 5G wireless networks," *IEEE/ACM Transactions on Networking*, vol. 28, no. 2, pp. 477–490, 2020.

[93] P. Korrai, E. Lagunas, S. K. Sharma, S. Chatzinotas, A. Bandi, and B. Ottersten, "A RAN resource slicing mechanism for multiplexing of eMBB and uRLLC services in OFDMA based 5G wireless networks," *IEEE Access*, vol. 8, pp. 45 674–45 688, 2020.

[94] L. You, Q. Liao, N. Pappas, and D. Yuan, "Resource optimization with flexible numerology and frame structure for heterogeneous services," *IEEE Communications Letters*, vol. 22, no. 12, pp. 2579–2582, 2018.

[95] P. K. Korrai, E. Lagunas, A. Bandi, S. K. Sharma, and S. Chatzinotas, "Joint power and resource block allocation for mixed-numerology-based 5G downlink under imperfect CSI," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 1583–1601, 2020.

[96] O. Alliance, "O-RAN: Towards an open and smart RAN," *[Online]*, 2018.

[97] A. B. Kihero, M. S. J. Solaija, and H. Arslan, "Inter-numerology interference for beyond 5G," *IEEE Access*, vol. 7, pp. 146 512–146 523, 2019.

[98] G. Zheng, I. Krikidis, C. Masouros, S. Timotheou, D.-A. Toumpakaris, and Z. Ding, "Rethinking the role of interference in wireless networks," *IEEE Communications Magazine*, vol. 52, no. 11, pp. 152–158, 2014.

[99] O. Alliance, "O-RAN Working Group 2 AI/ML Workflow Description and Requirements," *ORAN-WG2. AIML. v01*, vol. 1, 2019.

[100] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Operations Research*, vol. 26, no. 4, pp. 681–683, July-Aug. 1978.

[101] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization.* Philadelphia: MPS-SIAM Series on Optimi., SIAM, 2001.

[102] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Toward 6G networks: Use cases and technologies," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 55–61, 2020.

[103] D. Wypiór, M. Klinkowski, and I. Michalski, "Open RAN—Radio access network evolution, benefits and market trends," *Applied Sciences*, vol. 12, no. 1, p. 408, 2022.

[104] S. K. Singh, R. Singh, and B. Kumbhani, "The evolution of radio access network towards open-RAN: Challenges and opportunities," in *2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW).* IEEE, 2020, pp. 1–6.

[105] A. S. Abdalla, P. S. Upadhyaya, V. K. Shah, and V. Marojevic, "Toward Next Generation Open Radio Access Networks: What O-RAN Can and Cannot Do!" *IEEE Network*, vol. 36, no. 6, pp. 206–213, 2022.

[106] O-RAN.WG1.Use-Cases-Detailed-Specification-R003-v13.00, "O-RAN Use Cases Detailed Specification 13.0," *Technical Specification*, February 2024. [Online]. Available: https://www.o-ran.org/specifications(accessedonFebruary2024)

[107] SAMSUNG, "ORAN - The Open Road to 5G," *[Online]*, White Paper, July 2019, avaiable: https://www.samsung.com/global/business/networks/insights/whitepapers/O.

[108] M. Tayyab, X. Gelabert, and R. Jäntti, "A survey on handover management: From LTE to NR," *IEEE Access*, vol. 7, pp. 118 907–118 930, 2019.

[109] S. Vassilaras, L. Gkatzikis, N. Liakopoulos, I. N. Stiakogiannakis, M. Qi, L. Shi, L. Liu, M. Debbah, and G. S. Paschos, "The Algorithmic Aspects of Network Slicing," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 112–119, 2017.

[110] L. Weedage, C. Stegehuis, and S. Bayhan, "Impact of Multi-Connectivity on Channel Capacity and Outage Probability in Wireless Networks," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 6, pp. 7973–7986, 2023.

[111] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5G: Survey and challenges," *IEEE communications magazine*, vol. 55, no. 5, pp. 94–100, 2017.

[112] R. C. Notes, "Document 3GPP TSG RAN WG1 Meeting# 86," *Gothenburg, Sweden, Aug*, 2016.

[113] A. Yazar and H. Arslan, "A flexibility metric and optimization methods for mixed numerologies in 5G and beyond," *IEEE Access*, vol. 6, pp. 3755–3764, 2018.

[114] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133–3174, 2019.

