Intelligent User Association and Resource Scheduling in Open RAN with 7.2x Functional Split

Fatemeh Kavehmadavani, Thang X. Vu, and Symeon Chatzinotas
Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg
Email: {fatemeh.kavehmadavani, thang.vu, symeon.chatzinotas}@uni.lu

***Abstract*—Open Radio Access Network (RAN), with its open and disaggregated architecture, fosters innovation in traffic and congestion control in dynamic environments. However, achieving optimal user association and resource scheduling under incomplete information and varying traffic patterns remains challenging due to non-convexity and combinatorial aspects. To address this, we propose a hierarchical approach that features a heuristic, iterative successive convex approximation (SCA), and deep reinforcement learning (DRL) algorithm. The proposed solution considers a practical constraint on limited information exchange among radio units (RUs) and complies with the O-RAN Alliance's 7.2x functional split (FS) option. This scheme optimizes performance through intelligent user association, resource scheduling, and congestion control. The simulation results highlight its superiority over the benchmark schemes, confirming its effectiveness and demonstrating a throughput improvement of 106.27% compared to the benchmark scheme.**

I. Introduction

Traditional radio access network (RAN) architectures, due to their rigid nature, struggle to serve diverse services with heterogeneous demands such as enhanced mobile broadband (eMBB) and ultra-reliable low-latency communication (uRLLC), leading to suboptimal performance. The Open RAN, as a promising architecture, can manage traffic and control congestion for different services, making it a key innovation in the *NextG* networks [1]. Open RAN uses a concept called functional splitting (FS) to break traditional BSs into more manageable units, such as radio units (RUs), distributed units (DUs), and central units (CUs) per 3GPP standards. While Open RAN goes beyond that, it also integrates near-real-time (near-RT) and non-real-time (non-RT) RAN intelligent controller (RIC) modules at the management and control layers for automated control and feedback through open interfaces.This powerful combination enables network optimization, machine learning (ML) solutions, and, ultimately, an intelligent and adaptive RAN layer [2].

Although providing an intelligent traffic management framework in dynamic networks, including intelligent user association (UA) and resource scheduling, is crucial, there are still some challenges in obtaining an optimal strategy for diverse services. Hence, this study proposes a hierarchical distributed optimization method to address non-convex and combinatorial characteristics and limited information exchanged in a multi-traffic scenario. Additionally, this paper benefits from network slicing (NS) technology and the slice-awareness concept to improve data rates for eMBB services while reducing latency for uRLLC services [3]. To enhance the overall network performance, this research also explores the integration of mixed numerologies in the frequency domain, benefiting from the mini-slot concept to efficiently support uRLLC services.

Recent research has optimized ML-based UA and resource scheduling schemes. Traditional RANs have utilized actor-critic reinforcement learning (RL) to make intelligent user access decisions in [4] and [5]. To address global network information and frequent handovers, [6] has proposed a deep reinforcement learning (DRL) model. However, these efforts lack individual performance enhancement and require nearly complete information, which may not be practically feasible. Therefore, there is a need for intelligent UA and resource scheduling in the flexible Open RAN context. While only a few studies have examined UA schemes in Open RAN, for instance, [7] has proposed a DRL-based scheme to optimize access decisions, reduce handovers in massive BS deployments, and maximize throughput. However, none of the above-mentioned works investigated an intelligent UA and resource management for multi-traffic downlink Open RAN systems. To realize fully automated networks with enhanced control and optimization capabilities, the development of a DRL framework is essential. Therefore, in this study, an intelligent UA and resource scheduling within an Open RAN architecture aligned with 7.2x FS is investigated to optimize the utility function comprising both eMBB and uRLLC services. In summary, our key contributions are as follows:

- An multi-traffic scenario within an Open RAN architecture with 7.2x FS is introduced in this paper, utilizing RAN slicing, mini-slot-based frame structure, and mixed numerology multiplexing to prevent resource waste with a slice-aware approach. Our aim is to minimize the eMBB queue length and uRLLC latency while addressing congestion, power budgets, slice awareness, and the services' requirements. This framework also considers dynamic traffic patterns during the time-frame period.
- Regarding the 7.2x FS, in which different RAN nodes manage scheduling and power allocation, we design a hierarchical optimization scheme in a distributed architecture. This scheme includes heuristic, iterative successive convex approximation (SCA), and DRL-based algorithms to optimize the UA and resource allocation, addressing network congestion. While the majority of existing works assumes perfect channel state information (CSI), the proposed algorithm does not require interference channels

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).

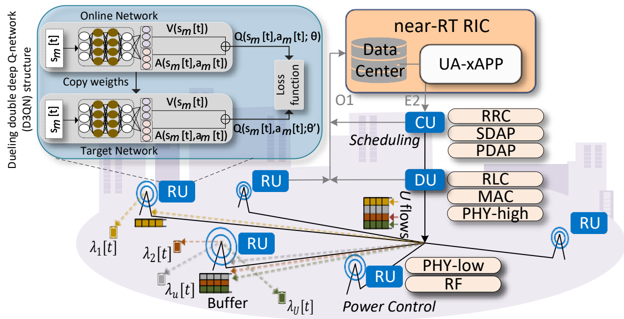

Fig. 1: High-level structure of deploying the proposed intelligent UA and scheduling considering 7.2x FS in Open RAN architecture.

for optimization.

- The numerical results demonstrate that our approach achieves results surpassing both centralized orthogonal frequency division multiple access (OFDMA) and uniform power schemes.

II. System Model

As depicted in Fig. 1, we consider an Open RAN-based downlink cellular network composed of a CU, a DU, and a set of $M = |\mathcal{M}|$ multi-antenna (K_{tx}) RUs serving a set of $U = |\mathcal{U}|$ single-antenna user equipment (UE), including $U^{\text{ur}} = |\mathcal{U}^{\text{ur}}|$ uRLLC and $U^{\text{em}} = |\mathcal{U}^{\text{em}}|$ eMBB UEs. This paper assumes two aware slices, which serve the eMBB and uRLLC services, respectively. Let $\lambda_u^{\times}[t]$ [packets/frame] denote the traffic demand of a generic UE u in the discrete time frame indexed by $t \in \{1, \ldots, T\}$, where $\times \in \{\text{ur}, \text{em}\}$. We assume that $\lambda_u^{\times}[t]$ follows the 3GPP FTP3 model [8] with the mean arrival rate $\mathbb{E}\{\lambda_u^{\times}[t]\} = \bar{\lambda}_u^{\times}$ ($\bar{\lambda}^{\text{em}} >> \bar{\lambda}^{\text{ur}}$), wherein the size of each packet is identical and equal to $Z^{\times}$ [Byte] ($Z^{\text{em}} >> Z^{\text{ur}}$).

