This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2024.3396273

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

Fatemeh Kavehmadavani, Van-Dinh Nguyen, *Senior Member, IEEE*,
Thang X. Vu, *Senior Member, IEEE* and Symeon Chatzinotas, *Fellow, IEEE*

***Abstract*—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. Our paper 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, 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 throughput by an average of 99.42% over slice isolation.**

***Index Terms*—Deep reinforcement learning, open radio access network, traffic steering, network intelligence, traffic prediction, intelligent radio resource management.**

I. 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 [1]. 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 [2]. 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 [3].

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)/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 [4]. These lead to creating an 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 [5].

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 [6]. 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 [7]. 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 [8], [9]. 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

F. Kavehmadavan, T. X. Vu, and S. Chatzinotas are with the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg (email: {fatemeh.kavehmadavani, thang.vu, symeon.chatzinotas}@uni.lu).

V.-D. Nguyen is with the College of Engineering and Computer Science and also with Center for Environmental Intelligence, VinUniversity, Vinhomes Ocean Park, Gia Lam District, Hanoi 100000, Vietnam (e-mail: dinh.nv2@vinuni.edu.vn).

This work was 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). For the purpose of open access, and in fulfilment of the obligations arising from the grant agreement, the author has applied a Creative Commons Attribution 4.0 International (CC BY 4.0) license to any Author Accepted Manuscript version arising from this submission. The work of V.-D. Nguyen was supported in part by the VinUniversity Seed Grant Program.

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 throughput [10]. NS segregates multiple types of traffic into logical network slices while using the same infrastructure, enabling simultaneous transmission on one channel [11]. 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 [12]. It subdivides each time slot into two mini-slots, each with 7 orthogonal frequency-division multiplexing (OFDM) symbols [13], reducing slot duration and transmission/frame alignment time.

A. 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 scheduling 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 long-short-term memory (RNN 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 [14], while overcoming long-term dependencies and gradient issues [15]. However, the DRL model excels in complicated algorithmic learning, extreme generalization, and dynamic wireless environments [14]. 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 throughput, 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 throughput) 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 throughput.

The subsequent sections of this paper are as follows. Section II provides the related literature on TS in the traditional RAN and Open RAN paradigms. Section III studies a general overview of Open RAN's concept and architecture, while Section IV introduces the system model. In Section V, the optimization problem is formulated, along with an overview of the DRL-based intelligent TS algorithm and its structure.

Section VI 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 VII. Finally, Section VIII concludes the study, summarizing key findings and insights.

II. Related Works

A. RAN Radio Resource Management

Extensive research studies have explored dynamic RAN resource allocation mechanisms in the traditional cellular RAN architecture. For instance, [16] and [17] have proposed a joint scheduling design for uRLLC and eMBB coexistence to maximize the overall throughput 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, [18] has used a preemptive puncturing approach, assigning contiguous RBs to the uRLLC and eMBB services. In [19], 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 [20] have proposed a dynamic MC-based joint scheduling scheme with traffic steering for eMBB and uRLLC services to achieve high throughput 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 [21] 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 can not 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.

B. Traffic Steering in Traditional RAN Architecture

In the literature, there are some studies that have primarily investigated TS in traditional RAN networks. For example, [22] introduced a TS technique to maximize QoE for eMBB users in the MC scenario. In [23], a proposal was presented to alter user association in overlapped ultra-dense networks. Furthermore, [24] 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 [25], a RAN slicing-based radio resource allocation scheme was proposed for dynamic TS RAN moderation in 5G. Praveen *et. al.* [26] 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, [27] explored flexible numerologies in the frequency domain to enhance the capacity of services with nonuniform requirements. Experimental field tests in [28] demonstrated the effectiveness of multiplexing mixed numerologies in the frequency domain for the performance assessment of OFDM-based 5G waveforms. In [29], 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, [30] suggested selecting a single numerology per time slot based on service priorities in multi-numerology resource allocation. In [31], 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 [32] 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 throughput while minimizing uRLLC latency.

C. 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 [33] 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. [34] has proposed a unified self-management mechanism based on fuzzy logic and RL to tune the handover parameters of adjacent cells. In [35] and [36], 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 [37] 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. [38] 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 [39], 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 [40], 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 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.

III. Intelligent TS Deployment on Open RAN Architecture

A. Open RAN Background

Fig. 1 shows the learning-based Open RAN architecture relying on the O-RAN Alliance [41] 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 [42]. 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) [43]. 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 [32], the system model's layers operate at different timescales, ranging from 1 to thousands of ms. Non-RT RIC handles activities such as service provisioning, design, policy definition, and AI/ML model training at intervals greater than 1 s, and hosts remote applications (referred to as rApps). However, the near-RT RIC heads tasks with timescales surpassing 10 ms, 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.

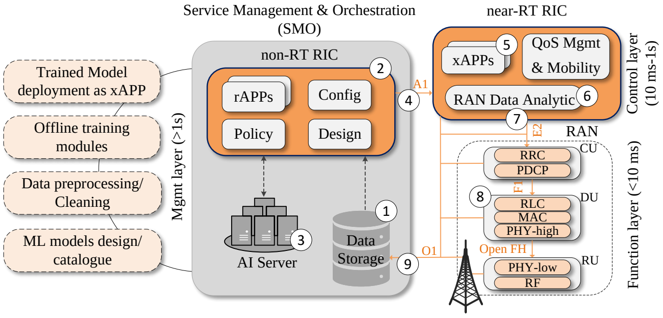


Fig. 1. Learning integration in Open RAN architecture based on the O-RAN Alliance.

B. 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. 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.*,), 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.

