

DRL-based Joint Resource Scheduling of eMBB and URLLC in O-RAN

Rana M. Sohaib[†], Syed Tariq Shah^{*}, Oluwakayode Onireti[†], Yusuf Sambo[†], Qammer H. Abbasi[†], M. A. Imran[†]

[†]James Watt School of Engineering, University of Glasgow, Glasgow, UK

Emails: {RanaMuhammad.Sohaib, Oluwakayode.Onireti, Yusuf.Sambo, Qammer.Abbasi, Muhammad.Imran}@glasgow.ac.uk

^{*}School of Computer Science & Electronic Engineering, University of Essex, Colchester, UK

Email: Syed.Shah@essex.ac.uk

Abstract—This work addresses resource allocation challenges in multi-cell wireless systems catering to enhanced Mobile Broadband (eMBB) and Ultra-Reliable Low Latency Communications (URLLC) users. We present a distributed learning framework tailored to O-RAN network architectures. Leveraging a Thompson sampling-based Deep Reinforcement Learning (DRL) algorithm, our approach provides real-time resource allocation decisions, aligning with evolving network structures. The proposed approach facilitates online decision-making for resource allocation by deploying trained execution agents at Near-Real Time Radio Access Network Intelligent Controllers (Near-RT RICs) located at network edges. Simulation results demonstrate the algorithm’s effectiveness in meeting Quality of Service (QoS) requirements for both eMBB and URLLC users, offering insights into optimising resource utilisation in dynamic wireless environments.

Index Terms—eMBB, DRL, URLLC, Resource Allocation, O-RAN.

I. INTRODUCTION

Traditional radio access networks (RAN) often rely on a single-vendor model, presenting a monolithic, black-box solution that hinders flexibility and innovation. In contrast, open RAN (O-RAN) disrupts this paradigm by separating hardware from software and promoting open interfaces [1]. This allows network operators to customise their infrastructure according to the requirements. The architecture of O-RAN incorporates several foundational elements, namely the Radio Unit (RU), Distributed Unit (DU), Central Unit (CU), and RAN Intelligent Controller (RIC). RIC is an innovative network component that introduces new services and optimisation capabilities in the network. More specifically, RIC encompasses both near real-time and non-real-time variants, fostering programmable-based functions that significantly enhance network flexibility and efficiency. By integrating embedded AI capabilities, RIC can dynamically adapt radio resource [2], mobility, and spectrum management operations, which includes admission control, radio resource allocation and scheduling [3], power allocation, and radio link management, to meet the specific requirements of applications. This adaptability proves especially crucial in Beyond 5G (B5G) networks, which cater to a diverse range of vertical industries, by ensuring network operations are finely tuned to serve varying demands efficiently [4].

In the context of O-RAN in B5G networks, two pivotal service categories are enhanced Mobile Broadband (eMBB) and Ultra-Reliable low latency communications (URLLC). The

eMBB targets high data rate applications, aiming for improved data rates and reliability, with a focus on supporting a packet error rate (PER) target of 10^{-3} . URLLC, in contrast, is engineered for critical applications requiring immediate data transmission with minimal delay, such as autonomous driving and remote surgery. It prioritises ultra-high reliability (PER around 10^{-5}) and low latency, necessitating significant physical and MAC layer adjustments in 5G NR to increase throughput capacity and reduce transmission time intervals, control signalling overhead and outage probability [5]. Accommodating these requirements is crucial for O-RAN’s adaptability and performance in diverse B5G applications. The coexistence of eMBB and URLLC services within the same network infrastructure necessitates innovative radio resource management (RRM) strategies to satisfy the contrasting requirements of high throughput for eMBB and low latency and high reliability for URLLC [6]. Conventional optimisation approaches face limitations in providing real-time solutions for the coexistence of eMBB and URLLC [7]. Deep Reinforcement Learning (DRL) is essential in this context as it enables systems to learn and adapt in real time, leveraging experience to make decisions in dynamic environments. The authors in [8] present a dual-strategy approach integrating risk-sensitive optimisation and DRL to optimise resource allocation for eMBB and URLLC coexistence in 5G. Their proposed approach improves the reliability of URLLC while protecting the eMBB reliability. The superposition scheme’s efficacy for facilitating uplink communication among eMBB, mMTC, and URLLC service classes is addressed in [9], where the authors highlight the advantages of employing superposition in this context. Existing DRL-based solutions for eMBB and URLLC co-existence employed the ϵ -greedy approach for action policy, primarily focused on exploitation by consistently selecting the action estimated to yield the highest potential reward. This approach may result in sub-optimal solutions or being trapped in local optima. Unlike the above-mentioned studies, which mainly focus on eMBB and URLLC co-existence in a generic ideal RAN scenario, this paper focuses on the non-ideal front haul (FH) scenario. In this paper, we employ the Thompson sampling-based DRL approach to balance exploration and exploitation, addressing the co-existence challenges between eMBB and URLLC in O-RAN.