We consider a distinct numerology per slice, indexed as $i = 1, 2$ for the eMBB and uRLLC services, respectively [9]. The system bandwidth (BW) B is divided into two independent parts by the split variable $\alpha \in (0, 1)$. This makes two slices with BW of $B_1 = (1-\alpha)B$ and $B_2 = \alpha B - B_G$ per slice including $F_i = \lfloor B_i/\beta_i \rfloor, i = 1, 2$, RBs, where the guard band $B_G = 180$ kHz. Let β_i be the BW of each RB, while $f_i = \{1, \ldots, F_i\}$ indicates the number of subcarriers per each transmission time interval (TTI). There are $T_i = \Delta/\delta_i$ TTIs indexed as t_i in the i-th numerology, where Δ and δ_i denote the frame and TTI duration, respectively.

Achievable Downlink Data Rate: Based on the Shannon-Hartley theorem, the downlink data rate [bits/s] of the u-th eMBB UE served by RU m at TTI t_i is given as

$$R_{m,u}^{\text{em}}(\boldsymbol{\varphi}, \boldsymbol{p}, \boldsymbol{\pi}) = \varphi_{m,u} \sum_{f_i=1}^{F_i} \beta_i \log_2 \Big(1 + \frac{\pi_{m,u}^{f_i,t_i} p_{m,u}^{f_i,t_i} g_{m,u}^{f_i,t_i}}{N_0 + I_{m,u}^{f_i,t_i}}\Big), \quad (1)$$

where N_0, $\pi_{m,u}^{f_i,t_i} \in \{0,1\}$, $p_{m,u}^{f_i,t_i}$ and $I_{m,u}^{f_i,t_i}$ are the noise power, the binary scheduling variable indicates whether RB(f_i, t_i) is assigned to u-th UE through RU m, the transmission power from RU m to UE u in RB(f_i, t_i), and the aggregated interference from other RUs on RB(f_i, t_i), respectively. While we model the quasi-static channel vector between RU m and UE u on all TTIs within one frame t as $\boldsymbol{h}_{m,u}^{f_i}$. Let us denote by $\boldsymbol{G}[t] \triangleq [g_{m,u}^{f_i}]; \forall m, u, f_i$ the channel gain between RBs and all UEs in frame t, where $g_{m,u}^{f_i,t_i} = g_{m,u}^{f_i} \triangleq \|\boldsymbol{h}_{m,u}^{f_i}\|_2^2; \forall t_i = \{1, \ldots, T_i\}$ is the effective channel gain. Meanwhile, $\varphi_{m,u} \in \{0,1\}$ denotes the binary variable UA, in which, if $\varphi_{m,u} = 1$, RU m is selected to transmit data of u-th data-flow; otherwise, $\varphi_{m,u} = 0$ which means single connectivity. To this end, $\boldsymbol{\varphi} \triangleq \big[\boldsymbol{\varphi}_u\big]^T; \forall u$, while $\boldsymbol{\varphi}_u \triangleq \Big[\varphi_{m,u}\Big]^T; \forall u$ is the binary vector UA of u-th UE, where $\sum_{m \in \mathcal{M}} \varphi_{m,u} \leq 1; \forall u$. Hence, the global UA decision is formulated as

$$\boldsymbol{\Psi} \triangleq \{\boldsymbol{\varphi}_{m,u} \in \{0,1\} | \sum_{m=1}^{M} \varphi_{m,u} \leq 1\}. \quad (2)$$

The sum of transmitted powers assigned to all scheduled RBs of each RU per each TTI must satisfy $\sum_{i,f_i,u} p_{m,u}^{f_i,t_i} \leq P_m^{\text{max}}, \forall m, t_i$. Combining the above constraints, the feasible set for power associated with both services (*i.e.*, eMBB and uRLLC) is defined as

$$\mathscr{P}[t] = \Big\{\boldsymbol{p}_{m,u}[t] \Big| p_{m,u}^{f_i,t_i} \geq 0, \sum_{i,f_i,u} p_{m,u}^{f_i,t_i} \leq P_m^{\text{max}}\Big\}, \quad (3)$$

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

Due to the band guard, there is no inter-slice interference, while there is only interference from other RUs that are using the same RB (intra-slice) as $I_{m,u}^{f_i,t_i} = \sum_{m' \in \mathcal{M} \setminus \{m\}} \sum_{u' \in \mathcal{U}_{m'}} \pi_{m',u'}^{f_i,t_i} p_{m',u'}^{f_i,t_i} g_{m',u}^{f_i,t_i}$, where $\mathcal{U}_{m'}$ is the set of UEs that are served by RU m', *i.e.*, $\varphi_{m',u'} = 1; \forall u', m'$.

By design, the UEs served by one RU occupy orthogonal RBs, *i.e.*, $\sum_u \pi_{m,u}^{f_i,t_i} \leq 1; \forall m, t_i, f_i$. Furthermore, each scheduled uRLLC UE should receive a minimum number of RBs from the dedicated uRLLC slice to vacate the arrival packets in the queue in the specific duration of time D^{ur} after the arrival time τ_u^{arv}, which is expressed as $\sum_{t_2=\tau_u^{\text{arv}}/\delta_2}^{(D^{\text{ur}}+\tau_u^{\text{arv}})/\delta_2} \sum_{f_2=1}^{F_2} \pi_{m,u}^{f_2,t_2} \geq \varepsilon_u^{\text{ur}}[t]; \forall t, u \in \mathcal{U}^{\text{ur}}$; $\varepsilon_u^{\text{ur}}[t] \triangleq \Big\lceil \frac{\lambda_u^{\text{ur}}[t] Z^{\text{ur}}}{\rho.\kappa} \Big\rceil$, where ρ and κ represent the number of symbols per RB and the number of bits per symbol. Based on the modulation and coding scheme (MCS) selected from [10], the signal-to-interference-plus-noise ratio (SINR) of the uRLLC service in error probability (P_e) must be greater than Γ^{th} dB as

$$\Gamma_{m,u}^{f_i,t_i} \triangleq \frac{\pi_{m,u}^{f_i,t_i} p_{m,u}^{f_i,t_i} g_{m,u}^{f_i,t_i}}{N_0 + I_{m,u}^{f_i,t_i}} \geq \pi_{m,u}^{f_i,t_i} \Gamma^{\text{th}}; \forall f_i, t_i, m, u \in \mathcal{U}^{\text{ur}}. \quad (4)$$

