Intelligent QoS-aware Slice Resource Allocation with User Association Parameterization for Beyond 5G O-RAN-based Architecture Using DRL

Suvidha Mhatre, *Student, IEEE*, Ferran Adelantado, *Senior, IEEE*, Kostas Ramantas, *Senior, IEEE*, and Christos Verikoukis, *Senior, IEEE*,

***Abstract*—The escalating demands of next-generation, beyond-5G services necessitate innovative approaches to dynamic resource management, critical for satisfying end-user expectations and maintaining quality of service (QoS). This study leverages the Open Radio Access Network (O-RAN) architecture's flexibility and programmability to introduce a novel deep reinforcement learning (DRL) strategy for QoS-aware intra-slice resource allocation. By employing intelligent agents and a Deep Q-Network (DQN)-based framework, our approach precisely tailors resource distribution within O-RAN, optimizing for enhanced mobile broadband (eMBB) and ultra-reliable low-latency communications (URLLC) slices. The proposed method, featuring intelligent QoS-aware resource allocation (IQRA) and its low-complexity variant (LIQRA), demonstrates significant throughput improvements for eMBB by 11.5% compared to state-of-the-art (SOTA) methods and reduces URLLC latency by 19.94% and 16.54%, achieving up to 45.5% lower latency than baseline. A streamlined algorithm effectively reduces computational complexity, ensuring robust performance under resource constraints. Simulation results underscore the algorithm's ability to substantially enhance 5G network slice performance, offering a parameterized solution for user association in O-RAN networks using DRL. This research not only meets high key performance indicators (KPIs) but also advances edge intelligence, fostering a more responsive network ecosystem.**

***Index Terms*—Slicing, DRL, QoS, Resource allocation and management, URLLC, eMBB, KPI**

I. Introduction

BEYOND 5th generation (5G) envisioned vertical applications and use cases, ranging from Industry 4.0 and smart cities to vehicular communication (V2X), video streaming and immersive experiences like augmented reality (AR) and virtual reality (VR), have set forth a diverse spectrum of requirements to satisfy the quality of service (QoS) of end users. These requirements are encapsulated within the 3rd Generation Partnership Project (3GPP) defined use cases: enhanced mobile broadband (eMBB), ultra-reliable low latency communications (URLLC), and massive machine-type communication (mMTC), each with varying key performance indicators (KPIs) depending on the application. To address the complexities and dynamic nature of these requirements, this research introduces a sophisticated programmable architecture inspired by the Open Radio Access Network (O-RAN) alliance. O-RAN is an architecture designed to achieve an open, intelligent, virtualized, and fully interoperable radio access network (RAN). Consequently, virtualization and intelligence are two of the main pillars of O-RAN. This work delves into these two concepts. O-RAN not only facilitates an infrastructure-based abstraction for both networks and applications but also enables closed-loop control for RAN and radio resource management (RRM). In this context, network slicing emerges as a pivotal technology enabler. It allows for the customization of services to meet specific demands [1], thereby adapting the network to fluctuating traffic requirements [2], [3].

This adaptability is particularly critical for managing the diversity and complexity of criteria for future applications under the eMBB and URLLC use cases, prompting a focused investigation into these areas by establishing distinct slice categories for each. The integration of artificial intelligence (AI) within this O-RAN-based architecture has attracted significant attention due to its potential to surmount the challenges inherent to wireless communications [4]. These challenges stem from the unpredictable, stochastic and complex nature of the models and parameters involved in RRM. AI, particularly through the application of model-free techniques such as reinforcement learning (RL) and its more sophisticated counterpart, deep reinforcement learning (DRL), offers a promising solution [4]. The DRL algorithms, including deep Q networks (DQN) and deep deterministic policy gradient (DDPG), have shown success in optimizing resource allocation within RANs, even before the advent of O-RAN, in architectures such as cloud RAN [5]. Since slicing leverages the virtualization potential to instantiate slices based on the traffic needs, intelligence must fuel the operation and management/orchestration of the resources for the slices. In that sense, given that DRL algorithms are intelligent control mechanisms, we couple these three crucial technology enablers and propose a novel method for slice-level resource management. Handling resources at the RAN edge domain presents challenges, notably traffic load variations that impact optimal resource allocation, thereby reducing resource utilization and causing system-level agreement (SLA) violations, which degrades end-user QoS.

Furthermore, fine-grained resource reconfiguration within network slices entails substantial computational complexity and presents a sequential challenge in resource management [4], [6]. This complexity is compounded by the need to preserve connectivity, manage rates, and facilitate offloading, among other RAN management tasks. This research work

Copyright (c) 2024 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

proposes a framework utilizing the synergies between O-RAN, network slicing, and AI to overcome these challenges in resource management. O-RAN enables the deployment of AI at the edge of the network, where intelligent agents can be applied to learn specific strategies tailored to the unique QoS requirements of different slices. The study investigates a novel dynamic resource allocation technique that operates at the intra-slice level, implementing a parameterized user association (UA) method that addresses gaps left by state-of-the-art (SOTA) approaches. This method functions within a 1-20 ms time frame, aligning with near-real-time (near-RT) control loops, and can be implemented as xAPPs within the near-RT RAN intelligent controller (RIC). This approach not only demonstrates the effective utilization of AI in optimizing RRM and O-RAN architecture along with slicing, but also integrates these technologies with the unique contributions to support dynamic, efficient, and QoS-compliant resource allocation strategies. This strategic confluence aims to provide end users across various slice categories with an enhanced quality of experience (QoE) and superior QoS, setting a new benchmark for resource management in 5G and beyond networks.

II. Related Work

The O-RAN Alliance is actively introducing a non-proprietary version of the RAN system that allows interoperation between network components by different vendors. In the past, intelligent solutions have also been proposed with proprietary architectures, as given in [5], [7], [8] but O-RAN architecture introduces a sophisticated programmable approach with RIC that enables the application of intelligence at different levels of infrastructure. In the O-RAN architecture, there are two separate components: the non-real-time RAN intelligent controller (non-RT RIC) and the near-real-time RIC (near-RT RIC), which reinforces the importance of the two control levels with different time granularities [9]. The O-RAN report and specification include a comprehensive review of a number of use cases that are expected to be integrated with architectural elements to exchange data between various network components. One of the use cases describes how to integrate intelligence at the network's edge in regards to RRM [10]. The components of the O-RAN framework enable the integration of dynamic changes at almost real-time levels for radio resource allocation to end users. RAN resource allocation across different slices is optimized at both intra- and inter-slice levels, engaging with the O-RAN architecture's near-RT and non-RT RIC components for shorter and longer timescales, respectively.

The research in [11] addresses joint power and physical resource block (PRB) allocation with a user association scheme. Although it aligns with the O-RAN architecture, it does not incorporate AI techniques to deliver various KPIs. The authors in [8] define an optimization problem that aims to maximize user perceived throughput while minimizing packet delay violations by modifying the medicum access control (MAC) scheduling algorithm's parameters. It comes under the intra-slice resource allocation category. This specific work is limited to the traditional network architecture, proposing optimization for individual base stations and their underlying users. Such proprietary solutions can be used with O-RAN architecture, but they lack the adaptation to utilize programmable O-RAN-based infrastructure, which facilitates access to the information available at a centralized entity from other base stations or O-RAN radio units (ORUs), which plays a crucial role in RRM decisions. The state-of-the-art intra-slice allocation discussed in [12], introduces a team learning algorithm to enhance O-RAN network performance by fostering xAPP collaboration, primarily focusing on throughput maximization. On the other hand, [13] explores inter-slice allocation through a correlated Q-learning strategy, aiming for improved throughput and reduced queueing delay.

However, these approaches require significant computational resources. Our method advances these concepts by efficiently managing user service based on slice-specific traffic, bandwidth allocation, KPIs, and channel conditions, offering a more nuanced understanding of resource distribution. Traditional table-based RL techniques are infeasible to handle large state and action spaces, as concluded in [6]. Whereas, deep reinforcement learning overcomes these limitations, like DQN and some of its variations, such as dueling deep Q-learning (DDQN) and DDPG. In the case of large and multidimensional discrete action spaces, branching architecture can be introduced in dueling Q networks, as presented in [14]. The researchers in [6] extend the given neural network (NN) for inherently discrete multidimensional and large action spaces to solve the network slice reconfiguration problem. More computational resources are required to support such solutions.

In state-of-the-art work such as [15], researchers opt for the approach of using parallel and multiple deep neural networks (DNN) to make optimal decisions for user association with the base station. The solution proposed in [15] builds multiple DNNs and a number of additional DNNs with random input to increase exploitation. It has K DNN, N random decisions, and $K + N$ functions to calculate the Q value for a network with a certain number of users. Here, the DRL goes through all the layers in every run. Furthermore, each DNN is trained to learn the optimal decision for the entire UA matrix. It requires higher computational capabilities. Instead of this heuristic-style approach, the association decision can be parameterized to learn useful information about the network, such as base station or ORU-based parameters. RL is a very effective tool to achieve optimal decisions in complex environments, such as non-deterministic wireless networks, similar to several aspects discussed in Section I.

A network-slicing 5G architecture and framework to enable efficient robotics operation while improving QoS are discussed in [16]. It provides insight into a user-centric paradigm for integrating vertical knowledge into existing 5G solutions. Different types of slicing framework considerations, edge computing aspects, and orchestration options are evaluated as part of the architecture. It is possible to establish various resource allocation strategies for network slice utilizing a variety of RL approaches [17], [18]. Dynamic network slice distribution techniques to improve performance for 5G-

based bandwidth offerings are evaluated in [17]. It compares different algorithms for the distribution of slice resources. One of the research works uses an approximation of resources and explores subchannel allocation that is different with changes in multiplexing techniques [18]. Future solutions need to be formulated in such a way that they work efficiently and provide the intended result, regardless of variations in techniques used in the wireless communication network underneath.