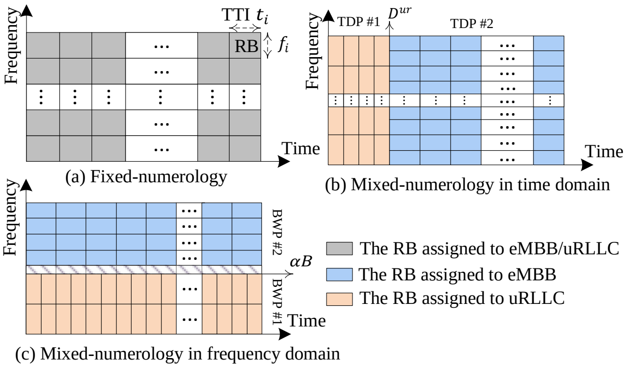
IV. System Model

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. 1 with the RAN part). Towards cost-effectiveness, each DU forms a cluster of RUs. Let $\mathcal{N} \triangleq \{1,\dots,N\}$ and $\mathcal{M} \triangleq \{1,\dots,M\}$ 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 $\mathcal{U} \triangleq \{1,\dots,U\}$ through the shared wireless medium. To deploy the coexistence of uRLLC and eMBB, we divide the set of users into two disjoint sets: $\mathcal{U}^{\mathsf{ur}} \triangleq \{1,\dots,U^{\mathsf{ur}}\}$ of uRLLC users and $\mathcal{U}^{\mathsf{em}} \triangleq \{1,\dots,U^{\mathsf{em}}\}$ of eMBB users, with $\mathcal{U} \triangleq \mathcal{U}^{\mathsf{ur}} \cup \mathcal{U}^{\mathsf{em}}$. 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 $\mathcal{M}_n \triangleq \{(n,1),\dots,(n,M_n)\}$ with $\sum_{n\in\mathcal{N}} M_n = M$.

A. 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 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. 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 $f_i = \{1,\dots,F_i\}$ and the number of TTIs per frame indexed as $t_i = \{1,\dots,T_i\}$, respectively.



(a) Fixed-numerology

(b) Mixed-numerology in time domain

(c) Mixed-numerology in frequency domain

Fig. 2. Illustration of fixed numerology and mixed numerologies in frequency and time domains.

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. 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 [44], 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. 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 $T_i = \Delta/\delta_i$.

B. Transmission Model and Downlink Throughput

In Fig. 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-serve 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 optimize the system performance. In the discrete time frame indexed by $t \in \{1,2,\dots,T\}$, 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 by $\boldsymbol{G}[t] \triangleq \big[g^{f_i,t_i}_{m,u}\big]|_{\forall m,u,f_i}$ 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 study defines the RB matrices $\boldsymbol{\pi}[t] \triangleq \big[\boldsymbol{\pi}_{m,u}[t]\big]^T_{\forall m,u}$ 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

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

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 $0 \le p^{f_i,t_i}_{m,u} \le \pi^{f_i,t_i}_{m,u}[t]P^{\max}_m; \forall u \in \mathcal{U}^{\mathsf{em}}$ 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 $\sum_{i\in\{1,2\}} R^{\mathsf{em}}_{m,u}(\boldsymbol{p}[t_i]) \ge \varphi_{m,u}[t]R^{\mathsf{th}}$ 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

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

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 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 paper we approximate $V = 1 - \frac{1}{1+(\Gamma^{\mathsf{ur}})^2} \approx 1$ [45]. To meet the Big-M formulation theory and the approximation $V \approx 1$, we impose the constraint $\frac{N_0\Gamma^{\mathsf{ur}}}{g^{f_i,t_i}_{m,u}}\pi^{f_i,t_i}_{m,u}[t] \le p^{f_i,t_i}_{m,u} \le \pi^{f_i,t_i}_{m,u}[t]P^{\max}_m; \forall u \in \mathcal{U}^{\mathsf{ur}}$. Besides, the constraint $\sum_i\sum_{f_i,u} p^{f_i,t_i}_{m,u} \le P^{\max}_m, \forall u \in \mathcal{U}$ 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

$$\mathcal{P}[t_i] = \left\{\boldsymbol{p}[t_i], \forall t_i \,\middle|\, \begin{aligned} & 0 \le p^{f_i,t_i}_{m,u} \le \pi^{f_i,t_i}_{m,u}[t]P^{\max}_m; \forall u \in \mathcal{U}^{\mathsf{em}} \\ & \frac{N_0\Gamma^{\mathsf{ur}}}{g^{f_i,t_i}_{m,u}}\pi^{f_i,t_i}_{m,u}[t] \le p^{f_i,t_i}_{m,u} \le \pi^{f_i,t_i}_{m,u}[t]P^{\max}_m; \forall u \in \mathcal{U}^{\mathsf{ur}} \\ & \sum_{i\in\{1,2\}}\sum_{f_i,u} p^{f_i,t_i}_{m,u} \le P^{\max}_m,\ \forall m, t_i, u \in \mathcal{U} \end{aligned}\right\}. \tag{3}$$

C. 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 throughput, 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.

$$\sum_{t_i=1}^{D^{\mathsf{ur}}/\delta_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i}[t] \geq e_u^{\mathsf{ur}}[t], \ \forall t, u \in \mathscr{U}^{\mathsf{ur}}, i = 1 \tag{4}$$

where $e_u^{\mathsf{ur}}[t] = \left\lceil \max\left(\lambda_u^{\mathsf{ur}}[t] - \Omega_u[t], 0\right)/2 \right\rceil$ 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 throughput given

$$\sum_{t_i=1}^{T_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i}[t] \geq e_u^{\mathsf{em}}[t], \ \forall t, u \in \mathscr{U}^{\mathsf{em}}, i = 2 \tag{5}$$