[115] Y. Huang, S. Li, C. Li, Y. T. Hou, and W. Lou, "A deep-reinforcement-learning-based approach to dynamic eMBB/uRLLC multiplexing in 5G NR," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6439–6456, 2020.

[116] M. Awais, A. Ahmed, M. Naeem, M. Iqbal, W. Ejaz, A. Anpalagan, and H. S. Kim, "Efficient joint user association and resource allocation for cloud radio access networks," *IEEE Access*, vol. 5, pp. 1439–1448, 2017.

[117] P. Munoz, R. Barco, D. Laselva, and P. Mogensen, "Mobility-based strategies for traffic steering in heterogeneous networks," *IEEE Communications Magazine*, vol. 51, no. 5, pp. 54–62, 2013.

[118] L. E. Chatzieleftheriou, A. Destounis, G. Paschos, and I. Koutsopoulos, "Blind Optimal User Association in Small-Cell Networks," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[119] P. L. Vo, M. N. Nguyen, T. A. Le, and N. H. Tran, "Slicing the edge: Resource allocation for RAN network slicing," *IEEE Wireless Communications Letters*, vol. 7, no. 6, pp. 970–973, 2018.

[120] P. Guan, D. Wu, T. Tian, J. Zhou, X. Zhang, L. Gu, A. Benjebbour, M. Iwabuchi, and Y. Kishiyama, "5G field trials: OFDM-based waveforms and mixed numerologies," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1234–1243, 2017.

[121] T. Bag, S. Garg, Z. Shaik, and A. Mitschele-Thiel, "Multi-numerology based resource allocation for reducing average scheduling latencies for 5G NR wireless networks," in *2019 European Conference on Networks and Communications (EuCNC)*. IEEE, 2019, pp. 597–602.

[122] L. Marijanovic, S. Schwarz, and M. Rupp, "A novel optimization method for resource allocation based on mixed numerology," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.

[123] F. Kavehmadavani, V.-D. Nguyen, T. X. Vu, and S. Chatzinotas, "Intelligent Traffic Steering in Beyond 5G Open RAN based on LSTM Traffic Prediction," *IEEE Transactions on Wireless Communications*, 2023.

[124] H. Yang, X. Xie, and M. Kadoch, "Intelligent resource management based on reinforcement learning for ultra-reliable and low-latency IoV communication networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4157–4169, 2019.

[125] P. Muñoz, R. Barco, and I. de la Bandera, "Load balancing and handover joint optimization in LTE networks using fuzzy logic and reinforcement learning," *Computer Networks*, vol. 76, pp. 112–125, 2015.

[126] Y. Li, C. Hu, J. Wang, and M. Xu, "Optimization of uRLLC and eMBB multiplexing via deep reinforcement learning," in *2019 IEEE/CIC International Conference on Communications Workshops in China (ICCC Workshops)*. IEEE, 2019, pp. 245–250.

[127] J. Zhang, X. Xu, K. Zhang, B. Zhang, X. Tao, and P. Zhang, "Machine learning based flexible transmission time interval scheduling for eMBB and uRLLC coexistence scenario," *IEEE Access*, vol. 7, pp. 65 811–65 820, 2019.

[128] O-RAN.WG2.Use-Case-Requirements-v02.01, "Non-RT RIC & A1 Interface: Use Cases and Requirements," *Technical Specification*, Nov. 2021. [Online]. Available: https://www.o-ran.org/specifications(accessedon10November2021)

[129] L. Liang, H. Ye, and G. Y. Li, "Spectrum sharing in vehicular networks based on multi-agent reinforcement learning," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2282–2292, 2019.

[130] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Network*, vol. 34, no. 3, pp. 134–142, 2019.

[131] O. R. Alliance, "ORAN-WG1 O-RAN Architecture Description-v01. 00.00," *Technical Specification*, 2020.

[132] ORAN Alliance, "O-RAN: Towards an open and smart RAN," https://www.o-ran.org/resources, 2018.

[133] S.-Y. Lien, S.-L. Shieh, Y. Huang, B. Su, Y.-L. Hsu, and H.-Y. Wei, "5G new radio: Waveform, frame structure, multiple access, and initial access," *IEEE communications magazine*, vol. 55, no. 6, pp. 64–71, 2017.

[134] P. Schulz, M. Matthe, H. Klessig, M. Simsek, G. Fettweis, J. Ansari, S. A. Ashraf, B. Almeroth, J. Voigt, I. Riedel *et al.*, "Latency critical IoT applications in 5G: Perspective on the design of radio interface and network architecture," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 70–78, 2017.