Although both our work and [19], [20] studies utilize DRL, specifically DQN, our Markov Decision Process (MDP) design diverges significantly. The cited research adopts an approach to managing activities and accessibility within virtualized RAN (vRAN) environments, focusing on QoS utility satisfaction and resource adjustment dynamics. Recent advancements in architecture envision these decisions as inter-slice or non-RT operations. In contrast, our investigation introduces intelligent agents at the intra-slice or near-RT level that assess key parameters for user association, with decisions informed by experienced KPIs as well as other metrics. A notable innovation in our model is the UA parameterization and the MDP design, which form the cornerstone of our decision-making process.

The study in [21] presents a DRL-based framework for dynamic resource allocation in 5G network slicing, emphasizing the autonomous partitioning of resources across various logical networks, each catering to unique requirements under varied management. It leverages transfer learning to extend its applicability across different network topologies, offering a scalable and flexible solution for managing the complexities of 5G networks. The research in [22] delves into multi-agent reinforcement learning (MARL) for optimizing RAN slicing within vehicular networks, addressing the dynamic allocation of spectrum and computing resources amidst varying QoS requirements. A deep deterministic policy gradient-based algorithm underpins the proposed MARL framework, enabling base stations to act as independent but cooperative agents for resource allocation, achieving a notable QoS satisfaction rate.

The MARL approach described in [22] involves substantial information exchange and cooperation among small and macro base station agents, increasing overhead and network complexity, which contrasts with our methodology. Our proposed solution utilizes xAPPs within the O-RAN architecture, promoting network openness without the need for inter-agent communication, thereby simplifying the system and reducing costs. Whereas, in [23], an energy-efficient DRL-assisted strategy (EE-DRL-RA) is introduced, combining DRL and deep learning to enhance energy efficiency and ensure performance isolation among slices, showcasing superior performance in energy efficiency and inter-slice isolation. Lastly, [24], [25] proposes SARA and DSARA, RL and DRL-based solutions for admission control and resource allocation in 5G core network slicing, aiming to maximize service provider profit and network resource utilization, offering a significant advancement in intelligent and efficient network slice request management.

Unlike the metrics taken into account in [7], [26], [27], [28] and other papers, the various slices in the network have different KPI thresholds to fulfill the defined QoS requirements. This poses a challenge in terms of achieving optimal system performance that is tackled in this paper by considering the variation in metrics or KPIs for different types of services that come under various RAN slice categories. Hence, the network should support distinct slice KPI-based reward definitions for individual end users to evaluate different service metrics. For instance, previous research [29], [30] considered different types of services based on varying bit rate and delay requirements, each of which took into account a distinct set of metrics for its requirements. Motivated by this work, the research presented in this paper considers the variation of QoS thresholds and KPI metrics as well as their impact while allocating resources to various slices in the formulation.

This work does not prioritize only a specific service category, as discussed in the above papers, such as constant bit rate. Instead, it defines weights to evaluate metrics of different slice categories included in the reward function design of intelligent agents. Furthermore, each slice category has a separate intelligent agent that learns the importance of parameters and receives rewards based on weighted metrics and QoS thresholds for different slice categories. The proposed work discusses a system model based on O-RAN architecture to deal with intra-slice RAN RRM decisions taken in real time for eMBB and URLLC slices. The problem is formulated to achieve optimal system performance and KPIs while satisfying the QoS of individual end users. In contrast to the work talked about in [8], the proposed work intends to learn the importance of UA parameters with a unique approach for slice-based beyond 5G networks with multiple ORUs. The presented research work considers resource availability at all ORUs to serve the buffered traffic at users under specific categories and allocates the resources to achieve a high QoS.

As indicated in the introduction, traffic uncertainty affects RAN edge domain resource allocation [4], [6]. Hence, the formulation in this paper considers parameters reflecting network traffic load that have a significant impact on resource management. It takes into consideration the most crucial aspects of a wireless communication network in terms of end users and ORUs present in the network, as indicated in the problem formulation section. A resource allocation or reconfiguration problem can be written in such a way that the action space is constrained to a limited number of actions to which DQN can converge. It is also vital for intra-slice RRM in terms of time granularity of 1–10 ms for the near-RT RIC component of O-RAN.

The contributions of this paper are as follows:

1) Our algorithms innovatively address the user-ORU association and manage traffic loads for different slices specifically tailored for the demanding KPIs of beyond-5G applications. Unlike SOTA methods, our approach, including the algorithm and its intelligent agent, operates from the near-RT RIC component at the remote edge within the O-RAN architecture. It is uniquely designed for O-RAN, enabling dynamic resource configurations on a rapid timescale of 1-20 ms and leveraging centralized data on end-users and ORUs for decision-making. Our technique stands out for its ability to dynamically

manage resources within small time scales by employing synergy between AI, O-RAN, and slicing.

2) An informed decision for user association is taken based on not only user performance but also the ability of ORU to serve users, defined as a novel metric τ (See Section V for detail definition). While SOTA in the field often focuses primarily on learning from channel conditions, or KPIs [12], [13], our work sets itself apart by introducing metric, τ. This metric encapsulates the ORU's capacity to support users in specific slice categories, evolving based on past experiences and ORU performance which is considered while taking decisions in addition to channel conditions and KPIs. The distinct approach to include τ aligns with O-RAN standards as well as advances beyond conventional methodologies by offering a comprehensive and informed as well as dynamic resource allocation.
3) The proposed formulation ensures limited action spaces that improve performance, reducing convergence time. Usually, the DQN approach iterates through all actions, and hence it can be computationally complex with a large number of actions. In addition to this, as it avoids large action spaces, the desired results are achieved without additional NN architectures or DQN variants, unlike [6], [14]. The computation capabilities required at the edge server, processing time, and training time are reduced compared to other approaches. This is vital in the case of smaller time scales for near-real-time decisions. Furthermore, the action space defined in the proposed MDP is independent of the number of users in the network.
4) Instead of learning the whole user association matrix as done in the state of the art, the proposed work distinguishes itself by parameterizing the user association. The agents learn the weights for each of the parameters crucial for the UA decision. With this, the computational and memory requirements can be reduced, as can the convergence time. A problem formulation is proposed that utilizes the intelligently selected weights and concludes the actual decision. It avoids the parallel and multiple DNN approaches used in several state-of-the art works for UA.
5) Our study introduces an innovative solution that merges DRL with a targeted analysis of key associations, synergizing these methods to optimize UA parameter prioritization. Distinct from the majority of SOTA methods, our strategy employs a reward function attuned to variations from the QoS benchmarks, thereby refining user performance adjustments. Additionally, we customize KPI weights to align with specific network slices, significantly improving essential performance indicators such as throughput, delay, bit error rate (BER), and overall system efficiency. Our methodology outperforms traditional baseline strategies and conventional DQN MDPs, showcasing its superior efficacy.

III. Network Architecture

In our study, we explore a RAN architecture as shown in figure 1, that adheres to the O-RAN specifications outlined by the O-RAN Alliance [31]. This RAN setup serves a multitude of users across various network slices, each designed to fulfill distinct QoS needs. Users are linked to an ORU, essentially a radio transceiver equipped with antennas and low physical layer (PHY). This component is connected to both the O-RAN Distributed Unit (ODU) and the O-RAN Centralized Unit (OCU), which include higher protocol layers.

Within this architectural framework, every user is served by an ORU within a specific ietwork slice subnet instance (NSSI), as per O-RAN specifications. Each NSSI is aligned with particular service requests and associated KPIs, ensuring end-to-end connectivity through the ODU, OCU, the service management and orchestration (SMO) system, and ultimately, the core network. At the edge of the network, the RIC is instrumental in guiding decisions related to resource allocation and management, leveraging the system's openness to enhance decision-making with comprehensive network parameter insights.

The ORU is positioned at the local edge, whereas the ODU, OCU, and near-RT RIC are incorporated into the server at the remote edge setup. Communication between the ORU and user equipment (UE) occurs via wireless channels, with open fronthaul interfaces facilitating the connection to the remote edge for both uplink and downlink transmissions. Our proposed algorithm's practical implementation assumes the availability of perfect channel state information (CSI) and UE positions, gathered from the near-RT RIC's database at the edge server, as conveyed by the E2 node through the E2-CP interface.

Connections between each ODU and OCU at the remote edge server and the SMO are established via O1 and A1 interfaces. The RAN's link to core network entities is maintained through a fronthaul connection, employing mediums like Ethernet or optical fronthaul, each with a specific maximum transmission capacity and constrained by the core network resources allocated to each slice or network subnet at the edge or cloud level, including computational resources.

RAN NSSI resource management operates across two control loop levels—non-RT and near-RT—consistent with O-RAN guidelines [31]. The SMO sets a baseline RAN resource configuration for each ORU through the O1 interface, covering radio and computational resources earmarked for various slice types as part of the non-RT resource management process. This ensures that each RAN NSSI receives a predetermined bandwidth allocation for serving its users. The performance of each network slice is evaluated against the requested QoS settings to meet SLAs.