To improve resource usage, a slice-aware approach is deployed to dynamically allocate RBs between two slices. Unlike isolated allocation, this allows traffic to be assigned to the most suitable numerology while still respecting the slice constraints. With slice awareness, an increased eMBB demand can borrow resources from the uRLLC slice (and vice versa), enhancing overall throughput. Although being more complex, this dynamic approach optimizes resource utilization based on real-time traffic. To respond to the priority of the scheduled uRLLC service, we consider $\sum_{t_1=\tau_u^{\text{arv}}/\delta_1}^{(D^{\text{ur}}+\tau_u^{\text{arv}})/\delta_1} \sum_{f_1=1}^{F_1} \pi_{m,u}^{f_1,t_1} \geq e_u^{\text{ur}}[t], \ \forall t, u \in \mathcal{U}^{\text{ur}}$, where $e_u^{\text{ur}}[t] = \lfloor (\varepsilon_u^{\text{ur}}[t] - \Omega_u[t])^+ \rfloor$, $\Omega_u[t] \triangleq (\varepsilon_u^{\text{ur}}[t] / \sum_{u \in \mathcal{U}^{\text{ur}}} \varepsilon_u^{\text{ur}}[t]) \times (F_2 \times D^{\text{ur}}/\delta_2)$, and $(x)^+ \triangleq \max\{x, 0\}$. In contrast, the following constraint allocates all the underused RBs of the other slices to the eMBB users to increase throughput as $\sum_{t_2} \sum_{f_2} \pi_{m,u}^{f_2,t_2} \geq e_u^{\text{em}}[t], \ \forall t, u \in \mathcal{U}^{\text{em}}$, where $e_u^{\text{em}}[t] = (\lfloor (F_2 \times T_2 - \sum_{u^{\text{ur}}} \min(\varepsilon_u^{\text{ur}}[t], \Omega_u[t])) / U^{\text{em}} \rfloor)^+$ is the slice awareness for mixed numerologies in the frequency domain. Herein, $\min(\varepsilon_u^{\text{ur}}[t], \Omega_u[t])$ represents the number of RBs scheduled for the u-th uRLLC user in the t-th frame.

Collecting all the constraints related to the RB scheduling variables from the mentioned scheduling and slice-awareness constraints makes the global scheduling vector as

$$\mathbf{\Lambda}[t] = \Big\{\boldsymbol{\pi}_{m,u}[t],\, \forall m, u \Big| \pi^{f_i,t_i}_{m,u} \in \{0,1\}, \sum_{u=1}^{U} \pi^{f_i,t_i}_{m,u} \le 1,$$
$$\sum_{t_2=\tau^{\mathsf{arv}}_u/\delta_2}^{(D^{\mathsf{ur}}+\tau^{\mathsf{arv}}_u)/\delta_2} \sum_{f_2=1}^{F_2} \pi^{f_2,t_2}_{m,u} \ge \varepsilon^{\mathsf{ur}}_u[t]|_{\forall u\in\mathcal{U}^{\mathsf{ur}}}, \sum_{t_1=\tau^{\mathsf{arv}}_u/\delta_1}^{(D^{\mathsf{ur}}+\tau^{\mathsf{arv}}_u)/\delta_1} \sum_{f_1=1}^{F_1} \pi^{f_1,t_1}_{m,u}$$
$$\ge e^{\mathsf{ur}}_u[t]|_{\forall u\in\mathcal{U}^{\mathsf{ur}}}, \sum_{t_2=1}^{T_2} \sum_{f_2=1}^{F_2} \pi^{f_2,t_2}_{m,u} \ge e^{\mathsf{em}}_u[t]|_{\forall u\in\mathcal{U}^{\mathsf{em}}} \Big\}, \tag{5}$$

where $\boldsymbol{\pi}_{m,u}[t] \triangleq \left[\pi^{f_i,t_i}_{m,u}\right]^T$ is denoted as the RBs matrix assigned to the u-th generic UE at the m-th RU in frame t. This study defines the RB matrix $\boldsymbol{\pi}[t] \triangleq \left[\boldsymbol{\pi}_{m,u}[t]\right]^T; \forall m, u$ in frame t. Let us define $\boldsymbol{\pi} \triangleq [\boldsymbol{\pi}[t]]$.

It should be noted that the processing queue follows the $M/M/1$ model, serving packets on a first-come-first-serve basis. The queue length [bits] of the generic data flow u that is waiting in the buffer of m-th RU in t_i-th TTI to be transmitted is calculated as $q^{\mathsf{x}}_{m,u}[t_i] = \left(q^{\mathsf{x}}_{m,u}[t_i-1] - R^{\mathsf{x}}_{m,u}(\boldsymbol{\varphi},\boldsymbol{p},\boldsymbol{\pi})\delta_i\right)^+$. The queue length of all packets waiting to be served by RU m at time-frame t is written as $q^{\mathsf{x}}_m[t] = \sum_u \big(\varphi_{m,u}\lambda^{\mathsf{x}}_u[t]Z^{\mathsf{x}}\Delta + \sum_{t_i} q^{\mathsf{x}}_{m,u}[t_i]\big)$, where $\varphi_{m,u}\lambda^{\mathsf{x}}_u[t]Z^{\mathsf{x}}$ [bits/frame] is the flow of UE u at RU m. Given that eMBB has larger packet sizes than uRLLC, we focus on eMBB queue length for congestion control. To prevent frequent handovers and ensure stability, using the average queue length over W lag frames, ($\bar{q}^{\mathsf{em}}_m \triangleq \frac{1}{W}\sum_{t-W}^{t} q^{\mathsf{em}}_m[t]$) as a congestion control constraint is effective.