II. SYSTEM MODEL

We model a wireless network that delivers two distinct services, namely, eMBB and URLLC, as illustrated in Fig. 1, where we strategically placed several edge cloud servers at the near-RT-RIC, establishing connections with a regional cloud server at non-RT-RIC. According to [1], Non-RT RIC is situated at the regional cloud server, while the Near-RT RICs are implemented on the edge cloud servers. In the considered model, we have multiple small cells, and set of all BS is defined as $\mathcal{K} = \{1, \dots, K\}$, where a BS $k \in \mathcal{K}$ covers a set of eMBB users $\mathcal{V}_k^e = \{1, \dots, V_k^e\}$, and URLLC users $\mathcal{V}_k^u = \{1, \dots, V_k^u\}$ present in the network. Moreover, each BS is linked with a single-edge cloud server. The radio resources in 5G-NR can be depicted in both the frequency and time domains. These domains are further subdivided into a set of M radio resources, commonly known as RB. Each RB is characterised by a bandwidth denoted as B . Every time slot is subdivided into L mini-slots. In general, the eMBB service extends over several TTIs to improve spectral efficiency (SE). However, due to the strict latency requirements, the incoming URLLC traffic cannot be delayed. Puncturing eMBB slots are employed to address URLLC's stringent latency and reliability demands. By puncturing eMBB slots, dedicated resources can be allocated to URLLC traffic, ensuring timely and reliable delivery of critical information. The URLLC service is scheduled with a brief TTI of 0.5 ms, while a longer 1 ms duration is allocated for the eMBB service. The immediate scheduling of URLLC transmission, which entails disrupting eMBB traffic, can exert a notable influence on the system's capacity and reliability. This, in turn, may result in a decline in the eMBB service's performance. Therefore, an appropriate framework is required to fulfil the QoS requirements.

eMBB rate: The immediate scheduling of URLLC transmission can degrade the rate of the eMBB service. A puncturing decision variable is introduced as follows:

$$\eta_{m,l}^{k,v}(t) = \begin{cases} 1, & \text{if the } l^{\text{th}} \text{ mini-slot is punctured by the} \\ & v^{\text{th}} \text{ URLLC user, } \forall m \in \mathcal{M}, k \in \mathcal{K} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

The signal-to-noise-interference-ratio (SINR) of the eMBB user v can be determined as follows

$$\chi_{k,m}^{e,v}(t) = \frac{p_{k,m}^{e,v}(t)g_{k,m}^{e,v}(t)}{\sum_{\substack{k' \in \mathcal{K} \\ k' \neq k}} \underbrace{p_{k',m}^{e,v}(t)g_{k',m}^{e,v}(t)}_{\text{eMBB interference}} + \sum_{\substack{k' \in \mathcal{K} \\ k' \neq k}} \underbrace{p_{k',m}^{u,v}(t)g_{k',m}^{u,v}(t)}_{\text{URLLC interference}} + \sigma^2}, \quad (2)$$

where $p_{k,m}^{e,v}(t)$, and $g_{k,m}^{e,v}(t)$ refers to the transmitted power and channel gain, respectively, of eMBB user v of BS k over RB m , and σ^2 denotes the noise power. The achievable rate of an eMBB user v of BS k on RB m at time slot t can be computed as

$$r_{k,m}^{e,v}(t) = B \left(1 - \frac{\sum_{l=1}^L \eta_{m,l}^{k,v}(t)}{L} \right) \log_2 (1 + \chi_{k,m}^{e,v}(t)), \quad (3)$$