To enact our dynamic RAN resource management strategy, intelligent agents are deployed, specializing in allocating radio resources across slices efficiently to satisfy the QoS criteria specified by the slicing manager and orchestrator. Each slice benefits from a dedicated intelligent agent tasked with mastering decision-making parameters to optimize slice-specific KPIs, thereby enhancing individual slice performance through

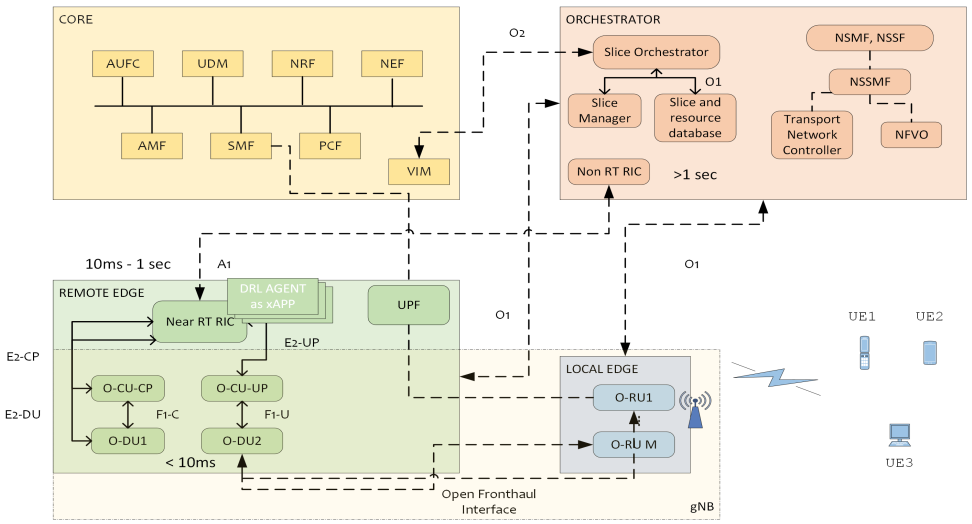


Fig. 1: O-RAN based Network Architecture

informed RRM decisions at the near-real-time RIC level. Aligned with 3GPP, O-RAN, and international telecommunication unit (ITU) specifications for next-generation wireless networks, O-RAN access points are strategically integrated within infrastructure. This ensures that network simulations reflect practical scenarios, facilitating optimization at various network elements for enhanced efficiency.

IV. System Model

As shown in the figure 2, consider the O-RAN network with a general set of users $\mathcal{K} = \{1, \cdots, K\}$. The users can request services under any of the two slice categories eMBB and URLLC, represented by the set of users denoted as $\mathcal{K}_E$ and $\mathcal{K}_U$ respectively, where $\mathcal{K} = \mathcal{K}_E \cup \mathcal{K}_U$ and $\mathcal{K}_E \cap \mathcal{K}_U = \emptyset$. Each user has QoS requirements to be fulfilled based on the type of slice requested. These QoS requirements are expressed in terms of the minimum data rate to be achieved per user $R_k^{\min}$ and the maximum allowed delay per user $d_k^{\max}$ where $k \in \mathcal{K}$. We assume that all users from the same slice have the same minimum data rate and maximum delay requirements. The packet arrival rates for eMBB and URLLC slices follow periodic deterministic traffic model with specific packet arrival intervals. The size of packets and number of packets for each user vary based on the slice category. We assume it is the same for all users under the same slice category indicated for each user k as S_k and L_k, where $k \in \mathcal{K}$. The slice type as well as service requirements are mentioned for each UE in the UE database available at the remote edge. The set of ORUs denoted by $\mathcal{M} = \{1, \cdots, M\}$ serves these users.

To make the best use of resources within a slice, there is a separate intelligent agent for each RAN slice at the near-RT RIC. These agents learn how to better manage and assign radio resources within the ORU's available bandwidth. Its objective is to associate users with ORU and schedule resources in an optimal way to achieve the required QoS performance. We consider downlink (DL) frequency division duplexing (FDD) transmission. Let the binary variable $a_{k,m}$ indicate the association between ORU and users in DL and can be expressed as matrix A. Each user will be associated with one of the ORUs for transmission at a given time for DL.

$$a_{k,m} = \begin{cases} 1, & \text{If } k \text{ is associated with } m \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Where, $m \in \mathcal{M}, k \in \mathcal{K}$

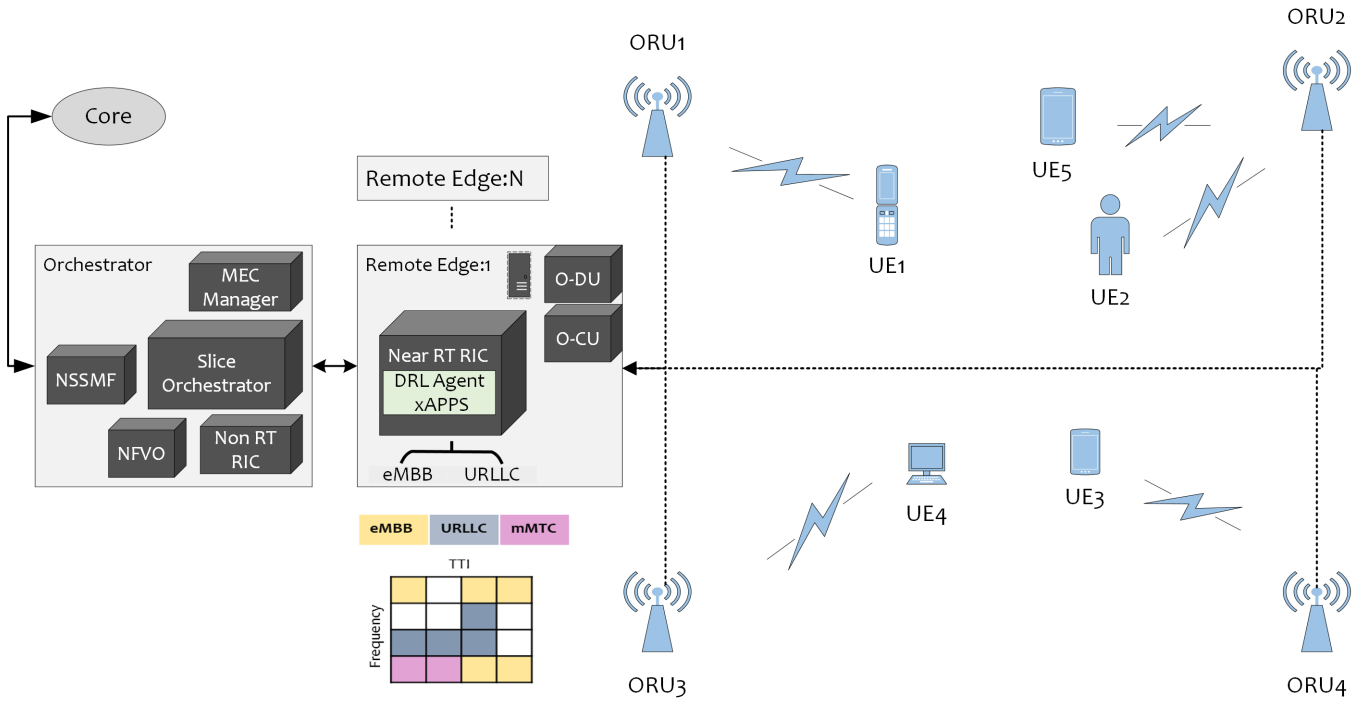
$$A = \begin{bmatrix} a_{11} & \cdots & a_{1M} \\ \vdots & \ddots & \vdots \\ a_{K1} & \cdots & a_{KM} \end{bmatrix} \tag{2}$$

As discussed earlier, there are two types of slices: URLLC and eMBB. These slices can be instantiated in any ORU. Hence, the subset of ORUs serving s slice users can be expressed as:

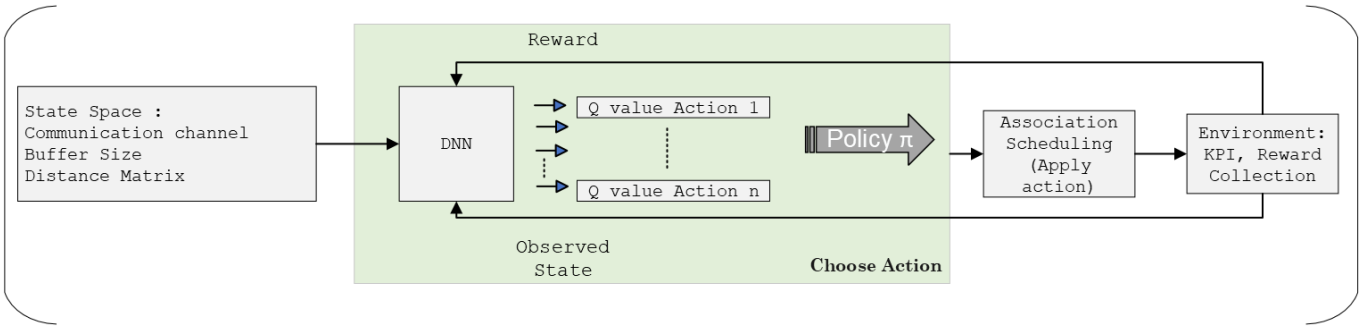
$$\begin{aligned} &\mathcal{M}_s = \{m \in \mathcal{M} \mid a_{k,m} = 1\}, \ k \in \mathcal{K}_s \\ &s \in \{E, U\} \end{aligned} \tag{3}$$

Each slice has a separate association matrix A_s with corresponding ORUs and users. The W bandwidth is divided into $\mathcal{PRB} = \{1, \cdots, PRB\}$ sub-channels. It is also assumed that a user can only be connected to one ORU at a time, and a user may be assigned one or multiple PRBs based on the traffic at each user. The bandwidth allocated to each ORU is divided among the slices instantiated in the ORU. Hence, ORU has a specific number of resources PRB_m available, and the corresponding slice has a total $PRB_{m,s}$ number of PRBs available where $s \in \{E, U\}$. We assume additive white guassian noise for all users of independent circular symmetric natured complex random variables with a zero mean and σ^2 variance. All users in the network follow a random mobility model with speed V. As per 5G new radio (NR), we only consider numerology $\mu = 0$.