One of the goals of this work is to minimize the uRLLC latency $\tau^{\mathsf{ur}}_u(\boldsymbol{\pi})$, considering only the transmission latency from RU to UE while neglecting constant delays that do not affect the proposed optimization problem (see [1] for details). The transmission latency, as the gap (measured in TTIs) between the packet arrival time to the queue τ^{arv}_u and the time it is served, is denoted as $\tau^{\mathsf{tx}}_u = \delta_i \arg\max_{t_i}\{\pi^{f_i,t_i}_{m,u}[t_i]\}; \forall u \in \mathcal{U}^{\mathsf{ur}}$. Thanks to the mini-slot concept, higher-priority uRLLC packets of small size are served immediately, unaffected by eMBB traffic due to separate slices. To ensure a minimum latency requirement for the u-th uRLLC UE, the latency is bound by a predefined threshold D^{ur}_u as

$$\tau^{\mathsf{arv}}_u \le \tau^{\mathsf{ur}}_u(\boldsymbol{\pi}) \approx \max(\tau^{\mathsf{tx}}_u, \tau^{\mathsf{arv}}_u) \le D^{\mathsf{ur}} + \tau^{\mathsf{arv}}_u. \tag{6}$$

A. Problem Formulation

Shorter queues lead to higher eMBB throughput, crucial to controlling congestion, particularly for eMBB traffic with large packets. This work aims to minimize the eMBB queue length and the worst-case uRLLC latency. In particular, the utility function is given as $\omega\sum_{t_i,m,u\in\mathcal{U}^{\mathsf{em}}} \frac{q^{\mathsf{em}}_{m,u}[t_i]}{q_0} + (1-\omega)\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\frac{\tau^{\mathsf{ur}}_u}{\tau_0}\}$, where $\omega \in [0,1]$ denotes the regulatory factor to control the influence of queue length and latency. Besides, $q_0 > 0$ and $\tau_0 > 0$ are the reference for the eMBB queue length and the uRLLC latency, respectively, to balance the two different dimensions of the two quantities. Consequently, intelligent UA decisions, congestion control, and scheduling optimization problem are mathematically formulated as

$$\min_{\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p}} \quad \omega \sum_{t_i,m,u\in\mathcal{U}^{\mathsf{em}}} \frac{q^{\mathsf{em}}_{m,u}[t_i]}{q_0} + (1-\omega)\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\frac{\tau^{\mathsf{ur}}_u}{\tau_0}\} \tag{7a}$$
$$\text{s.t.} \quad R^{\mathsf{em}}_{m,u}(\boldsymbol{\varphi},\boldsymbol{p},\boldsymbol{\pi}) \ge R^{\min}; \quad \forall m, u \in \mathcal{U}^{\mathsf{em}} \tag{7b}$$
$$\bar{q}^{\mathsf{em}}_m \le q^{\max}_m; \quad \forall m, t \tag{7c}$$
$$(2), (3), (4), (5), \text{and}(6),$$

where the constraint (7b) satisfies the QoS of eMBB in which each scheduled eMBB UE must receive a minimum data rate ($R^{\min}$) per frame t. The constraint (7c) ensures that the m-th queue buffer should not exceed the maximum buffer size ($q^{\max}_m$) while minimizing the packet loss from buffer overflow. **Challenges of Solving Problem (7):** The main challenge in solving the proposed problem is the inability to optimize $\boldsymbol{p}$ and $\boldsymbol{\pi}$ jointly due to 7.2x FS [11]. Accordingly, power control happens in RUs (lower layer), while resource scheduling is managed in CUs (higher layer). However, optimizing UA variables requires a holistic network-wide view with prior knowledge of traffic demands, queue length, and network conditions. On top of these, the proposed optimization problem is categorized as a mixed-integer non-convex program due to the binary nature of $\boldsymbol{\pi}$ and $\boldsymbol{\varphi}$, along with the coupling of $\varphi_{m,u}.\pi^{f_i,t_i}_{m,u}.p^{f_i,t_i}_{m,u}$ in the objective function and some constraints, making it challenging to solve jointly. Hence, due to incomplete information, such as queue length, varying traffic demands, network conditions, and interference in such a dynamic environment, in this paper a distributed optimization method is developed to reduce computational complexity.

III. Proposed Methodology

This section proposes an efficient distributed optimization algorithm for the problem (7), implemented in three phases across Open RAN components.

A. UA-xAPP

Heuristic methods leverage experience and data for informed decisions in decentralized networks. They are valuable for periodic network optimization based on past data, suggesting adjustments to improve performance. The proposed UA algorithm is robust against time-varying traffic, UE locations, and channel conditions. To reduce the number of unnecessary handovers in the single-connectivity technique, it considers the average RB utilization rate instead of immediate gains. The network parameters are continuously monitored, triggering alarms for potential congestion. Steps 2-10 of the algorithm (1) manage congestion by identifying overloaded RUs and UEs with low data rates. It monitors network parameters, triggering a congestion flag when UE buffers exceed a threshold q^{th} and RU's RBs utilization is high. To alleviate congestion, a scoring mechanism prioritizes RUs based on proximity to UEs with large backlog buffers and available buffer capacities.

B. RB Scheduling

As mentioned, due to the unknown inter-RU interference and the assumed 7.2x FS, two variables $\boldsymbol{\pi}$ and $\boldsymbol{p}$ cannot be solved jointly. After execution of $\boldsymbol{\varphi}^*$ on the RAN nodes through the E2 interface, RB scheduling is optimized at CU in a centralized way thanks to the closed control loop between CU and near-RT RIC. In this method, the large-scale channel is used to calculate the interference, assuming equal power allocation. Since minimizing the eMBB queue length is equivalent to maximizing the eMBB data rate, we employ $\omega\sum_{m,u\in\mathcal{U}^{\mathsf{em}}}\{\frac{R^{\mathsf{em}}_{m,u}[t]}{R_0}\} - (1-\omega)\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\frac{\tau^{\mathsf{ur}}_u[t]}{\tau_0}\}$ as the objective function in the optimization problem of RB scheduling. However, solving the RB allocation problem with standard solvers is challenging due to the binary nature of the variables. To overcome this, we relax the binary variables to continuous ones, replacing $\pi_{m,u}^{f_i,t_i} \in [0,1]$ with $\pi_{m,u}^{f_i,t_i} \in \{0,1\}$ in (5), allowing $\tilde{\Lambda}[t]$ to include relaxed rather than binary variables.

To expedite the convergence of the iterative algorithm, we introduce the penalty function $\mathcal{P}(\boldsymbol{\pi}) = \sum_{m,u,f_i,t_i}\big[(\pi_{m,u}^{f_i,t_i})^2 - \pi_{m,u}^{f_i,t_i}\big]$, which is convex in $\boldsymbol{\pi}$. In particular, $\mathcal{P}(\boldsymbol{\pi}) \leq 0$ for any $\pi_{m,u}^{f_i,t_i} \in [0,1]$, serving to penalize relaxed variables and approximate binary solutions at optimum (*i.e.*, satisfying (5)). Furthermore, we introduce the slack variables $\boldsymbol{z} = \big[z_{m,u}^{f_i,t_i}\big]^T$ to convert the first term of the objective function into an achievable one. According to the above-mentioned penalty function and auxiliary variables in the objective function, the parameterized relaxed problem is expressed as