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. an RB can only be allotted to a single user. As a result, we define the set of RB allocation constraints as

$$\begin{aligned}\mathbf{\Lambda}[t] = \Big\{ \boldsymbol{\pi}[t], \forall t \Big| \pi_{m,u}^{f_i,t_i}[t] \in \{0,1\}, \sum_{m,u} \pi_{m,u}^{f_i,t_i}[t] \leq 1; \forall f_i, t_i,\\ \sum_{t_i=1}^{D^{\mathsf{ur}}/\delta_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i}[t] \geq e_u^{\mathsf{ur}}[t]|_{\forall t, u \in \mathscr{U}^{\mathsf{ur}}, i=1},\\ \sum_{t_i=1}^{T_i} \sum_{f_i=1}^{F_i} \pi_{m,u}^{f_i,t_i}[t] \geq e_u^{\mathsf{em}}[t]|_{\forall t, u \in \mathscr{U}^{\mathsf{em}}, i=2} \Big\}.\end{aligned} \tag{6}$$

D. Network Queues and e2e Latency

We denote $\boldsymbol{\lambda}[t] \triangleq \left[\lambda_u^{\mathsf{x}}[t]\right]_{\forall u, \mathsf{x}}$ 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 $\lambda^{\max} < \infty$. 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

$$\begin{aligned} q_{m,u}[t_i] = \max\big(&q_{m,u}[t_i - 1] + \varphi_{m,u}[t]\lambda_u^{\mathsf{x}}[t]Z^{\mathsf{x}}\Delta \\ &- R_{m,u}^{\mathsf{x}}(\boldsymbol{p}[t_i])\delta_i, 0\big) \end{aligned} \tag{7}$$

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 $\sum_u q_{m,u}[t_i] \leq q_m^{\max}$, $\forall m$ 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 $\boldsymbol{q}[t_i] \triangleq \left[q_{m,u}[t_i]\right]_{\forall m,u}^T$. At each frame, the proposed model predicts data arrivals and flow-split decisions based on analyzing the queues' status and provides the RBs' allocation according.

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

$$\begin{aligned} \tau_u^{\mathsf{ur}}[t] = &\tau_{cu}^{\mathsf{pro}}[t] + \tau_{cu,du}^{\mathsf{tx}}[t] + \tau_{du}^{\mathsf{pro}}[t] + \tau_{du,ru}^{\mathsf{tx}}[t] \\ &+ \tau_{ru,u}^{\mathsf{tx}}[t] + \tau_{ru}^{\mathsf{pro}}[t], \ \forall u \in \mathscr{U}^{\mathsf{ur}} \end{aligned} \tag{8}$$

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 [32]. 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 $\tau_{ru,u}^{\mathsf{tx}}[t] = \delta_i . \arg\max_{t_i}\{\pi_{m,u}^{f_i,t_i}[t]\}$ 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.*, $\tau_u^{\mathsf{ur}}[t] \approx \tau_{ru,u}^{\mathsf{tx}}[t] \leq D^{\mathsf{ur}}$.

V. Problem Formulation

Utility Function: This paper 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 throughput 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 higher eMBB throughput. Besides, controlling congestion to avoid large queues is crucial, especially for eMBB traffic with large packet sizes. This work 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 $\omega \sum_{u \in \mathscr{U}^{\mathsf{em}}} \frac{\bar{q}_u}{q_0} + (1 - \omega) \max_{u \in \mathscr{U}^{\mathsf{ur}}} \frac{\delta_i . \mathbb{E}_t\left(\arg\max_{t_i}\{\pi_{m,u}^{f_i,t_i}[t]\}\right)}{\tau_0}$, 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

$$\min_{\boldsymbol{\lambda},\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p}} \quad \omega \sum_{u\in\mathscr{U}^{\mathsf{em}}} \frac{\bar{q}_u}{q_0} + (1-\omega)\frac{\bar{t}^{\mathsf{ur}}}{\tau_0} \tag{9a}$$

$$\text{s.t.} \quad \boldsymbol{\pi}[t] \in \boldsymbol{\Lambda}[t],\ \forall t \tag{9b}$$

$$\boldsymbol{p}[t_i] \in \mathscr{P}[t_i], \forall t_i \tag{9c}$$

$$\boldsymbol{\varphi}[t] \in \boldsymbol{\Psi}[t],\ \forall t \tag{9d}$$

$$\sum_i R^{\mathsf{em}}_{m,u}(\boldsymbol{p}[t_i]) \geq \varphi_{m,u}[t]R^{\mathsf{th}},\ \forall m \in \mathscr{M}_n, u \in \mathscr{U}^{\mathsf{em}} \tag{9e}$$

$$\Gamma^{\mathsf{ur}}(\boldsymbol{\pi}[t], \boldsymbol{p}[t_i]) \geq \pi^{f_i,t_i}_{m,u}[t]\Gamma^{\mathsf{th}} \tag{9f}$$

$$\tau^{\mathsf{ur}}_u(\boldsymbol{\pi}[t]) \leq D^{\mathsf{ur}},\ \forall u \in \mathscr{U}^{\mathsf{ur}} \tag{9g}$$

$$\sum_u q_{m,u}[t_i] \leq q^{\max},\ \forall t_i, m \in \mathscr{M}_n, u \in \mathscr{U} \tag{9h}$$

where $\bar{t}^{\mathsf{ur}} \triangleq \max_{u\in\mathscr{U}^{\mathsf{ur}}} \delta_i.\mathbb{E}_t\big(\arg\max_{t_i}\{\pi^{f_i,t_i}_{m,u}[t]\}\big)$. 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 (9f) guarantees that the SNR of each RB assigned to the user u-th uRLLC via m-th RU must be greater than $\pi^{f_i,t_i}_{m,u}[t]\Gamma^{\mathsf{th}}$, where Γ^{th} represents the given SNR threshold for each RB assigned to the uRLLC service.

Challenges of solving problem (9): The non-convexity of constraints (9f) and (9h) and the binary nature of the RB assignment variables in constraint (9b) make problem (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 work, 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 (9) is needed. MADRL is a promising technique to solve non-convex optimization problems with reduced computational complexity. The proposed Algorithm 1 summarizes the overall approach to solve problem (9), with detailed solutions for each step to follow. Fig. 3 shows the end-to-end high-level intelligent TS deployment.