[135] 3GPP, "Study on New Radio (NR) Access Technology Physical Layer Aspects," *Technical Report (TR) 38.802, V14. 2.0*, 2017.

[136] PARALLELWIRELESS, "Open RAN Functional Splits, Explained," *[Online]*, 2021, avaiable: https://www.parallelwireless.com/wp-content/uploads/WP-Open-RAN-Functional-Splits_2021.pdf.

[137] H. Zhang, L. Song, and Y. J. Zhang, "Load balancing for 5G ultra-dense networks using device-to-device communications," *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 4039–4050, 2018.

[138] K. Ghanem, H. Alradwan, A. Motermawy, and A. Ahmad, "Reducing ping-pong Handover effects in intra EUTRA networks," in *2012 8th International Symposium on Communication Systems, Networks & Digital Signal Processing (CSNDSP)*. IEEE, 2012, pp. 1–5.

[139] U. Siddique, H. Tabassum, E. Hossain, and D. I. Kim, "Channel-access-aware user association with interference coordination in two-tier downlink cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 7, pp. 5579–5594, 2015.

[140] R. Arshad, H. ElSawy, S. Sorour, T. Y. Al-Naffouri, and M.-S. Alouini, "Handover management in 5G and beyond: A topology aware skipping approach," *IEEE Access*, vol. 4, pp. 9073–9081, 2016.

[141] P. Szilágyi, Z. Vincze, and C. Vulkán, "Integrated mobility load balancing and traffic steering mechanism in LTE," in *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, 2013, pp. 2148–2153.

[142] M. Mollel, S. Kaijage, and K. Michael, "Deep reinforcement learning based handover management for millimeter wave communication," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 2, 2021.

[143] Y. Cao, L. Zhang, and Y.-C. Liang, "Deep reinforcement learning for multi-user access control in UAV networks," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.

[144] Y. Cao, S.-Y. Lien, Y.-C. Liang, K.-C. Chen, and X. Shen, "User access control in open radio access networks: A federated deep reinforcement learning approach," *IEEE Transactions on Wireless Communications*, vol. 21, no. 6, pp. 3721–3736, 2021.

[145] Y. S. Nasir and D. Guo, "Multi-agent deep reinforcement learning for dynamic power allocation in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2239–2250, 2019.

[146] A. Garcia-Saavedra, X. Costa-Perez, D. J. Leith, and G. Iosifidis, "Fluidran: Optimized vran/mec orchestration," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 2366–2374.

[147] 3rd Generation Partnership Project (3GPP), "Technical Specification Group Radio Access Network; Study on New Radio Access Technology: Radio Access Architecture and Interfaces Release 14," 3GPP, Sophia Antipolis, France, Tech. Rep. 38.801, 2017.

[148] NGMN, "5G extreme requirements: Radio access network solutions," O-RAN Alliance, Tech. Rep., 2018.

[149] F. Kavehmadavani, V.-D. Nguyen, T. X. Vu, and S. Chatzinotas, "Empowering Traffic Steering in 6G Open RAN with Deep Reinforcement Learning," *IEEE Transactions on Wireless Communications*, 2024.

[150] D. López-Pérez, A. Ladányi, A. Jüttner, H. Rivano, and J. Zhang, "Optimization method for the joint allocation of modulation schemes, coding rates, resource blocks and power in self-organizing LTE networks," in *2011 Proceedings IEEE INFOCOM*. IEEE, 2011, pp. 111–115.

[151] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures (Release 15)," 3GPP, Tech. Spec. 36.213, Mar 2018. [Online]. Available: http://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2427

[152] M. K. Motalleb, V. Shah-Mansouri, S. Parsaeefard, and O. L. A. López, "Resource allocation in an Open RAN system using network slicing," *IEEE Transactions on Network and Service Management*, vol. 20, no. 1, pp. 471–485, 2022.

[153] 3GPP, "5G; NG-RAN; Architecture description (3GPP TS 38.401 version 16.3.0 Release 16)," 3GPP, Tech. Spec. 38.401, Nov 2020. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/138400_138499/138401/16.03.00_60/ts_138401v160300p.pdf

[154] F. Meng, P. Chen, L. Wu, and J. Cheng, "Power allocation in multi-user cellular networks: Deep reinforcement learning approaches," *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 6255–6267, 2020.