Here, all the parameters indicated are with respect to a single transmit time interval (TTI) t; hence, for simplicity, the notation t is omitted from all variables unless the time duration is other than 1 TTI or a reference to a future or past TTI value is required. In DL, we use $h_{k,m}$ to denote the channel



(a) System Model



(b) DRL Agent at Near-RT RIC (Remote Edge)

Fig. 2: System Mdoel

coefficient (including pathloss and shadowing effect) from ORU m to user k. The frequency-selective flat fading channel passes through Rayleigh fading in the wireless medium. Let us assume that P_m is the total transmit power of ORU. The signal-to-interference-noise ratio (SINR) of the k-th user for the wireless link with the m-th ORU for DL can be expressed as,

$$\gamma_{k,m} = \frac{P_m \left|h_{k,m}\right|^2}{I_{k,m} + \sigma^2} \tag{4}$$

Where the interference from the i ORUs (other than the associated m-th ORU) for the k-th user can be formulated as,

$$I_{k,m} = \sum_{i \in \mathcal{M}, i \neq m} P_i \left|h_{k,i}\right|^2 \tag{5}$$

V. Problem Formulation

The proposed formulation aims to allocate and manage radio resources within each slice in a dynamic and optimal way such that all users satisfy the QoS requirements of requested services. The majority of existing proprietary and SOTA-based solutions address the importance of including intelligence at the inter-slice level. But it is equally crucial to include such intelligence in the intra-slice resource allocation to adapt to the diverse KPIs of future envisioned applications. The proposed work tries to fill these gaps while achieving optimal system performance. The proposed algorithms deal with near-RT RIC-level time granularity and are available as xAPPs at the remote edge of the network architecture. The algorithms are implemented in O-RAN architectures (in the direction of cell-less), enabling dynamic radio resource allocation decisions towards end users within the available resources of each slice. It focuses on the optimal and intelligent way to learn the importance of various decision-making parameters. The formulated solutions tend to rely on parameters such as SINR, estimation of PRBs assigned to each user, and ORU capabilities to serve the traffic load in the current network at

any given time.

The SINR values $\gamma_{k,m}$ are calculated as per equation (4) and they include effects such as path loss and shadowing based on the distance as well as the wireless channel quality between the user and ORU. Furthermore, it considers other interfering components in the network that affect the signal strength of the actual signal. The estimation of assigned PRBs is included in one of the proposed algorithms, as prior knowledge of this parameter increases the KPI performance observed through several tests. The ability of ORU to serve traffic in a network is evaluated with a performance metric for ORU, $\tau_{m,t}$ defined as time required to transmit the bits which are currently in buffer at ORU m. It is ratio of total number of bits for transmission in the buffer at ORU m to the total number of bits actually transmitted per TTI at ORU m, as given in Equation (7). This metric is related to delay. It gives measure of the ability of ORU to transmit packets in the buffer.

$$\tau_{m,t} = \frac{C_{m,t}}{J_{m,t}} \tag{6}$$

$$\begin{aligned} C_{m,t} &= \sum_{k \in \mathcal{K}_m} B_{k,t} \\ J_{m,t} &= \sum_{k \in \mathcal{K}_m} TB_{k,t} \end{aligned} \tag{7}$$

where $\mathcal{K}_m$ is a set of users associated with m-th ORU, $B_{k,t}$ is number of bits in buffer for each associated user and $TB_{k,t}$ is number of bits transmitted for each associated user under ORU m at TTI t. The equation (8) provide details on how to update this metric during the simulation over time t. Depending on MAC-level scheduler, different bits transmitted per TTI per user can be observed. Over simulation time we can calculate $\tau_{m,t}$. We store the global value as $\tau^g_{m,t}$ and update it by taking average of the last TTI global value, $\tau^g_{m,t-1}$, and current value, $\tau_{m,t}$. This consider the balanced importance for immediate value based on current association decision equally compared to past experienced τ values.

$$\tau^g_{m,t} = \frac{\tau_{m,t} + \tau^g_{m,t-1}}{2} \tag{8}$$

These parameters represent crucial aspects to be considered in the decision-making process of associating users with ORUs. The proposed formulation computes the indicated parameters to evaluate their importance and learn optimal decisions for the given state of the network environment.

A. Intelligent QoS aware Resource Allocation (IQRA)

The IQRA focuses on selecting the optimal association decision based on a combination of the above-discussed parameters calculated in current TTI. It sits at near-RT RIC as an xAPP. Initially, a database is generated that stores all possible combinations of associations between the available users and ORUs of a specific slice. With this database as a reference, the algorithm can calculate and estimate the required parameters that contribute to association and scheduling decisions within the available slice resources. As discussed in the earlier section, we indicate association with matrix A defined in equation (2). For a simplified approach in IQRA, the association is expressed in a vector form deducted from the same matrix A. Each vector element is the value of the associated ORU m, where the index of the vector element indicates the UE k. It is defined as given below. This definition reduces the number of possible combinations for associating UEs with ORUs.

$$\vec{A} = \{m_1, m_2....., m_k\} \tag{9}$$

For given $\mathcal{K}$ and $\mathcal{M}$, there are I such combinations available. Each i-th combination represents a unique A matrix with values of $a_{k,m,i}$ for $k, m \in \mathcal{K}, \mathcal{M}$ as per equation (2). For each of these combinations, SINR $\gamma_{k,m,i}$ is calculated as per equation (4). Based on the range of calculated $\gamma_{k,m,i}$ value in dBm, the modulation and coding scheme (MCS) such as QPSK, 16 QAM, 64 QAM, etc. is selected for each user from the look-up table (LUT). The MCS scheme determines the modulation order $O_{k,i}$ for each user in every i-th combination and the number of bits to be transmitted per resource element $B_{\mathrm{RE}_{k,i}}$.

As per 5G NR configurations, the number of symbols per slot, number of slots per subframe and total number of subcarriers can be determined. Hence, we can calculate bits per PRB for each user as $B_{\mathrm{PRB}_{k,i}}$. The estimate of required number of PRBs at user k is calculated based on discussed parameters. It is defined as fraction of available number of bits in buffer at user k, B_k to the number of bits that can be transmitted at user k in each PRB $B_{\mathrm{PRB}_{k,i}}$,

$$PRB^{\mathrm{required}}_{k,i} = \lceil \frac{B_k}{B_{\mathrm{PRB}_{k,i}}} \rceil \tag{10}$$

With reference to the required number of PRBs, and available bandwidth resources, an estimate of assigned resources is evaluated as $PRB^{\mathrm{assigned}}_{k,i}$. Correspondingly, the buffer size in bits, $C_{m,i}$ along with the number of bits transmitted, $J_{m,i}$ is evaluated as given in equation (7), providing us the metric $\tau_{m,i}$ at each ORU as given in equation (6). The global value for each i is updated as follows:

$$\tau^g_{m,i} = \frac{\frac{C_{m,i}}{J_{m,i}} + \tau^g_{m,t-1}}{2} \tag{11}$$

The estimation of $PRB^{\mathrm{assigned}}_{k,i}$ and $\tau^g_{m,i}$ for each i-th association contributes to the association decision AD as given below:

$$AD_{s,i} = w_{s1} \sum_k \sum_m a_{k,m,i} \cdot PRB^{\mathrm{assigned}}_{k,i} - w_{s2} \sum_m \tau^g_{m,i} \tag{12}$$

The values for w_{s1}, w_{s2} are learned and updated as described in Section VI by intelligent agent for each slice where $s \in \{E, U\}$. The optimal association decision is selected from all the available i combinations in the set with the following equation:

$$AD_{s,opt} = \arg\max_i \left(AD_{s,i}\right) \tag{13}$$

subjected to:

$$\begin{aligned} &C1: d_k \leq \ d_k^{max}, \ k \in \mathcal{K} \\ &C2: R_k \geq R_k^{min}, \ k \in \mathcal{K} \\ &C3: \sum_{m \in \mathcal{M}} \sum_{k \in \mathcal{K}} R_{k,m} \leq N_{fronthaul} \\ &C4: a_{k,m} \in \{0,1\}, \ \forall_{k,m} \in \mathcal{K}, \mathcal{M} \\ &C5: \sum_{k} PRB_k^{\text{assigned}} \leq PRB_s, \ k \in \mathcal{K}_s, \ s \in \{E,U\} \end{aligned} \tag{14}$$

Once $AD_{s,opt}$ is selected, we calculate $\tau_{m,t}^g$ using equation (8) where $\tau_{m,t} = \tau_{m,opt}$. The constraints C1-C5 mentioned in equation (14) provide different checkpoints. Such as, C1 and C2 check if expected user KPI is within defined threshold ranges. C3 ensures that the summation of all assigned and scheduled resources does not exceed fronthaul capacity link. Constraint 4 checks that the association variable $a_{k,m}$ is either 1 or 0 that means the user is only associated with single ORU at a time. Constraint 5 ensures that the PRBs allocated under each slice are less than or equal to the resources available to each slice. Once the association decision is known, MAC-level scheduling is executed. The proposed user association selection scheme is independent of the selected scheduler. In Section VIII, the proportional fair (PF) scheduler is simulated.