Algorithm 1 Proposed Intelligent TS Algorithm for Solving Problem (9)

```
Initialization: Set $t = 1$, $\boldsymbol{\varphi}_u[1] = \frac{1}{M}\mathbf{1}_{M\times 1}, \forall u$, and the initial
    queues are set to be empty $q_{m,u}[1] = 0$.
for $t = 1, 2, \ldots, T$ do {/long-timescale}
    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 (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 $t_i = 1, 2, \ldots, T_i$ do {/short-timescale}
        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 (15) to obtain power allocation $\boldsymbol{p}^*[t_i]$.
        Updating queue-lengths: Queue-lengths are updated as:
            $q_{m,u}[t_i] = \max\big\{\big(q_{m,u}[t_i-1] + \varphi_{m,u}[t]\lambda^{\mathsf{x}}_u[t]Z^{\mathsf{x}}\delta_i$
                $- R^{\mathsf{x}}_{m,u}(\boldsymbol{p}^*[t_i])\delta_i\big), 0\big\}, \mathsf{x} \in \{\mathsf{ur}, \mathsf{em}\}$.
    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
```

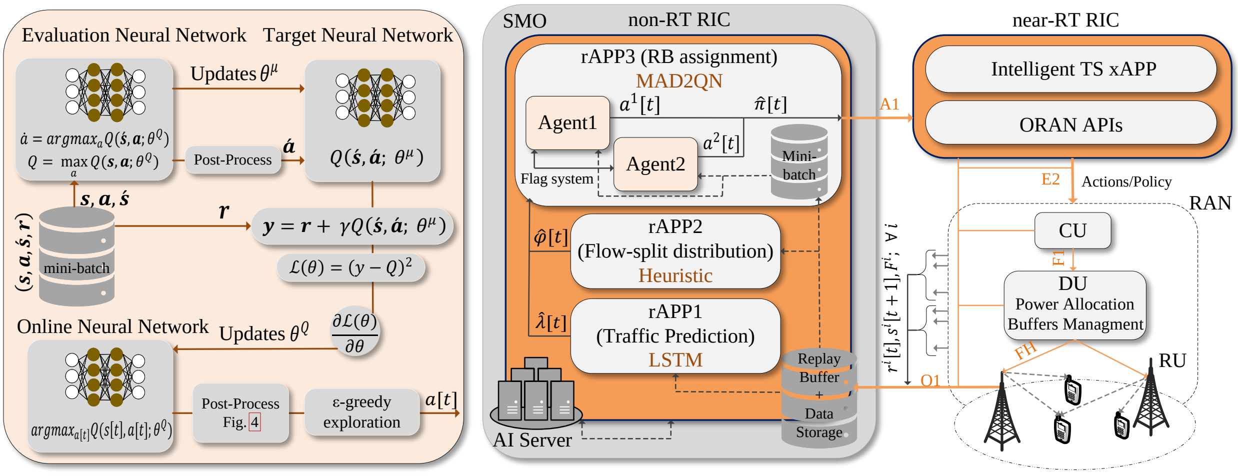
VI. The Proposed Algorithms

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 paper is to leverage observable historical system knowledge via the O1 interface to build a smoother optimal response.

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

The main challenge in 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 $\lambda^{\mathsf{x}}_u[t]$ 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 [32]. 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


Evaluation Neural Network
Target Neural Network
Updates θ^{μ}
$\dot{a} = argmax_a Q(\acute{s}, \boldsymbol{a}; \theta^Q)$
$Q = \max_a Q(\boldsymbol{s}, \boldsymbol{a}; \theta^Q)$
Post-Process
$\acute{\boldsymbol{a}}$
$Q(\acute{\boldsymbol{s}}, \acute{\boldsymbol{a}}; \theta^{\mu})$
$\boldsymbol{s}, \boldsymbol{a}, \acute{\boldsymbol{s}}$
$\boldsymbol{r}$
$(\boldsymbol{s}, \boldsymbol{a}, \acute{\boldsymbol{s}}, \boldsymbol{r})$
mini-batch
$\boldsymbol{y} = \boldsymbol{r} + \gamma Q(\acute{\boldsymbol{s}}, \acute{\boldsymbol{a}}; \theta^{\mu})$
$\mathcal{L}(\theta) = (y - Q)^2$
Online Neural Network
Updates θ^Q
$\frac{\partial \mathcal{L}(\theta)}{\partial \theta}$
$argmax_{a[t]} Q(s[t], a[t]; \theta^Q)$
Post-Process Fig. 4
ε-greedy exploration
$a[t]$
SMO
non-RT RIC
rAPP3 (RB assignment)
MAD2QN
Agent1
$a^1[t]$
$\hat{\pi}[t]$
Agent2
$a^2[t]$
Mini-batch
Flag system
$\hat{\varphi}[t]$
rAPP2 (Flow-split distribution)
Heuristic
$\hat{\lambda}[t]$
rAPP1 (Traffic Prediction)
LSTM
Replay Buffer + Data Storage
AI Server
A1
near-RT RIC
Intelligent TS xAPP
ORAN APIs
E2
Actions/Policy
RAN
CU
F1
DU
Power Allocation
Buffers Managment
FH
RU
O1
$r^i[t], s^i[t+1], F^i; \forall i$

(a) Structure of Double DQN (D2QN) agent (b) The streamlined workflow for deploying ML applications in the Open RAN architecture.

Fig. 3. A comprehensive workflow for intelligent TS deployment using ML application in Open RAN architecture.

learn flow-split decisions and RB assignments between traffic flows. The long-term variables 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.