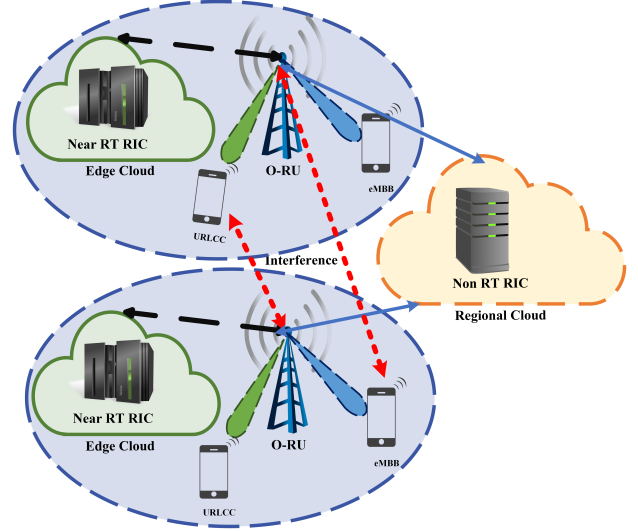


Fig. 1: Illustration of System model of O-RAN

where the expression $\frac{\sum_{l=1}^L \eta_{m,l}^{k,v}(t)}{L}$ indicates the degradation of eMBB rate caused by puncturing. We make the assumption that each BS reserves an RB for a single user. We define the RB allocation binary decision variable as $(\beta_{v,m}^k(t))$, that takes the value of 1 if RB (m) of BS (k) is assigned to the eMBB user (v), for all (k) in the set (\mathcal{K}). It is set to 0 in all other instances. Thus, the total sum rate obtained by the eMBB user v can be computed as

$$r_{k,v}^e(t) = \sum_{m \in \mathcal{M}} \beta_{v,m}^k(t) r_{k,m}^{e,v}(t). \quad (4)$$

URLLC rate: To ensure minimal transmission delay, it is necessary to limit the blocklength in URLLC. Shannon's capacity theorem is relevant when dealing with an infinite blocklength. A study in [10] delves into analysing resource management challenges specific to URLLC services, considering the achievable data rate within the finite blocklength regime. The achievable rate of URLLC for finite blocklength can be computed as

$$r_{k,m}^{u,v}(t) = \sum_{m \in \mathcal{M}} B_m \left(\frac{\sum_{l=1}^L \eta_{m,l}^{k,v}(t)}{L} \right) \left[\log_2 (1 + \chi_{k,m}^{u,v}(t)) - \sqrt{\frac{Y_{k,m}^{u,v}}{W_{k,m}^{u,v}(t)}} \cdot Q^{-1}(x)} \right], \quad (5)$$

where $W_{k,m}^{u,v}(t)$ refers to the number of symbols, and $\chi_{k,m}^{u,v}(t)$ denote the SINR of URLLC user, formulated as

$$\chi_{k,m}^{u,v}(t) = \frac{p_{k,m}^{u,v}(t)g_{k,m}^{u,v}(t)}{\sum_{\substack{k' \in \mathcal{K} \\ k' \neq k}} \underbrace{p_{k',m}^{u,v}(t)g_{k',m}^{u,v}(t)}_{\text{URLLC interference}} + \sum_{\substack{k' \in \mathcal{K} \\ k' \neq k}} \underbrace{p_{k',m}^{e,v}(t)g_{k',m}^{e,v}(t)}_{\text{eMBB interference}} + \sigma^2}. \quad (6)$$

Here, $Y_{k,m}^{u,v} = 1 - \frac{1}{(1 + \chi_{k,m}^{u,v}(t))^2}$ refers to the dispersion of the channel.

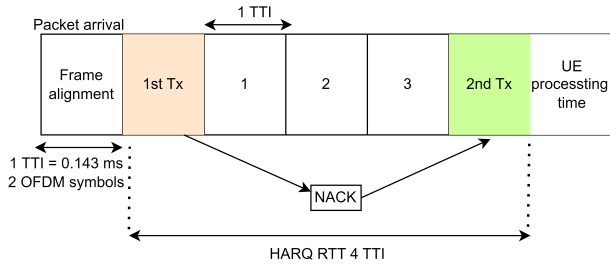


Fig. 2: URLLC 2 symbols HARQ RTT

A. URLLC frame structure

The URLLC frame structure in 5G is designed to meet stringent latency requirements for mission-critical applications. The 3GPP has acknowledged the diverse needs of users and adopted different TTI durations, such as 0.5 ms slots composed of 7 OFDM symbols or 0.143 ms mini-slots composed of 2 OFDM symbols, are considered to provide greater flexibility and adaptability to various use cases [11]. In this work, we consider a mini-slot comprised of 2 OFDM symbols. Since we consider non-ideal FH, it's crucial to address the challenge posed by hybrid automatic repeat request (HARQ) round-trip time (RTT) and the associated latency target of 1 ms for URLLC. The standard 8×0.143 ms HARQ RTT exceeds the specified 1 ms latency target, which may impact reliability. To mitigate this, the study explores the reduction of HARQ RTT to 4 TTIs as shown in Fig. 2, allowing room for one HARQ retransmission. Considering one HARQ retransmission, the maximum allowable latency is reduced to 0.86 ms (6×0.143 ms), which meets the strict latency demands of URLLC service. This calculation does not include queuing delay.