B. Low Complexity Intelligent QoS-Aware Resource Allocation (LIQRA)

LIQRA is proposed for the same objective in intra-slice RAN RRM but with reduced computational complexity, which is crucial for some applications. This algorithm concludes association decisions in a different way. The UA decision is evaluated individually for each user k by iterating through all ORUs to find optimal choice $m*$ to serve the user. This reduces the computation complexity at every iteration of DQN. The algorithm aims to make an intelligent association decision based on a combination of signal-to-noise ratio (SNR) and the global ORU metric τ_m^g. At the stage of $m*$ selection, information about other user associations is not available, hence SNR is included instead of SINR. The SNR $\zeta_{k,m}$ for user k with each ORU m for the given frequency flat fading channel $h_{k,m}$ is calculated as follows:

$$\zeta_{k,m} = \frac{P_m \times |h_{k,m}|^2}{\sigma^2} \tag{15}$$

The global ORU metric τ_m^g is calculated as discussed earlier according to equation (8). For each individual user k, the combination of SNR $\zeta_{k,m}$ and metric $\tau_{k,m}^g$ with every available ORU m is calculated. The algorithm evaluates the optimal association decision with the help of the following equation, where for each k, we check,

$$m* = \arg\max_{m}(w_{s1} \cdot \zeta_{k,m} - w_{s2} \cdot \tau_{k,m}^g) \tag{16}$$

The values for w_{s1}, w_{s2} are selected by an intelligent agent for each slice. Once $m*$ for each corresponding k is estimated, we can write the association matrix A as,

$$A = [a_{k,m}]$$
$$\text{Where, } \ a_{k,m} = \begin{cases} 1, & \text{If } m = m* \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

Here, the association matrix A is same as defined in equation (2). Once the association decision is taken and matrix A is estimated, SINR is calculated based on equation (4) and hence, MCS is selected. Further, the required PRBs and other parameters for each user k are calculated as indicated in the earlier algorithm to schedule radio resources for end users.

Algorithm 1: IQRA and LIQRA using DQN for intra-slice resource allocation

```
Initiate network scenarios, intelligent agents, set QoS requirements, update databases, and load models: set number of UE, ORU, and simulation time;
for tti ← 1 to T do
    Update UE position, databases, traffic;
    Initialize replay memory D to capacity N;
    Initialize action-value function Q with random weights θ;
    Initialize target action-value function Q̂ with weights θ⁻ = θ;
    for episode ← 1 to R do
        Select a random action a_t with probability ϵ;
        Otherwise select a_t = arg max_{a∈A} Q(s_t, a_t; θ);
        Update w_s1, w_s2 according to action a_t and associate users with ORUs based on equation (13) for IQRA and equation (16) for LIQRA;
        Schedule the resource, transmit packets and receive KPIs as per Section V-C;
        Receive reward r_t as per equation (24) and observe next state s_{t+1};
        Store transition (s_t, a_t, r_t, s_{t+1}) in D;
        Sample random minibatch of transitions (s_j, a_j, r_j, s_{j+1}) from D;
        Calculate output Q values based on equation (26);
        Update target Q values by equation (27);
        Train DNN using loss function given in equation (25);
        Every C (target network update frequency given in Table I) steps reset Q̂ = Q by updating θ⁻ = θ;
    end
end
```

C. Key Performance Indicators

Once the resources are allocated, the transmission takes place, and experienced key performance indicators such as achieved throughput, delay, and BER are collected. These KPIs are stored, and the mean performance of TTIs for each user is forwarded to DRL. Additional KPIs, such as successful packet transmissions and packet drop rate, for each UE are also evaluated over the defined time interval. The throughput of user k is calculated by multiplying the assigned number of PRBs with bits per PRB selected for transmission in that TTI. The total delay experienced per user is a combination of delays experienced by each packet in queue, transmission,

and processing, as given below, where sub-index l is the packet index for the user.

$$d_{k,l} = \ d_{k,l}^{tx} + d_{k,l}^{que} + d_{k,l}^{processing} \tag{18}$$

We consider the processing delay for each packet to be $2 \times t_{symb}$ and the symbol duration t_{symb} is calculated based on the selected numerology, which is $\mu = 0$ and therefore $t_{symb} = 71.37\mu sec$. The transmission delay $d_{k,l}^{tx}$ for every packet per user can be calculated as the time required to transmit the number of bits in the packet, which can be a multiple of TTI based on how many TTIs the packet requires to transmit successfully. The queuing delay $d_{k,l}^{que}$ is defined as the wait time in the queue or buffer at the ORU before transmission. It is calculated as the difference between the transmission time stamp $t_{k,l}^{current}$ and packet arrival time stamp $t_{k,l}^{arrival}$ for each packet per user. Further, the total delay per user k, $\bar{d}_k$ is calculated as the averaged delay experienced by each packets l from total number of packets per user L_k given as,

$$\bar{d}_k = \ \frac{\sum_{l \ \in L_k} d_{k,l}}{|L_k|} \tag{19}$$

The BER ψ_k for each user k associated with m-th ORU in every t TTI for frequency selective channel is calculated as [32],

$$\psi_{k,m} = \frac{\left(1 - \sqrt{O_{k,m}}\right)}{B_{k,m}} \cdot Q\left(\sqrt{\frac{3 \cdot B_{k,m}}{O_{k,m} - 1} \cdot \gamma k, m}\right) \tag{20}$$

Where $O_{k,m},\ B_{k,m}$ are modulation order and number of bits between k-th user and m-th ORU, respectively. The packet loss rate for every T TTIs is calculated as the ratio of packets discarded in given time interval to the total number of packets transmitted for each user. The packet is discarded if it exceeds the threshold for the maximum delay limit d_k^{max} mentioned in the QoS requirement for each user k. The successful transmission rate for each user for T TTIs is calculated as the ratio of successfully transmitted packets in a given time interval to the total transmitted packet for each user in the same time interval.

VI. Proposed DRL-based intelligent agents

In this study, distinct intelligent agents for the slice types eMBB and URLLC, are proposed. The UA decision is parameterized as discussed in the above section. The DRL technique, namely DQN, is used by the intelligent agent to discover the significance of either the assigned PRB estimate or the SNR metric γ and the ORU metric τ for suggested strategies. Based on the state of the network environment at any given time, the DNN determines the weights for each of these parameters and evaluates the optimum course of action. The DNN trains and learns the weight values for various network environment states. IQRA and LIQRA are then given with these values for w_{s1} and w_{s2} separately for each slice. The formulated solutions in the earlier section take into account the weights and evaluate a decision for ORU selection to serve the corresponding end user. This results in better performance for the KPIs compared to baseline and SOTA solutions. The proposed MDP establishes the significance of the weights for UA parameters using DQN-based intelligent agents. Each intelligent agent learns different values based on the impact of individual experienced user KPIs under each slice category.

A. Markov Decision Process

The MDP for the intelligent agents based on the above formulation is defined with a tuple of $\{S_s, A_s, R_s\ , \Gamma_s\}$ corresponding to state, action, reward, and discount factor, where subscript s indicates slice type for intelligent agents $s \in \{E, U\}$. The state space S_s of the environment includes the current log normalized channel matrix $H_{M\times K}$, number of packets to be transmitted in the buffer per user $\bar{P}_k$, and the distance between each user and ORU in the current TTI t $D_{M\times K}$. The state space definition remains the same for the different slices, as it represents the current state of the environment based on which the actions will be chosen by the intelligent agent. Different intelligent agents learn the importance of the UA parameters, which are specific and valid to the varying traffic flow for the services and user-specific to each slice.

$$\begin{aligned} &S_s = \left\{H_{M\times K}, \bar{P}_k, D_{M\times K}\right\} \ \text{for } k, m \in \mathcal{K}_s, \mathcal{M}_s \\ &s \in \{E, U\} \end{aligned} \tag{21}$$

The action space is combination of weights w_{s1} and w_{s2}, where w_{s1} and w_{s2} are defined as discrete spaces as follows:

$$w_{s1}, w_{s2} \in \left\{ \frac{1}{N}, \frac{2}{N}, \ldots.., \ \frac{N}{N} \right\} \tag{22}$$

Hence, the action space is given as:

$$\begin{aligned} A_s &= (w_{s1}, w_{s2}) \\ |A_s| &= |w_{s1}| \times |w_{s2}| \end{aligned} \tag{23}$$

As defined above, the number of actions is limited, and the agent learns the importance of each parameter to make an optimal decision for each slice separately. Convergence with a large action space is challenging for a number of deep learning methods and MDP designs. Our proposed algorithms in the given formulation use $N = 20$ which resolves this issues.

The intelligent agents will learn the optimal way to associate user-ORU and eventually schedule the resources based on the observation at a given point of time in an environment of wireless communication. The reward function is designed in such a way that the agent receives the reward for the user with a positive value equal to a fraction of exceeding QoS thresholds. Whereas, if any user fails to reach the QoS threshold, it is awarded a negative value equal to the amount of KPI that failed to reach the threshold. This reward is further normalized by the threshold values themselves. The total reward is the accumulated reward of each individual user divided by the total number of users. Both IQRA and LIQRA can use this definition of reward function. The reward function can be expressed as follows:

$$\begin{aligned} &R_s^i\left(S_s, A_s\right) = \frac{\left(\alpha_s \cdot \sum_k \frac{R_{k,i} - R_k^{min}}{R_k^{min}} + \beta_s \cdot \sum_k \frac{d_k^{max} - d_{k,i}}{d_k^{max}}\right)}{K} \\ &k \in \mathcal{K}_s, \ \ s \in \{E, U\} \end{aligned} \tag{24}$$

Where i indicates the episode or run of the DRL algorithm and $\alpha_s = 1 - \beta_s$. The values of α_s and β_s are used to prioritize a specific KPI value according to slice type. Users receive rewards for each agent according to a KPI reward function that is prioritized. For instance, minimum data rate requirements are relatively stricter or given priority over experienced delays for eMBB slice services like social media, messaging, large file downloading, web browsing, etc. Whereas for URLLC slice services, the latency or delay is prioritized as a KPI. We can define slices dedicated to specific service and set the priorities of KPIs in the reward function according to the requirements of that service. This ensures that algorithms achieve optimal system performance by keeping the service requirements in check and the QoS of each user is maintained. The slices achieve the intended and improved system and user performance when the agent learns the ideal weights for various conditions.