B. 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

$$\hat{\varphi}_{m,u}[t] = \frac{\bar{R}_{m,u}[t]}{\sum_{m \in \mathcal{M}_n} \bar{R}_{m,u}[t]}, \ \forall m, u. \tag{10}$$

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 $\boldsymbol{\pi}[t]$.

C. 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 work.

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. 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

$$Q(\mathbf{s},\mathbf{a};\boldsymbol{\theta}) \leftarrow (1-\eta)Q(\mathbf{s},\mathbf{a};\boldsymbol{\theta}) + \eta\Big(\mathbf{r}+\gamma\max_{\mathbf{a}} Q(\mathbf{s}',\mathbf{a};\boldsymbol{\theta})\Big) \tag{11}$$

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 $y = \mathbf{r} + \gamma Q(\mathbf{s}', \arg\max_{\mathbf{a}} Q(\mathbf{s}',\mathbf{a};\boldsymbol{\theta}^Q);\boldsymbol{\theta}^\mu)$), which is expressed as the loss function $\mathcal{L}(\boldsymbol{\theta}^Q)$

$$\mathcal{L}(\boldsymbol{\theta}^Q) = \mathbb{E}\big(y - Q(\mathbf{s},\mathbf{a};\boldsymbol{\theta}^Q)\big)^2 \tag{12}$$

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 a 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 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 properties of the intrinsic 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 [46]. 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.