$$\max_{\boldsymbol{\pi},\boldsymbol{z}} \quad \Xi \triangleq \Big(\omega \min_{m,u\in\mathcal{U}^{\mathsf{em}}}\{\sum_{f_i,t_i} z_{m,u}^{f_i,t_i}\} - (1-\omega)\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\frac{\tau_u^{\mathsf{ur}}[t]}{\tau_0}\} + \nu\mathcal{P}(\boldsymbol{\pi})\Big) \tag{8a}$$

$$\text{s.t.} \quad \tilde{\Lambda}[t] \tag{8b}$$

$$\log_2\Big(1+\frac{\pi_{m,u}^{f_i,t_i}p_{m,u}^{f_i,t_i}g_{m,u}^{f_i,t_i}}{N_0+I_{m,u}^{f_i,t_i}}\Big) \geq z_{m,u}^{f_i,t_i}; \forall u \in \mathcal{U}^{\mathsf{em}} \tag{8c}$$

$$(4), (6), (7b), \text{and} (7c),$$

where $\nu > 0$ is a given penalty parameter. In problem (8), the objective function is non-concave due to $\mathcal{P}(\boldsymbol{\pi})$, while constraints (8c), (7b), and (7c) are non-convex. Using first-order Taylor, $\mathcal{P}(\boldsymbol{\pi})$ is approximated to $\mathcal{P}^{(j)}(\boldsymbol{\pi}) \triangleq \sum_{m,u,f_i,t_i}\big[\pi_{m,u}^{f_i,t_i}(2(\pi_{m,u}^{f_i,t_i})^{(j)}-1) - ((\pi_{m,u}^{f_i,t_i})^{(j)})^2\big]$ at the j-th iteration. To deal with the non-convexity of the constraint (8c), it is rewritten as $\log_2\Big(N_0 + I_{m,u}^{f_i,t_i} + \pi_{m,u}^{f_i,t_i}p_{m,u}^{f_i,t_i}g_{m,u}^{f_i,t_i}\Big) \geq z_{m,u}^{f_i,t_i} + \log_2\big(N_0 + I_{m,u}^{f_i,t_i}\big)$, where its second term of RHS still makes it non-convex, to handle this, we approximate $f(\pi_{m,u}^{f_i,t_i}) \triangleq \log_2(N_0+I_{m,u}^{f_i,t_i})$ at the j-th iteration via first-order Taylor series as $f^{(j)}(\pi_{m,u}^{f_i,t_i},(\pi_{m,u}^{f_i,t_i})^{(j)}) \triangleq f((\pi_{m,u}^{f_i,t_i})^{(j)}) + \frac{(\pi_{m,u}^{f_i,t_i}-(\pi_{m,u}^{f_i,t_i})^{(j)})}{\ln 2(N_0+(I_{m,u}^{f_i,t_i})^{(j)})} \times \big(\sum_{m'\in\mathcal{M}_n\setminus\{m\}}\sum_{u'\in\mathcal{U}_{m'}} p_{m',u'}^{f_i,t_i}g_{m',u}^{f_i,t_i}\big)$, in which $(I_{m,u}^{f_i,t_i})^{(j)} \triangleq I_{m,u}^{f_i,t_i}((\pi_{m,u}^{f_i,t_i})^{(j)})$. Hence, (8c) is approximated as

$$\log_2\Big(N_0 + I_{m,u}^{f_i,t_i} + \pi_{m,u}^{f_i,t_i}p_{m,u}^{f_i,t_i}g_{m,u}^{f_i,t_i}\Big) \geq z_{m,u}^{f_i,t_i} + f^{(j)}(\pi_{m,u}^{f_i,t_i},(\pi_{m,u}^{f_i,t_i})^{(j)}) \tag{9}$$

Similarly, we utilize the same method to address the non-convexity in constraints (7b) and (7c), respectively as

$$\sum_{m,f_i,t_i}\varphi_{m,u}^*\beta_i\log_2\Big(N_0 + I_{m,u}^{f_i,t_i} + \pi_{m,u}^{f_i,t_i}p_{m,u}^{f_i,t_i}g_{m,u}^{f_i,t_i}\Big) \geq R^{\mathsf{min}} + \sum_{m,f_i,t_i}\varphi_{m,u}^*\beta_i f^{(j)}(\pi_{m,u}^{f_i,t_i},(\pi_{m,u}^{f_i,t_i})^{(j)}), \tag{10}$$

and,

$$\sum_{u}\varphi_{m,u}^*\sum_{f_i,t_i}\beta_i\log_2\Big(N_0 + I_{m,u}^{f_i,t_i} + \pi_{m,u}^{f_i,t_i}p_{m,u}^{f_i,t_i}g_{m,u}^{f_i,t_i}\Big) \geq \phi + \sum_{m,f_i,t_i}\varphi_{m,u}^*\beta_i f^{(j)}(\pi_{m,u}^{f_i,t_i},(\pi_{m,u}^{f_i,t_i})^{(j)}), \tag{11}$$

where $\phi \triangleq \big(q_m^{\mathsf{max}} - q_m^{\mathsf{em}}[t-1] - \sum_u \varphi_{m,u}^*\lambda_u^{\mathsf{em}}[t]Z^{\mathsf{em}}\Delta\big)^+$. Finally, to handle the non-convexity of constraint (4), we introduce the auxiliary variables $\boldsymbol{y} = [y_{m,u}^{f_i,t_i}]^T; \forall u \in \mathcal{U}^{\mathsf{ur}}$ such that $y_{m,u}^{f_i,t_i} \leq I_{m,u}^{f_i,t_i}$ then via the first-order Taylor approximation technique we approximate it as

$$(y_{m,u}^{f_i,t_i} + \pi_{m,u}^{f_i,t_i})^2 \leq 2\pi_{m,u}^{f_i,t_i}(\frac{p_{m,u}^{f_i,t_i}g_{m,u}^{f_i,t_i}}{\Gamma^{\mathsf{th}}} - N_0) + 2(y_{m,u}^{f_i,t_i})^{(j)}y_{m,u}^{f_i,t_i} + 2(\pi_{m,u}^{f_i,t_i})^{(j)}\pi_{m,u}^{f_i,t_i} - ((y_{m,u}^{f_i,t_i})^{(j)})^2 - ((\pi_{m,u}^{f_i,t_i})^{(j)})^2; \forall u \in \mathcal{U}^{\mathsf{ur}}. \tag{12}$$

Bearing all the above approximations in mind, the convex approximate program (8) is solved at iteration j as

$$\max_{\boldsymbol{\pi},\boldsymbol{z},\boldsymbol{y}} \quad \Xi^{(j)} \tag{13a}$$

$$\text{s.t.} \quad y_{m,u}^{f_i,t_i} \leq I_{m,u}^{f_i,t_i}; \quad m, f_i, t_i, \forall u \in \mathcal{U}^{\mathsf{ur}} \tag{13b}$$

$$(8b), (9), (10), (11), \text{and} (12).$$

Steps 11-17 in Algorithm 1 provide a summary of the SCA-based iterative algorithm.