The conducted research has analyzed various designs of reward functions to get better convergence and KPIs. We use deep Q-learning methods from DRL algorithms. Here, the Q value, target Q value, and loss function are calculated as given below.

$$L_s = \left[TD_s - \left(Q_s^i\left(s_s, a_s\right)\right)^2\right] \tag{25}$$

$$\begin{aligned} Q_s^i\left(s_s, a_s\right) = \; & Q_s^i\left(s_s, a_s\right) + \Omega_s \\ & \left[R_s^{i+1} + \Gamma_s \max_{a_s \in A_s} Q_s^{i+1}\left(s_s, a_s\right) - {Q^i}_s^2\left(s_s, a_s\right)\right] \\ \text{i.e.} \;\; Q_s^i\left(s_s, a_s\right) = \; & Q_s^i\left(s_s, a_s\right) + \Omega_s \times L_s \end{aligned} \tag{26}$$

Here, $s \in \{E, U\}$ and the values for learning rate Ω_s, discount factor Γ_s as well as other hyperparameters are indicated in the simulation setup. The temporal difference, or target Q values, is calculated as follows:

$$TD_s = {r_s}^{i+1} + \Gamma_s \max_{a_s \in A_s} Q_s^{i+1}\left(s_s, a_s\right) \tag{27}$$

B. Discussion on Complexity Analysis

In this study, the proposed algorithms strategically determine weights of the parameter within the UA decision process, as delineated by equations (12) and (16), respectively. The IQRA evaluates I unique combinations of the matrix A, with each $i - th$ combination undergoing assessment through equation (12) alongside constraints C1-C5 to identify the optimal UA decision ($AD_{s,opt}$) in each iteration of the DQN. Conversely, LIQRA identifies the optimal matrix A for each DQN iteration by applying equations (16) and (17). The complexity of the DQN model inherently increases with an expanded action and state space. However, our proposed methodologies maintain a stable action space, enhancing both reusability and scalability. This adaptability is achieved by enabling the algorithms to discern the significance of global parameters in UA decisions across different slices. Therefore, despite an increase in the number of users—which inherently enlarges the state space and, by extension, the computational and storage demands of the DQN—the acquired knowledge remains applicable for UA decision-making. To mitigate the escalated complexity due to the growing number of users, one approach is to eliminate the users count dependency in the state space configuration. Alternatively, employing separate instances of the same intelligent agent, each powered by a similar DNN, to independently evaluate Q-values presents a viable solution.

VII. Simulation Setup

The designed simulator is 3GPP-standard-compliant for wireless communication models. The network elements are defined and implemented in line with the O-RAN architecture. The simulation set-up is based on ITU Report M (ITU-R M) recommendations for testing and verification. It is implemented in a Python environment. It uses OpenAI-gym, Keras, and Tensorflow modules to implement DQN-based agents. The intelligent DQN agents for both slices are tested and tuned by running multiple combinations of hyperparameters. The final selected values are mentioned in Table I. For each network slice, the number of users requested to be served and their QoS requirements are available in the database at near-RT RIC. We have tested the proposed algorithms for a number of settings of QoS values based on different packet arrival rate (PAR). The simulation set-up includes packet arrival based on stochastic distribution models as well as periodic deterministic models. The variation in traffic volume (high, medium, and low) as well as different traffic generation models are considered to analyze the impact and verify the performance of algorithms. The results presented in next section utilize periodic deterministic traffic for both slices with differing PAR, as listed in Table II. We utilize an infinite buffer queuing model, where packets that reach the maximum delay threshold defined in QoS are prioritized for scheduling within each slice category. The number of packets available to transmit for each user is determined per TTI. Each of these packets has a different time of arrival. For all UEs, the location is updated every TTI based on the mobility model, with a fixed 30 kmph speed for all users.

TABLE I: DQN Agent Hyperparameter

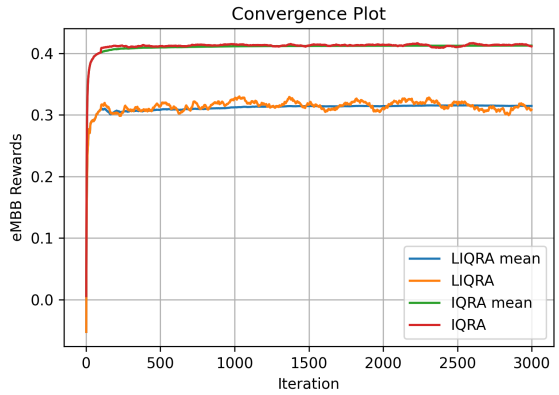
Parameter	Value
Network Architecture	256×256
Learning Rate	$1e^{-3}$
Batch Size	64
Epsilon minimum	0.01
Epsilon decrement	0.99
Target network update frequency	100
Discount factor	0.995
Activation	ReLU
Optimizer	Adam

As we have discussed in the system model, we assume 5G NR numerology 0 for all users, which corresponds to subcarrier spacing of 15 KHz and 12 subcarriers per PRB, hence the bandwidth of a single PRB W_{PRB} is 180 KHz. Here, $W_{m,s}$ bandwidth is allocated to each ORU under each slice category and a ORU can serve users under both eMBB and URLLC slices. The number of PRBs available at each ORU is given with PRB_s as shown in Table II. The symbol duration t_{symb}, number of symbols per slot, and other parameters are calculated based on a selected numerology,

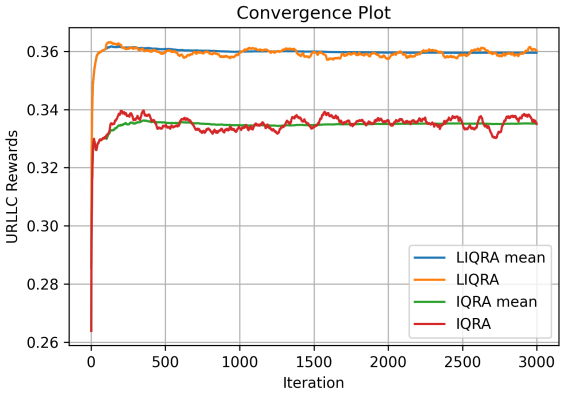
$\mu = 0$ and the corresponding values are indicated in earlier Section V-C. The pathloss and shadowing models are taken as it is given in table A1-2 of ITU-R M.2412-0 [33] and other communication model parameters are listed in Table II. The ITU-R M specification [33] specifies the guidelines for the evaluation of radio interface technologies, specifically for simulation and testing purposes. The ORUs are placed around 50 meters away from each other, with their heights of 3-3.5 meters and maximum power levels of 200 mW (linear power of the ORU for 24.25–27.5 GHz) according to A1-16 and A1-23 of [33] compliant with O-RAN standardization and the ITU framework to support network slicing [34].

TABLE II: Simulation Parameters

Parameter	Value
Transmit power of ORU P_m	200mW
Noise variance σ^2	-146.424 dBm
Noise Figure	5
Boltzmann constant k	1.38e-23 J/K
Temperature in Kelvin (K)	290
Path Loss Exponent	3.5
PRB bandwidth	180 KHz
5G Numerology	$\mu = 0$
Subcarrier Spacing	15KHz
No. of PRBs	PRB_E=32, PRB_U=15
No. of ORUs	4
No. of users: eMBB	5
No. of users: URLLC	5
$\boldsymbol{R}_k^{\min}$ (Mbps)	E = 16, U = 3.8
$\boldsymbol{d}_k^{\max}$ (ms)	E = 10, U = 2
Packet Size (Bytes)	E= 1024, U = 480
Traffic Model	E = periodic deterministic with packet arrival interval 0.5 ms U = periodic deterministic with packet arrival interval 1 ms



(a) eMBB reward performance for DRL Intelligent agent



(b) URLLC reward performance for DRL Intelligent agent

Fig. 3: Reward Performance

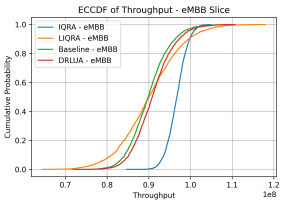
VIII. Results and Analysis

The results presented in this section are based on the simulation parameters for the environment and DQN agents that were previously described. The eMBB reward function, R_E uses $\alpha_E = 0.7$. Whereas the URLLC reward function, R_U uses $\alpha_U = 0.4$. Different values of α_s have been tested for both slices. The configured values generate optimal results based on testing. The DRL agents use the reward function defined in equation (23). The algorithms are simulated for 3000 iterations of DRL and 30 seconds of simulation time. The figure 3 shows the reward values of eMBB and URLLC slices for both the IQRA and LIQRA algorithms. Subfigure 3a and 3b show the value for reward for each run, where LIQRA and IQRA refer to reward values plotted with a sliding window of 100 iterations, and mean refers to the mean reward value. The reward function depends highly on KPI values such as throughput and delay. The instantaneous KPI values are calculated within each TTI, which is 1 ms in duration, and subsequently, individual reward values vary frequently. Therefore, the mean and sliding values of reward have been plotted to analyze the performance of both algorithms. We can see that convergence is achieved within the first 200–300 iterations of DQN. As seen in Subfigure 3a and 3b, IQRA performs better for eMBB, and LIQRA has better reward values with URLLC slice. LIQRA has SNR and τ as UA parameters, whereas IQRA has an estimate of PRB assignment instead of SNR. This is weighted by w_{s1} and w_{s2} for both algorithms. In IQRA, the number of PRBs towards end users is given as one of the parameters to take an UA decision. This associates users with ORUs who can assign more PRBs, achieving higher throughput. The eMBB slice has more stringent throughput thresholds; hence, the more PRBs assigned to end users, the better throughput is achieved, resulting in better reward values. Whereas, for URLLC, latency requirements are more strict, with the first parameter being only SNR helps the algorithm assign users to ORU with better signal quality, giving more importance to the τ metric.