Algorithm 2 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 $C^{\max}$ and reward values to $\mathbf{r}^i[t] = 0$.
for epoch do
    Receive initial states for all agents $\mathbf{s}[1]$;
    for $t = 1, 2, \ldots, T$ do
        for agent do
            Generate a random number $\mathrm{rand}()$;
            if $\mathrm{rand}() < \epsilon$ 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 (9b) and (9g) then
                Set reward value as $\mathbf{r}^i[t]$+= penalty;
            else if $\mathbf{a}^i[t]$ satisfies constraints (9b) and (9g) then
                Set reward value as $\mathbf{r}^i[t]$+= 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 $\mathbf{r}^i[t]+ =$ 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: $\mathcal{L}(\boldsymbol{\theta}^Q) = \mathbb{E}\big(y - Q(\mathbf{s}^i, \mathbf{a}^i; \boldsymbol{\theta}^Q)\big)^2$;
            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
```

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 vectors $\mathbf{s}^i[t]$ 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 $\boldsymbol{a}^i[t-1]$

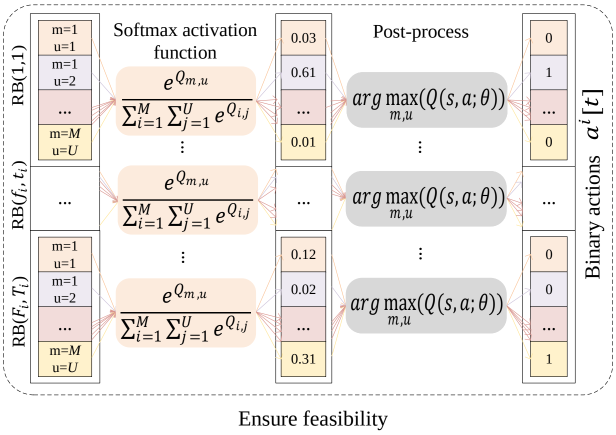

RB(1,1)
RB(f_i, t_i)
RB(F_i, T_i)
m=1 u=1
m=1 u=2
...
m=M u=U
Softmax activation function
$\frac{e^{Q_{m,u}}}{\sum_{i=1}^{M}\sum_{j=1}^{U} e^{Q_{i,j}}}$
0.03
0.61
0.01
0.12
0.02
0.31
Post-process
$arg\max_{m,u}(Q(s,a;\theta))$
0
1
0
0
0
1
Binary actions $\boldsymbol{a}^i[t]$
Ensure feasibility

Fig. 4. The customized activation function design for ensuring the feasibility of last layer D2QN outputs per agent.

and $\boldsymbol{e}^{\mathsf{x}}[t] = [e^{\mathsf{x}}_u[t]]^T$, *i.e.*,

$$\mathcal{S}^i|_{i=1} := \Big\{ \mathbf{s}^1[t] \Big| \mathbf{s}^1[t] = (\boldsymbol{\lambda}[t], \boldsymbol{\varphi}[t], \boldsymbol{q}[t-1], \boldsymbol{G}^1[t-1], \boldsymbol{a}^1[t-1], \boldsymbol{e}^{\mathsf{ur}}[t]) \Big\}$$

$$\mathcal{S}^i|_{i=2} := \Big\{ \mathbf{s}^2[t] \Big| \mathbf{s}^2[t] = (\boldsymbol{\lambda}[t], \boldsymbol{\varphi}[t], \boldsymbol{q}[t-1], \boldsymbol{G}^2[t-1], \boldsymbol{a}^2[t-1], \boldsymbol{e}^{\mathsf{em}}[t]) \Big\}. \quad (13)$$

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

$$\mathcal{A}^i := \Big\{ \mathbf{a}^i[t] \Big| \mathbf{a}^i[t] = \big[\pi^{f_i,t_i}_{m,u}[t]\big]^T; \pi^{f_i,t_i}_{m,u}[t] \in \{0,1\} \Big\}. \quad (14)$$

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. 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 belongs 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 (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.* (9b), (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". 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 $\omega\big(\frac{\sum_{u\in\mathcal{U}^{\mathsf{em}}} \mathcal{R}^{\mathsf{em}}_u(\boldsymbol{p}[t_i])}{R_0}\big) - (1-\omega)\big(\frac{\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\tau^{\mathsf{ur}}_u\}}{\tau_0}\big)$.

D. 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

$$\min_{\boldsymbol{p}} \sum_{u\in\mathcal{U}^{\mathsf{em}}} q_u[t_i] \quad (15a)$$

$$\text{s.t.} \quad (9c), (9e), (9f), \text{and} (9h). \quad (15b)$$

Problem (15) is inherently convex in $\boldsymbol{p}[t_i]$. The worst-case complexity of the interior-point method [47, Chapter 6] used to solve (15) is $\mathcal{O}\big(\sqrt{c}(v)^3\big)$, where $c = M_n U(F_1+F_2)+M_n U+2M_n$ and $v = M_n U(F_1+F_2)$ are the numbers of constraints and scalar variables, respectively. It is noted that all the constraints in (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.

VII. Performance Evaluation

A. 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. 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 frames traffic. The model has two hidden layers (fully connected), each includes 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] [26], 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 1 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

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2024.3396273

TABLE I
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

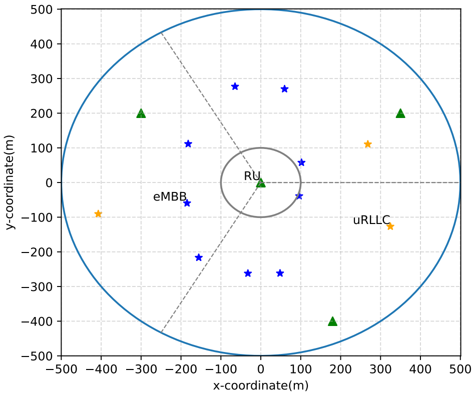


Fig. 5. The considered topology with $M = 4$ RUs, $U^{\text{ur}} = 3$ uRLLC users and $U^{\text{em}} = 9$ eMBB users.

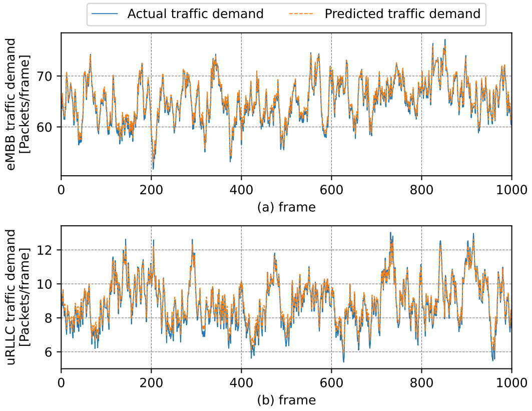


Fig. 6. The actual and predicted traffic demands via LSTM model for both eMBB and uRLLC services per frame.

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 I summarizes the main simulation parameters used in the experiments.

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 [32]. 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 1 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 1 to solve the problem 9 and demonstrate its superiority over existing approaches.

B. Numerical Results and Performance Comparison

The performance of the LSTM RNN model in predicting both eMBB and uRLLC traffic demands is illustrated in Fig. 6. Fig. 6(a) displays the predicted and actual values for one of the eMBB users, while Fig. 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. 6(a), the MSE value is measured to be 0.00232. Similarly, for the uRLLC users shown in Fig. 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. 7, we present a comprehensive visualization of the performance achieved by Algorithm 2 across different numerologies: Fig. 7(a) with the mixed numerologies in the frequency domain with slice awareness (SA), Fig. 7(b) with the mixed numerologies in the time domain with SI, and Fig. 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

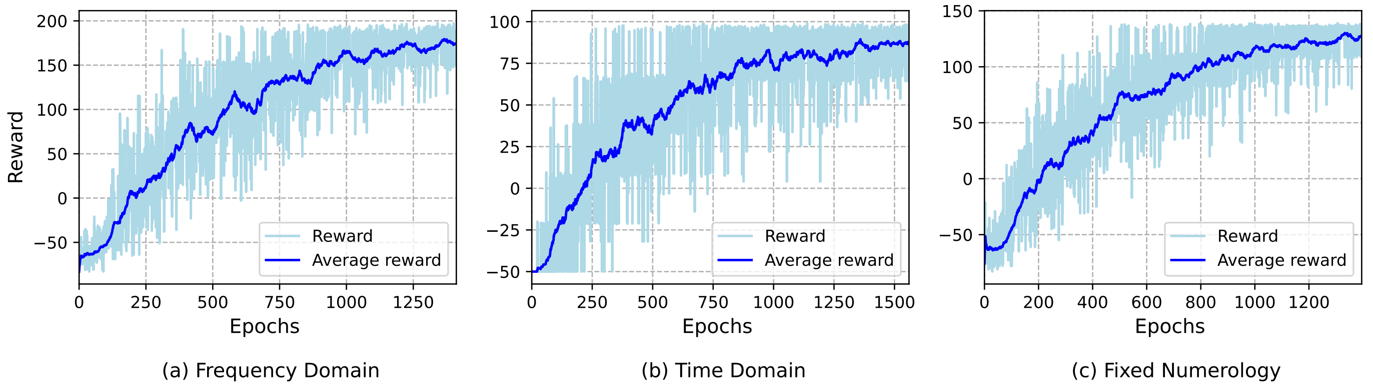


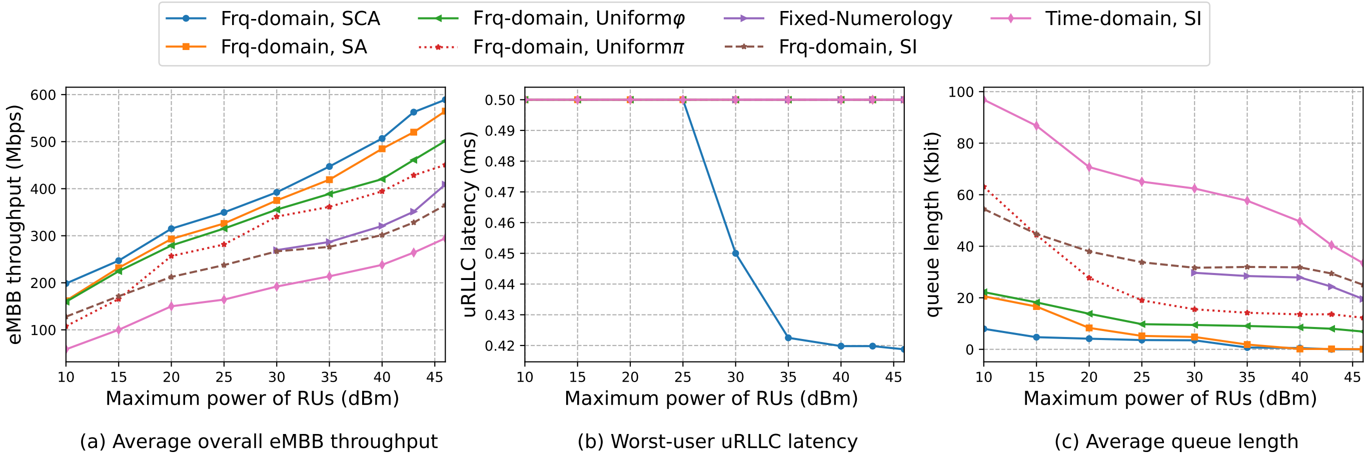