Algorithm 1 Hierarchical Proposed Solution for Solving (7)

Initialization: Set $t = 1$, $\varphi_{m,u}$ is initialized as each UE connects to the nearest RU, the queue $q_{m,u}^{\mathsf{x}}[1]$ is initialized randomly, UEs' location $(\boldsymbol{x}^{\mathsf{UE}}, \boldsymbol{y}^{\mathsf{UE}})$, and RUs' location $(\boldsymbol{x}^{\mathsf{RU}}, \boldsymbol{y}^{\mathsf{RU}})$. Set $U' = \emptyset$, $M' = \emptyset$, and $\mathbb{S} = \mathbf{0}_{M\times U^{\mathsf{em}}}$ as a set of re-associated UEs, congested RUs, and scores. Set $j := 0$ and generate the initial feasible point as $\boldsymbol{\pi}^{(0)}[t] := \boldsymbol{\pi}[t-1]$.

1: **for** $t = 1, 2, \ldots, T$ **do**
2: `UA-xAPP`: Given historical data from previous W frames: the average queue length $\bar{q}_m^{\mathsf{em}} \triangleq \frac{1}{W}\sum_{t-W}^{t} q_m^{\mathsf{em}}[t]$, $\varphi_{m,u}$, and the network conditions,
3: **while** $\bar{q}_m^{\mathsf{em}} \geq q^{\mathsf{th}}$ **do**
4: $M' \leftarrow M' \cup \{m\}$, measure score of RU m' and all UEs (*i.e.*, $u \in U_m$) as
5: **for** $m' = 1, 2, \ldots, M\setminus\{m\}$ **do**
6: **for** $u = 1, 2, \ldots, U_m$ **do**
7: Calculate the normalized distance as $d_{m',u}$, the normalized average queue length $\bar{q}_{m'}$ and $\bar{q}_u$. Then measure $\mathbb{S}_{m',u} = \rho_1(\frac{1}{1+d_{m',u}}) + \rho_2(\frac{1}{1+\bar{q}_{m'}}) + \rho_3(\frac{1}{1+\frac{1}{\bar{q}_u}})$, where $\rho_i \in [0,1], \sum_i \rho_i = 1$ are the priority factors.
8: **end for**
9: **end for**
10: Update UA variable based on RU with the maximum score $\varphi^*_{\{\arg\max_{m',u}\mathbb{S}\}} = 1$.
11: **end while**
12: `RB Scheduler`: Given the optimized $\boldsymbol{\varphi}^*$, the estimated channel gains feedback from the RUs to the DUs and CUs, and with the assumption of uniform power allocation, $\boldsymbol{\pi}$ is optimized by solving the optimization problem (8) with the help of the SCA Algorithm as follows
13: **repeat**
14: Solve (13) to obtain $(\boldsymbol{\pi}^*, \boldsymbol{z}^*)$ and Ξ^*;
15: Update $\boldsymbol{\pi}^{(j)} := \boldsymbol{\pi}^*$ and $\Xi^{(j)} := \Xi^*$;
16: Set $j := j + 1$;
17: **until** Convergence or $|\Xi^{(j)} - \Xi^{(j-1)}| \leq \varepsilon$ {/**Satisfying a given accuracy level**/}
18: Recover an exact binary by computing $\boldsymbol{\pi}^* = \lfloor\boldsymbol{\pi}^{(j)} + 0.5\rfloor$;
19: `Power Control`: A MAD3QN method presented in Algorithm (2) via M agents predicts transmission power allocation to each RB based on updated $\boldsymbol{\varphi}^*$, optimized $\boldsymbol{\pi}^*$, and their local observations.
20: Update queue length as $q_{m,u}^{\mathsf{x}}[t] = \big(q_{m,u}^{\mathsf{x}}[t-1] + \varphi_{m,u}^*\lambda_u^{\mathsf{x}}[t]Z^{\mathsf{x}}\Delta - \sum_{t_i}(R_{m,u}^{\mathsf{x}}(\boldsymbol{\varphi}^*, \boldsymbol{p}^*, \boldsymbol{\pi}^*)\delta_i)\big)^+$
21: Update $\{\boldsymbol{\varphi}, \boldsymbol{q}, \boldsymbol{\pi}\}$.
22: **end for**

C. Power Control

To efficiently handle dynamic environments with incomplete information, we propose a cooperative multi-agent DRL approach for power allocation, reducing the computational complexity. Each agent utilizes local observations to predict power without knowledge of interference in a decentralized manner. To improve learning efficiency and rapid convergence, this study adopts the dueling double deep Q-network (D3QN) approach as depicted in Fig. 1 for the agent per RU. The problem at hand is commonly modeled as *Markov Decision Process*, where the agent $m \in \mathcal{M}$ can learn with a tuple of four elements $(\mathbf{s}_m[t], \mathbf{a}_m[t], \mathbf{r}_m[t], \mathbf{s}_m[t+1])$. Accordingly, each agent m in multi-agent D3QN (MAD3QN) model chooses an action $\mathbf{a}_m[t] \in \mathcal{A}_m$ for the current state $\mathbf{s}_m[t] \in \mathcal{S}_m$. Then, each agent moves to the next state $\mathbf{s}_m[t+1]$, while receiving the reward $\mathbf{r}_m[t]$ based on its action $\mathbf{a}_m[t]$. As each agent receives distinct rewards, selfish behavior might arise, impacting overall network performance among interference from other agents. Due to the cooperative nature, the implementation of a flag system enables distributed agents to collaborate efficiently with minimal communication overhead just by sharing their flags together. Here, $\mathtt{F}_m[t] \in \{0,1\}$ shows the flags of the agents in frame t.