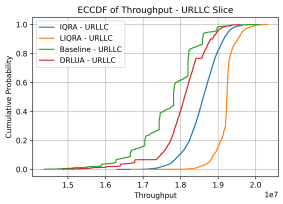
Figure 4 shows the comparison for system performance between the baseline approach, state-of-the-art, referred to as DRLUA now onward, and the proposed algorithms for both slices. The baseline approach uses a simple maxSNR method for user-ORU association, whereas DRLUA uses a number of parallel DNNs along with DQN as proposed in [15]. All of them are implemented on top of proportional fair schedulers. Subfigure 4a and 4b show the Empirical Cumulative Distribution Function (ECCDF) for system throughput over DRL iterations for baseline, IQRA, LIQRA, and DRLUA. Both the proposed algorithms outperform the baseline and DRLUA approaches. The resource allocation to end users is performed separately for each slice category. Subfigure 4c and

4d show the ECCDF for latency for all four approaches in the simulation.

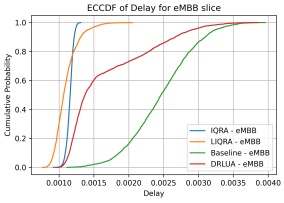
The algorithm makes decisions based on weighted values of UA parameters learned from intelligent agents. This allows the proposed methods to customize the importance of decision-making parameters specific to the current state of the wireless environment. For the eMBB slice, IQRA works better among the 4 approaches, as it considers estimates of PRBs and the τ metric as explained earlier in reward convergence results. In simulation runs, we observe that IQRA selects higher values for w_{s1} compared to w_{s2} in the majority of cases for the eMBB slice. Hence, users are associated with ORUs by giving more importance to the quantity of available resources for transmission, considering the allowed delay threshold for eMBB is larger, up to 10 ms. Whereas for URLLC, based on the network state, the importance lies with τ as the delay thresholds are more strict up to 2 ms. The algorithms allocate more resources to the users based on the QoS thresholds, availability of resources, and state of the network, including the current traffic load. It prioritizes and balances the decisions for UA. Both algorithms have been tested with a number



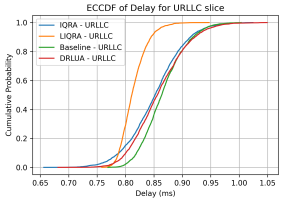
(a) ECCDF of Throughput for eMBB Slice



(b) ECCDF of Throughput for URLLC Slice



(c) ECCDF of Latency for eMBB Slice



(d) ECCDF of Latency for URLLC Slice

Fig. 4: System Performance Comparison

of traffic load variations and traffic models. This study not only captures the impact of proposed IQRA and LIQRA performance under varying traffic volumes but also verifies that the proposed algorithms perform better compared to baseline and DRLUA even under varying traffic loads.

Figure 5 shows KPI performances, namely, throughput, delay, packet drop rate, and bit error rate for each user under both eMBB and URLLC slices, respectively, using the LIQRA algorithm. Subfigure 5a and 5b show experienced throughput for individual users under eMBB and URLLC slices. All users achieve throughput greater than the set QoS thresholds, which are 16 Mbps and 3.8 Mbps for E and U, respectively. Where subfigure 5c shows the delay values for users in eMBB slice. Here, all users experience a delay of less than 4 ms on average for all successfully transmitted packets. Subfigure 5e shows that all users in the eMBB slice have less than 0.5% packet drop rate. The User 3 experienced a 0.9% packet drop rate at the start of simulation, which is reduced to 0.7% as DRL agents make better association decisions. The reason for this relatively high packet drop rate (PDR) compared to other users throughout the simulation is the user's location and movement around the cell edge during the simulation time frame. Every user experiences a PDR below 1%, surpassing the acceptable performance standards for eMBB services as per standards [35], [36]. Subfigure 5d shows all users under URLLC slice experience an average delay less than 1 ms. The QoS threshold for URLLC, d_k^{max}, is 2 ms; hence, users are associated with the respective ORU by giving more weight to τ_m, which ensures the available buffer traffic at each UE can be served within the delay budget. Subfigure 5f shows that the majority of users experience PDR less than $1e^{-4}$. Subfigure 5g and 5h give insights about BER performance for users, which is less than $1e^{-6}$ acceptable for all services considered under both slices.

IX. Conclusion

From the results presented in the above section, we can conclude that the designed intelligent agents and proposed algorithms improved the performance of intra-slice resource allocation compared to baseline and DRLUA techniques. The intelligence agent learns different weights w_{s1} and w_{s2} for each slice for the respective parameters affecting association decisions. From the comparison, we can see the proposed IQRA algorithm is more suitable for eMBB, whereas LIQRA performs better for URLLC slices. IQRA provides 11.5% and 7.42% improvement in throughput for eMBB slices compared to DRLUA and LIQRA, respectively. While LIQRA provides 19.94% and 16.54% improvement in latency compared to baseline and DRLUA approaches, respectively. LIQRA improves latency by achieving a minimum latency value of 45.5% less than compared to the baseline approach. It also improves system throughput for eMBB slices by 6.7%. Whereas, IQRA improves throughput by achieving an increment in maximum throughput value up to 26.7% compared to the baseline approach. It also improves the minimum latency value to less than 8.2% compared to baseline.

Acknowledgment

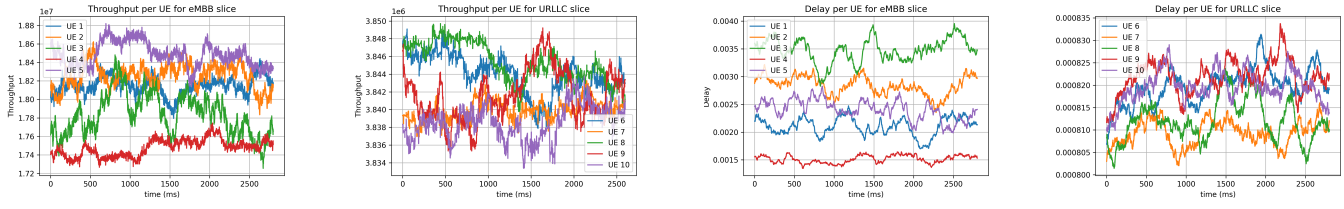
The above work is carried out as part of the MSCA SEMANTIC project with ITN under grant 861165, ADROIT6G (101095363), 6G-BRICKS (101096954) SNS JU projects, RFVOLUTION (PID2021-122247OB-I00) under Spanish Ministry of Science, Innovation and Universities, Generalitat de Catalunya under Grant 2021 SGR 174.

References

[1] S. Zhang, "An Overview of Network Slicing for 5G," *IEEE Wireless Communications*, vol. 26, no. 3, pp. 111-117, Jun. 2019.

[2] R. Wen, G. Feng, J. Tang, T. Q. S. Quek, G. Wang, W. Tan, and S. Qin, "On Robustness of Network Slicing for Next-Generation Mobile Networks," *IEEE Transactions on Communications*, vol. 67, no. 1, pp. 430–444, Jan. 2019.

[3] G. Wang, G. Feng, T. Q. S. Quek, S. Qin, R. Wen, and W. Tan, "Reconfiguration in Network Slicing: Optimizing the Profit and Performance," *IEEE Transactions on Network and Service Management*, vol. 16, no. 2, pp. 591–605, 2019.



(a) User Throughput eMBB (b) User Throughput URLLC (c) User Delay eMBB (d) User Delay URLLC

(e) User PDR eMBB (f) User PDR URLLC (g) User BER eMBB (h) User BER URLLC

Fig. 5: User KPI for eMBB and URLLC slice with IQRA2 Algorithm

[4] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C.g Liang, and D. I. Kim, "Applications of Deep Reinforcement Learning in Communications and Networking: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2664–2732, 2020. DOI: 10.1109/COMST.2020.2988348.

[5] Z. Xu, Y. Wang, J. Tang, J. Wang, and M. C. Gursoy, "A deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs," in *Proc. of IEEE International Conference on Communications (ICC)*, 2017.

[6] F. Wei, G. Feng, Y. Sun, Y. Wang, Y.-C. Liang, "Dynamic Network Slice Reconfiguration by Exploiting Deep Reinforcement Learning," *Proc. of the IEEE International Conference on Communications (ICC)*, Ireland, DOI: 10.1109/ICC40277.2020.9148848, 2020.

[7] Y. Yang, Y. Li, K. Li, S. Zhao, R. Chen, J. Wang, and S. Ci, "Decco: Deep-learning enabled coverage and capacity optimization for massive MIMO systems," *IEEE Access*.

[8] J. Song, Y. Nam, H. Kwon, I. Sim, S. Maeng, and S. Jang, "Adaptive Generalized Proportional Fair Scheduling with Deep Reinforcement Learning," *Samsung Research, Samsung Electronics, Seoul, Republic of Korea*, 2021.