Fig. 7. The converge behavior of Algorithm 2 in different frequency-time grid numerologies: (a) mixed numerology in the frequency domain, (b) mixed numerology in the time domain, and (c) fixed numerology.

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 2 and the benefits of considering different numerologies and SA in optimizing the system performance.

In Fig. 8, we assess the performance of Algorithm 1 using various strategies compared to the above-mentioned benchmark schemes. To assess the eMBB throughput under various resource allocation schemes, Fig. 8(a) presents the total throughput 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 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 throughput 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 throughput 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. 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 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 benchmark schemes. Similarly, the fixed numerology scheme is infeasible when $P^{\max} < 30$ 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. 9 highlights the superiority of the SA scenario, especially in high-traffic conditions, compared to the SI approach. Upon closer examination of Fig. 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



(a) Average overall eMBB throughput

(b) Worst-user uRLLC latency

(c) Average queue length

Fig. 8. The performance comparison between Algorithm 1 and existing benchmark schemes versus maximum power budget of RU $P^{\max}$ in terms of (a) average overall eMBB throughput, (b) worst end-to-end uRLLC users, and (c) backlog queue length.

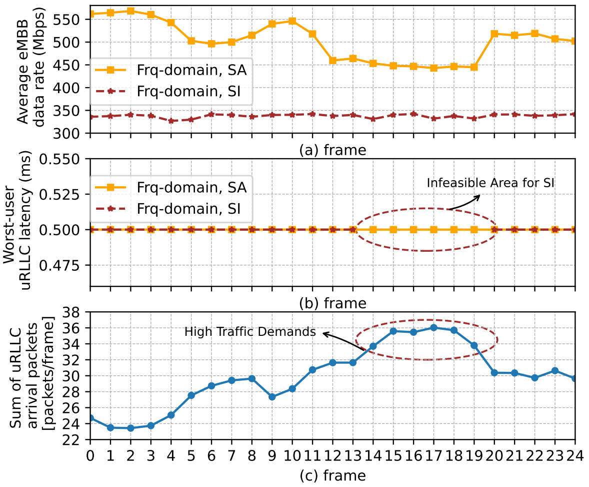


Fig. 9. Impact of the rate of uRLLC arrival packets/frame on the overall performance of eMBB and uRLLC in both SA and SI scenarios with $P^{\max} = 43$ dBm.

technique was shown to be infeasible during these frames. As we expected in Fig. 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. 9(a) plots the average eMBB throughput of Algorithm 1 with SA and SI in different frames. The figure proves the effectiveness of the SA scenario in achieving higher throughput 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. 9 confirm that SA is a beneficial technique in improving the eMBB throughput and effectively managing the demands of uRLLC users that exceed the allocated RBs in their dedicated slice.

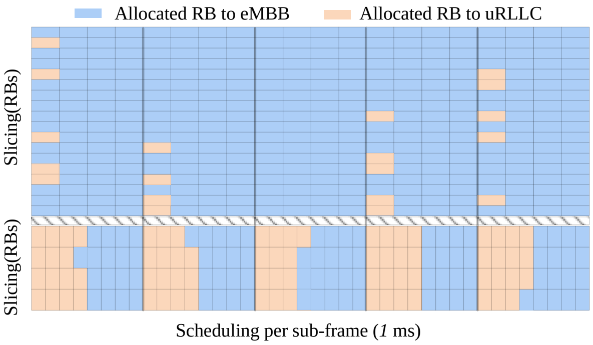


Fig. 10. Demonstration of RB scheduling to both eMBB and uRLLC services based on Algorithm 1 considering mixed numerology in the frequency domain. The vertical solid lines separate the sub-frame from the next sub-frame.

In Fig. 10, we provide a more detailed visualization of how the SA technique works in uRLLC preemption and its impact on improving eMBB throughput. The figure focuses on five sub-frames as an example, illustrating the RBs's 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. 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 eMBB throughput. 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.

VIII. Conclusion

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 variables 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 study 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/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 grouped RBs, thus alleviating computational burdens.

References

[1] 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.

[2] 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.

[3] 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)*, pp. 1–6, IEEE, 2020.

[4] 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.

[5] O-RAN Working Group 1, "O-RAN Use Cases Detailed Specification 6.0," Technical Specification O-RAN.WG1.Use-Cases-Detailed-Specification-v06.00, July 2021.

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

[7] M. Tayyab, X. Gelabert, and R. Jäntti, "A survey on handover management: From lte to nr," *IEEE Access*, vol. 7, pp. 118907–118930, 2019.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] R. C. Notes, "document 3gpp tsg ran wg1 meeting# 86," *Gothenburg, Sweden, Aug*, 2016.

[13] 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.

[14] 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.

[15] 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.

[16] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55765–55779, 2018.

[17] 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.

[18] 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.

[19] 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)*, pp. 1396–1400, IEEE, 2017.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

[24] 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*, pp. 1–10, IEEE, 2021.

[25] 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.

[26] 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. 45674–45688, 2020.

[27] 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.

[28] 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.

[29] 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.

[30] 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)*, pp. 597–602, IEEE, 2019.

[31] 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)*, pp. 1–6, IEEE, 2019.

[32] 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.

[33] 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.

[34] 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.

[35] 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)*, pp. 245–250, IEEE, 2019.

[36] 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. 65811–65820, 2019.

[37] 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*, 2023.