Algorithm 2 MAD3QN-based Power Control

Initialization: Randomly initialize weights of online ($\boldsymbol{\theta}$) and target ($\boldsymbol{\theta'}$) networks as $\boldsymbol{\theta} = \boldsymbol{\theta'}$, and set $\boldsymbol{r}[1] = \mathbf{0}_{\boldsymbol{M}\times\mathbf{1}}$, and $\mathtt{F}[1] = \mathbf{0}_{\boldsymbol{M}\times\mathbf{1}}$.

```
for epoch do
    Initial observation states s_m[1] ∈ S_m; ∀m;
    for t = 1, 2, ..., T do
        for m = 1, 2, ..., M do
            Take action a_m[t] ∈ A_m via ε-greedy method, then if meets
            constraint (3) F_m[t] = 1, otherwise F_m[t] = 0
            if ∏_m F_m[t] = 1 then
                r_m[t] += r^com
            else if F_m[t] = 1 then
                r_m[t] += r^indv
            else
                r_m[t] += r^neg
            end if
            Receive the reward r_m[t] and move to next state s_m[t+1]
            Store transition (s_m[t], a_m[t], r_m[t], s_m[t+1]) to the replay
            memory of agent m;
            Sample random mini-batches from replay memory m;
            Update θ by minimizing the loss function [9]; while update
            θ' every C steps by resetting θ' = θ.
        end for
    end for
end for
```

State, Action Spaces and Reward Function: The state space encompasses the observation accessible to each agent per RU, for example, the m-th state in frame t is a combination of RB scheduled to UEs served by RU m ($u \in \mathcal{U}_m$) and their channel gains, the m-th action and all Flags in the previous frame, which is expressed as $\mathbf{s}_m[t] = \Big\{\boldsymbol{\pi}_m[t] \times \boldsymbol{G}_m[t], \mathbf{a}_m[t-1], \mathtt{F}[t-1]\Big\}$.

The discrete power domain has been widely used as a feasible solution for learning-based dynamic scheduling for multi-traffic scenarios [12]. This method can ensure stable convergence and reduce the computational complexity of distributed learning models conducted by RUs who have limited computational resources. Given this context, we consider a discrete action space, where the power is quantized into L levels which are determined as $\hat{p}_l = lP^{\mathsf{max}}_m/L; l = \{1, \dots, L\}$, where $\hat{p}_l$ is the l-th transmission power level. Let us assume K_m as the total number of RBs that m-th RU occupied to serve its linked UEs per frame. Thus, the action for the m-th agent in frame t from the action space $\mathcal{A}_m$ is defined as $\mathbf{a}_m[t] = \Big\{\mathbf{a}^{1,1}, \dots, \mathbf{a}^{k_m,l}, \dots, \mathbf{a}^{K_m,L}\Big\}$. Where, $\mathbf{a}^{k_m,l}$ indicates that agent m select l-th transmission power level for RB index k_m. Thus, the action space size of agent m is LK_m and the overall action space size of all agents is $(LK_m)^M$.

To design an efficient reward function, this paper adopts a penalty-based approach that integrates constraints (3), (4), and (7b) into the main objective function. To this end, we formulate a reward mechanism for each agent such that a negative value r^{neg} is set as its reward value to punish the agent if it does not meet the constraint (3), then its $\mathtt{F}_m[t] = 0$, otherwise $\mathtt{F}_m[t] = 1$, it deserves a r^{indv} based on its action $\boldsymbol{a}_m[t]$ as $r_m[t] = \kappa \frac{\sum_{u\in\mathcal{U}^{\mathsf{em}}}(R^{\mathsf{em}}_{m,u} - R^{\mathsf{min}})}{R_0} + (1-\kappa)\frac{\sum_{u\in\mathcal{U}^{\mathsf{ur}}, f_i, t_i}(\Gamma^{f_i,t_i}_{m,u} - \pi^{f_i,t_i}_{m,u}\Gamma^{\mathsf{th}})}{\Gamma_0}; \forall u \in \mathcal{U}_m$, where κ is the priority factor to balance both services, while Γ_0 denotes the reference of uRLLC SINR. Finally, to improve overall network performance and avoid selfish behaviour of agents during the learning process, we consider $r^{\mathsf{com}} \triangleq \sum_m r_m[t]$ to feed to all agents if $\prod_m \mathtt{F}_m[t] = 1$. Throughout the learning process, agents explore the environment to find the best policies while there is no complete observation from others, such as interference that we expect agents to learn that will maximize their reward, ultimately leading to optimal network performance.

IV. Simulation Results

We simulate a system comprising 3 RUs and 20 UEs which are randomly located in a circular area with a radius of 500 m. We model the quasi-static channel vector between RU m and UE u on RB(f_i, t_i) of numerology i as $h^{f_i,t_i}_{m,u} = \sqrt{10^{-\mathsf{PL}_{m,u}/10}}\bar{h}^{f_i}_{m,u}$, where $\mathsf{PL}_{m,u}$ is log-distance path-loss, and $\bar{h}^{f_i}_{m,u} \in \mathbb{C}^{1\times K_{\mathsf{tx}}}$ represents Gaussian noise with zero mean and unit variance. We assume that the u-th traffic flow of eMBB/uRLLC follows the FTP3 model with a mean arrival rate of 21.12 and 2.12, respectively. Based on $P_e = 10^{-3}$, this paper chooses MCS 12 with $\Gamma^{\mathsf{th}} = 17$ dB, while $\alpha = 0.3$, $q^{\mathsf{th}} = 0.8q^{\mathsf{max}}_m$ and $B_G = 180$ kHz. The proposed algorithm (1) runs on $T = 100$ sub-frames, followed by the 5G NR frame structure. The rest of the simulation parameters are given in Table I.

Benchmark Schemes: The proposed Algorithm 1 is compared with two benchmark schemes: **Scheme 1**, which follows Algorithm 1 except that each RU equally allocates power to its users as $p^{f_i,t_i}_{m,u} \triangleq P^{\mathsf{max}}_m / \sum_i F_i$, and **Scheme 2**, which employs OFDMA mechanism across the RUs in a centralized approach, while interference is absent [9].

Fig. 2(a) presents the learning performance of Algorithm 2 in terms of the normalized accumulated reward for different transmit power levels (L) over 500 episodes. Higher power levels yield greater rewards but also increase the action space complexity. Distributed agents effectively adapt to dynamic environments with varying factors such as arrival packets, channel states, and changes in UA variables across frames. Since the total number of RBs per TTI is $\sum_i F_i = 13$, we select L in varying ranges (less than, equal to, and greater than 13) to demonstrate their effectiveness in network performance. As evident in Fig. 2(a), a power level of L = 20 achieves the highest performance compared to other levels as L = 13, 8, and 4. A reward mechanism employing a flag system manages interference, ensuring stable agent behavior. The steadily rising reward throughout training episodes indicates an increase in attainable average rewards.