[9] O-RAN Technical Specifications, "O-RAN.WG1.Slicing-Architecture-v07.00: Technical Specification", July 2022.

[10] O-RAN TR, "O-RAN.WG1.Use-Cases-Analysis-Report-v08.00", O-RAN Alliance, 2022.

[11] Md. A. Rahman, S. Mhatre, F. Kooshki, S. Pietrzyk, and A. Flizikowski, "Cooperative radio resource allocation in a wireless communication network and methods for use therewith," US Patent: US20220173849A1, https://patents.google.com/patent/WO2022115869A1/en, 2022.

[12] H. Zhang, H. Zhou and M. Erol-Kantarci, "Team Learning-Based Resource Allocation for Open Radio Access Network (O-RAN)", *IEEE International Conference on Communications*, Seoul, Korea, 2022, pp. 4938-4943, doi: 10.1109/ICC45855.2022.9838763.

[13] H. Zhou, M. Elsayed and M. Erol-Kantarci, "RAN Resource Slicing in 5G Using Multi-Agent Correlated Q-Learning," *IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Helsinki, Finland, 2021, pp. 1179-1184, doi: 10.1109/PIMRC50174.2021.9569358.

[14] P. Tavakoli, A. Pardo, and K. Kormushev, "Action Branching Architectures for Deep Reinforcement Learning," in *AAAI Conference on Artificial Intelligence*, 2018, pp. 4131-4138.

[15] Z. Li, M. Chen, K. Wang, C. Pan, N. Huang and Y. Hu, "Parallel Deep Reinforcement Learning Based Online User Association Optimization in Heterogeneous Networks," *IEEE International Conference on Communications Workshops (ICC Workshops)*, Dublin, Ireland, 2020, pp. 1-6, doi: 10.1109/ICCWorkshops49005.2020.9145209.

[16] S. S. Mhatre, K. Ramantas, R. Qiu and C. Verikoukis, "Slicing-enabled 5G experimentation platform for the Robotics vertical industry," 2021 IEEE International Mediterranean Conference on Communications and Networking (MeditCom), Athens, Greece, 2021, pp. 123-126, doi: 10.1109/MeditCom49071.2021.9647665.

[17] C. Zhang, M. Dong, K. Ota, "Vehicular Multi-slice Optimization in 5G: Dynamic Preference Policy using Reinforcement Learning," *roc. of IEEE Global Communications Conference (GLOBECOM)*, ISBN: 978-1-7281-8298-8, DOI: 10.1109/GLOBECOM42002, 2020.

[18] H. Xiang, M. Peng, Y. Sun, S. Yan, "Mode Selection and Resource Allocation in Sliced Fog Radio Access Networks: A Reinforcement Learning Approach," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4271, April 2020.

[19] G. Sun, Z. T. Gebrekidan, G. O. Boateng, D. Ayepah-Mensah and W. Jiang, "Dynamic Reservation and Deep Reinforcement Learning Based Autonomous Resource Slicing for Virtualized Radio Access Networks," in IEEE Access, vol. 7, pp. 45758-45772, 2019, doi: 10.1109/ACCESS.2019.2909670.

[20] G. Sun, K. Xiong, G. O. Boateng, D. Ayepah-Mensah, G. Liu and W. Jiang, "Autonomous Resource Provisioning and Resource Customization for Mixed Traffics in Virtualized Radio Access Network," in IEEE Systems Journal, vol. 13, no. 3, pp. 2454-2465, Sept. 2019, doi: 10.1109/JSYST.2019.2918005.

[21] F. Mason, G. Nencioni and A. Zanella, "Using Distributed Reinforcement Learning for Resource Orchestration in a Network Slicing Scenario," in IEEE/ACM Transactions on Networking, vol. 31, no. 1, pp. 88-102, Feb. 2023, doi: 10.1109/TNET.2022.3187310.

[22] Y. Cui, H. Shi, R. Wang, P. He, D. Wu and X. Huang, "Multi-Agent Reinforcement Learning for Slicing Resource Allocation in Vehicular Networks," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 2, pp. 2005-2016, Feb. 2024, doi: 10.1109/TITS.2023.3314929.

[23] Y. Abiko, T. Saito, D. Ikeda, K. Ohta, T. Mizuno and H. Mineno, "Flexible Resource Block Allocation to Multiple Slices for Radio Access Network Slicing Using Deep Reinforcement Learning," in IEEE Access, vol. 8, pp. 68183-68198, 2020, doi: 10.1109/ACCESS.2020.2986050.

[24] Y. Azimi, S. Yousefi, H. Kalbkhani and T. Kunz, "Energy-Efficient Deep Reinforcement Learning Assisted Resource Allocation for 5G-RAN Slicing," in IEEE Transactions on Vehicular Technology, vol. 71, no. 1, pp. 856-871, Jan. 2022, doi: 10.1109/TVT.2021.3128513.

[25] W. F. Villota-Jacome, O. M. C. Rendon and N. L. S. da Fonseca, "Admission Control for 5G Core Network Slicing Based on Deep Reinforcement Learning," in IEEE Systems Journal, vol. 16, no. 3, pp. 4686-4697, Sept. 2022, doi: 10.1109/JSYST.2022.3172658.

[26] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "ColORAN: Developing machine learning-based xApps for open RAN closed-loop control on programmable experimental platforms," *IEEE Transactions on Mobile Computing*, 2022.

[27] M. Yan, G. Feng, J. Zhou, Y. Sun, and Y.-C. Liang, "Intelligent resource scheduling for 5G radio access network slicing," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7691–7703, 2019.

[28] J. Mei, X. Wang, K. Zheng, G. Boudreau, A. B. Sediq, and H. Abou-Zeid, "Intelligent radio access network slicing for service provisioning in 6G: A hierarchical deep reinforcement learning approach," *IEEE Transactions on Communications*, vol. 69, no. 9, pp. 6063–6078, 2021.

[29] B. Khodapanah, A. Awada, I. Viering, A. Noll Barreto, M. Simsek, and G. Fettweis, "Framework for slice-aware radio resource management

utilizing artificial neural networks," *IEEE Access*, vol. 8, pp. 174972-174987, 2020.
[30] R. Schmidt, C.-Y. Chang, and N. Nikaein, "Slice scheduling with QoS-guarantee towards 5G," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, 2019, pp. 1–7.
[31] O-RAN Alliance, "O-RAN Architecture Specification," Version 1.0.0, 8 July 2019.
[32] M. K. Simon and M. S. Alouini, "Digital Communication over Fading Channels: A Unified Approach to Performance Analysis John Wiley & Sons," 2005.
[33] ITU-R M.2412-0 (10/2017) "Guidelines for evaluation of radio interface technologies for IMT-2020".
[34] ITU-T Recommendation Y.3112 "Framework for the support of network slicing in the IMT-2020 network".
[35] ITU-R M.2083-0, "IMT Vision: Framework and overall objectives of the future development of IMT for 2020 and beyond," September 2015.
[36] 3GPP TS 22.261, "Service requirements for the 5G system; Stage ,1" December 2022.

Suvidha Mhatre received her B.Eng. in Electronics and Telecommunications from Mumbai University, India, and both her M.Sc. in Communication Engineering from Hochschule Darmstadt, Germany, and M.Tech. from VIT, Vellore, India, in 2016 and 2019, respectively. She is currently pursuing a Ph.D. at the Polytechnic University of Catalonia (UPC) and works as a RAN Engineer at Sateliot, Barcelona, Spain. Previously, she was an early-stage researcher at Iquadrat Informatica S.L., Barcelona, under the MSCA SEMANTIC project funded by the European Commission. Suvidha has contributed to multiple European H2020 projects, including EuWireless, 5GESSENCE, and MORPHEMIC. She holds one patent (US20220173849A1) and has published four conference papers. Her research focuses on resource allocation and management using AI/ML techniques for network slicing in O-RAN-based 6G networks. e-mail: (suvidha.sudhakar.mhatre@upc.edu).

Ferran Adelantado received PhD degree in Telecommunications from UPC in 2007. He is Associate Professor at UOC and an Adjunct Associate Professor at Universitat de Barcelona (UB). His research interests are Wireless Networks, particularly Virtualization and Slicing in 5G/B5G Networks, LPWAN, and IoT technologies. e-mail: (ferranadelantado@uoc.edu).

Kostas Ramantas has received the Diploma of Computer Engineering, the MSc degree in Computer Science and the PhD degree from the University of Patras, Greece, in 2006, 2008 and 2012 respectively. He has been the recipient of two national scholarships. In June 2013, he joined IQUADRAT as a senior researcher and has co-supervised 2 PhD student, 3 ESRs in ITNs and many secondments in the framework of RISE projects (e.g., CASPER, WATER4CITIES) and ITN projects (e.g., Spotlight, 5GAura). He was involved in multiple E.C. funded projects as a WPL, TM, and PC. He has published more than 35 journal and conference papers. His research interests are in modelling and simulation of network protocols, and scheduling algorithms for QoS provisioning.). email: (kramantas@iquadrat.com).

Christos Verikoukis Christos Verikoukis received the Ph.D. degree from the Technical University of Catalonia (UPC), Barcelona, Spain, in 2000. He is currently the Scientific Director with Iquadrat Informatica, and an Associate Professor with the Univerisity of Patras. He has authored 165 journal articles and over 200 conference papers. He is the coauthor of three books, 14 chapters in other books, and two patents. He has participated in more than 60 competitive projects and he has coordinated 15 of them. He has served as the Principal Investigator for national projects in Greece and Spain. He is currently the IEEE ComSoc GITC member and the Editor-in-Chief of the IEEE NETWORKING LETTERS. e-mail: (cveri@ceid.upatras.gr).