[38] L. Bonati, S. D’Oro, M. Polese, S. Basagni, and T. Melodia, “Intelligence and learning in O-RAN for data-driven NextG cellular networks,” *IEEE Communications Magazine*, vol. 59, no. 10, pp. 21–27, 2021.

[39] S. Niknam, A. Roy, H. Dhillon, S. Singh, R. Banerji, J. Reed, N. Saxena, and S. Yoon, “Intelligent O-RAN for beyond 5G and 6G wireless networks. arXiv 2020,” *arXiv preprint arXiv:2005.08374*.

[40] 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 oran: A multi-layer optimization framework,” *arXiv preprint arXiv:2302.02711*, 2023.

[41] O. Alliance, “O-RAN: Towards an open and smart RAN,” *[Online]*, 2018. Avaiable: https://www.o-ran.org/resources.

[42] O-RAN.WG2.Use-Case-Requirements-v02.01, “Non-RT RIC & A1 interface: Use cases and requirements,” *Technical Specification*, Nov. 2021.

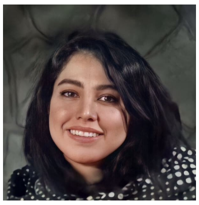
[43] S. Niknam *et al.*, “Intelligent O-RAN for beyond 5G and 6G wireless networks,” 2020. [Online]:. https://arxiv.org/abs/2005.08374.

[44] A. B. Kihero, M. S. J. Solaija, and H. Arslan, “Inter-numerology interference for beyond 5G,” *IEEE Access*, vol. 7, pp. 146512–146523, 2019.

[45] 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.*, pp. 13–22, 2015.

[46] 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.

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



Fatemeh Kavehmadavani received her B.S. degree in Electrical and Communication Engineering from Yazd University, Iran, in 2014, and her M.Sc. degree in Telecommunication Systems from Shahid Bahonar University of Kerman, Iran, in 2017. She is currently a Ph.D. student at the research group SIGCOM in the Interdisciplinary Centre for Security, Reliability, and Trust (SnT) at the University of Luxembourg. Her research interests include wireless communications for 6G and IoT, focusing on open radio access networks (Open RAN), xRAN architectures merged with artificial intelligence (AI) and machine learning (ML), orchestration and monitoring, intelligent traffic steering (TS) and management, dynamic traffic prediction, and radio resource management (RRM).

Van-Dinh Nguyen (Senior Member, IEEE) has been an Assistant Professor with VinUniversity, Vietnam since September 2022. He was a Research Associate with SnT-University of Luxembourg, a Post-Doctoral Researcher and a Lecturer with Soongsil University, a Post-Doctoral Visiting Scholar with the University of Technology Sydney, and a Ph.D. Visiting Scholar with Queen’s University Belfast. He has authored or co-authored over 90 papers published in international journals and conference proceedings. His current research activity is focused on Open RAN, wireless sensing, edge/fog computing, and AI/ML solutions for wireless communications. He received four best conference paper awards and four Exemplary Editor Awards from IEEE COMMUNICATIONS LETTERS and IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY. He has served as a reviewer for many top-tier international journals on wireless communications and a technical program committee member for several flagship international conferences in related fields. He is an Editor of the IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY and IEEE SYSTEMS JOURNAL and a Senior Editor of IEEE COMMUNICATIONS LETTERS.

Thang X. Vu (Senior Member, IEEE) received the B.S. and the M.Sc., both in Electronics and Telecommunications Engineering, from the VNU University of Engineering and Technology, Vietnam, in 2007 and 2009, respectively, and the Ph.D. in Electrical Engineering from the University Paris-Sud, France, in 2014. In 2010, he received the Allocation de Recherche fellowship to study Ph.D. in France. From July 2014 to January 2016, he was a postdoctoral researcher with the Singapore University of Technology and Design (SUTD), Singapore. Currently, he is a research scientist at the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg. His research interest includes wireless communications, with particular interests of applications of optimization and machine learning on design and analyze the multi-layer 6G networks. He has successfully acquired, as the PI and vice PI, several Luxembourg national and ESA projects with a total funding of 2.6 MEURs. He was a recipient of the SigTelCom 2019 Best Paper Award. He is also serving as an Associate Editor for the IEEE Communications Letters.

SYMEON CHATZINOTAS (Fellow, IEEE) received the M.Eng. degree in telecommunications from Aristotle University of Thessaloniki, Greece, in 2003, and the M.Sc. and Ph.D. degrees in electronic engineering from the University of Surrey, U.K., in 2006 and 2009, respectively. He is currently a Full Professor/Chief Scientist I and the Head of the Research Group SIGCOM, Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg. In the past, he has lectured as a Visiting Professor with the University of Parma, Italy, and contributed in numerous research and development projects for the Institute of Informatics and Telecommunications, National Center for Scientific Research “Demokritos,” the Institute of Telematics and Informatics, Center of Research and Technology Hellas, and Mobile Communications Research Group, Center of Communication Systems Research, University of Surrey. He has authored more than 700 technical papers in refereed international journals, conferences, and scientific books. Prof. Chatzinotas received numerous awards and recognitions, including the IEEE Fellowship and an IEEE Distinguished Contributions Award. He is currently in the editorial board of the IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE OPEN JOURNAL OF VEHICULAR TECHNOLOGY, and the International Journal of Satellite Communications and Networking.