Fig. 2(b)-(d) illustrate a performance evaluation and com-

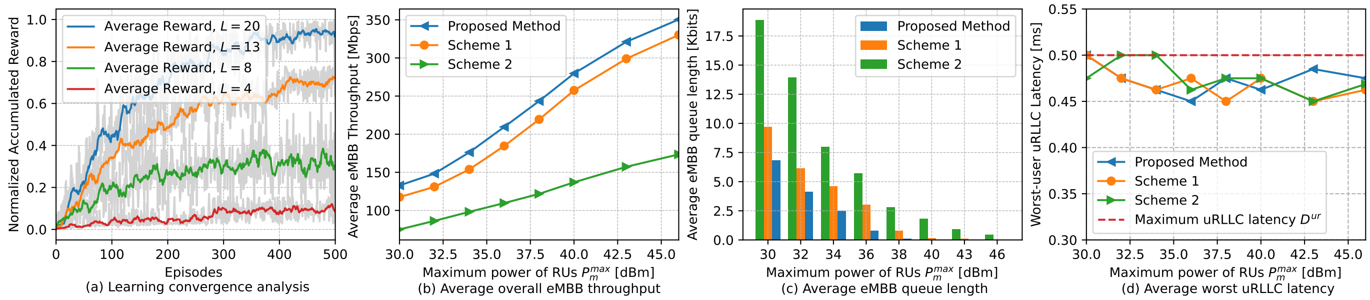

Fig. 2: The convergence behaviour of Algorithm 1 and performance comparison with benchmark schemes

parison for different transmit power budgets $P_m^{\max}$. Increasing the power budget range from 30 to 46 dBm leads to increased eMBB throughput while at the same time reducing the eMBB queue length on all schemes. Specifically, Fig. 2(b) illustrates the overall average eMBB throughput as a function of $P_m^{\max}$. The proposed method clearly outperforms both benchmark schemes across the range of $P_m^{\max}$ values. Notably, at $P_m^{\max} = 43$ dBm, the performance gap between the proposed method (Algorithm 1) and Scheme 1 is nearly 11.3%. This significant difference underscores the superiority of the proposed power control strategy over the equal power allocation approach used in Scheme 1. The ability of the proposed method to dynamically adjust power allocations to each RB proves to be more effective in optimizing network performance. Compared to Scheme 2, the proposed method exhibits an even more substantial improvement of 106.27% at $P_m^{\max} = 43$ dBm. This remarkable enhancement highlights the efficacy of the MAD3QN model in managing power control to mitigate interference from neighboring RUs.

In Fig. 2(c), the average backlog is depicted for different benchmark schemes. The results show that higher power budgets correspond to lower average queue lengths. Consistent with previous findings, the proposed method demonstrates superior performance in reducing the queue buffer. Algorithm 1 can completely vacate the buffer at $P^{\max} \geq 40$ dBm, indicating its efficient power allocation method. Scheme 2 exhibits the poorest performance in terms of average queue length. Additionally, the proposed method outperforms Scheme 1, as shown in Fig. 2(c), due to Scheme 1's inefficient handling of inter-RU interference. In Fig. 2(d), the average worst-user uRLLC latency is portrayed across various maximum power levels. Due to the uRLLC latency formulated in this paper, when packets arrive randomly within a frame, a predefined latency (D^{ur}) must be satisfied. This visualization indicates that all schemes meet the required uRLLC latency of 0.5 ms shown as dashed red line in Fig. 2(d). Notably, this finding indicates the ability of all schemes to meet the uRLLC latency requirements with the available resources.

TABLE I: Simulation Parameters

Parameter	Value	Parameter	Value
BW of RU	5 MHz	No. of layers, units	5, 512
No. RBs per sub-frame	52	Learning rate	0.0001
Noise power	-137 dBm	Discount factor	0.99
uRLLC packet size	32 B	Maximum reply memory size	1e+05
eMBB packet size	64 KB	Batch size	64
Required eMBB data rate	10 Mbps	Optimizer	*adam*
Required uRLLC latency	0.5 ms	Activation function	*ReLU*
Maximum RU's queue-length	10 KB	r^{neg}	-2

V. Conclusions

This paper presented an intelligent framework for UA, resource scheduling, and power control to support multi-service scenarios while preventing cell congestion. Leveraging NS technology, the mini-slot concept, and mixed scalable numerologies within an Open RAN architecture, we addressed dynamic environments with varying traffic demands. To manage dynamic UA and scheduling in line with the 7.2x FS and reduce complexity, we adopted a hierarchical optimization approach using heuristic, SCA, and D3QN-based multi-agent models across different components. Simulations demonstrate the superior performance of our design compared to benchmark schemes.

References

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

[2] L. Gavrilovska, V. Rakovic, and D. Denkovski, "From cloud RAN to open RAN," *Wireless Personal Communications*, vol. 113, pp. 1523–1539, 2020.

[3] V. Beschastnyi, D. Ostrikova, S. Melnikov, and Y. Gaidamaka, "Modelling Multi-connectivity in 5G NR Systems with Mixed Unicast and Multicast Traffic," in *International Conference on Distributed Computer and Communication Networks*, pp. 52–63, Springer, 2020.

[4] 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.

[5] M. Mollel, S. Kaijage, and K. Michael, "Deep reinforcement learning based handover management for millimeter wave communication," 2021.

[6] Y. Cao, L. Zhang, and Y.-C. Liang, "Deep reinforcement learning for multi-user access control in UAV networks," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2019.

[7] Y. Cao, S.-Y. Lien, Y.-C. Liang, K.-C. Chen, and X. Shen, "User access control in open radio access networks: A federated deep reinforcement learning approach," *IEEE Transactions on Wireless Communications*, vol. 21, no. 6, pp. 3721–3736, 2021.

[8] 3GPP, "Study on New Radio (NR) Access Technology Physical Layer Aspects," *Technical Report (TR) 38.802, V14. 2.0*, 2017.

[9] F. Kavehmadavani, V.-D. Nguyen, T. X. VU, and S. Chatzinotas, "On Deep Reinforcement Learning for Traffic Steering Intelligent ORAN," in *IEEE Global Communications Conference*, 2023.

[10] D. López-Pérez, A. Ladányi, A. Jüttner, H. Rivano, and J. Zhang, "Optimization method for the joint allocation of modulation schemes, coding rates, resource blocks and power in self-organizing LTE networks," in *2011 Proceedings IEEE INFOCOM*, pp. 111–115, IEEE, 2011.

[11] 3GPP, "5G; NG-RAN; Architecture description (3GPP TS 38.401 version 16.3.0 Release 16)," Tech. Spec. 38.401, 3GPP, Nov 2020.

[12] F. Meng, P. Chen, L. Wu, and J. Cheng, "Power allocation in multi-user cellular networks: Deep reinforcement learning approaches," *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 6255–6267, 2020.