III. PROBLEM FORMULATION

Organising URLLC traffic alongside eMBB users experiencing varying channel quality imposes additional strain on eMBB transmissions. We assume that the URLLC users generate small packet fragments. The rate at which packets arrive in mini-slot l , where l belongs to the set $\mathcal{L} = \{1, \dots, l, \dots, L\}$, during TTI t , follows a Poisson point process (PPP) distribution and is represented by $\phi_l(t)$. Moreover, the total number of URLLC packets received during the TTI t can be calculated as $\phi(t) = \sum_{l \in \mathcal{L}} \phi_l(t)$. Based on $\phi(t)$, the reliability of URLLC service can be estimated as

$$\Pr \left[r_{k,m}^{u,v}(t) \leq \rho \phi(t) \right] \leq \Psi_u, \quad \forall k \in K \quad (7)$$

where ρ denotes the packet size associated with the URLLC service. More specifically, the expression states that the probability of the achievable data rate of a URLLC user at a given time slot t is less than or equal to the product of URLLC packet size. The total number of arrived URLLC packets should be less than or equal to a small positive value, denoted Ψ_u . In practical terms, this ensures that the system can reliably handle the communication requirements of the URLLC traffic and also ensures that the probability of failure (not meeting the

specified data rate constraint) remains below a predefined limit. As mentioned earlier, the proposed system model considers non-ideal FH, and in order to incorporate the round-trip delay of HARQ retransmission into the outage probability of URLLC service, we have to modify the reliability equation given at (7). In other words, the HARQ retransmissions introduce an additional delay, and the outage probability should consider the reliability of both the initial transmission and potential retransmissions. Therefore, the updated reliability considering non-ideal FH and HARQ can be expressed as

$$\Pr \left[r_{k,m}^{u,v}(t + \Lambda_{RTT}) \leq \rho \phi(t) \right] \leq \Psi_u, \quad \forall k \in K, \quad (8)$$

where Λ_{RTT} denotes the round-trip delay of HARQ retransmission in TTI units. This modification reflects the fact that we are considering the achievable data rates for URLLC users at time $(t + \Lambda_{RTT})$ instead of time t . The modification to $r_{k,m}^{u,v}(t + \Lambda_{RTT})$ is meant to account for the time at which we assess the reliability of the system. The success probability for each HARQ transmission can be modelled based on the BER [12], [13]. At time $(t + \Lambda_{RTT})$, we estimate the probability of successful transmission by considering the CDF of the BER. A Bernoulli experiment is run to determine whether the Transport Block (TB) transmitted to each user is successful or not based on the estimated success probabilities. This modelling approach allows us to evaluate the impact of delays on the success or failure of HARQ transmissions and, subsequently, on the overall data rate for URLLC services. Consequently, we formulate an objective function aimed at enhancing the allocation of eMBB resource blocks, optimising transmission power for eMBB users, and refining the scheduling strategy for URLLC users in the following manner.

$$\mathbf{P} : \max_{\beta, P, \eta} \left\{ \sum_{v \in \mathcal{V}_k^e} r_{k,v}^e \right\} \quad (9a)$$

$$\text{subject to } \sum_{v \in \mathcal{V}_k^e} \beta_{v,m}^k(t) \leq 1, \quad \forall m \in \mathcal{M}, k \in \mathcal{K} \quad (9b)$$

$$\sum_{v \in \mathcal{V}_k^u} \eta_{m,l}^{k,v}(t) \leq 1, \quad \forall m \in \mathcal{M}, k \in \mathcal{K} \quad (9c)$$

$$\sum_{l \in \mathcal{L}} \eta_{m,l}^{k,v}(t) \leq L, \quad \forall m \in \mathcal{M}, k \in \mathcal{K} \quad (9d)$$

$$\Pr \left[r_{k,m}^{u,v}(t + \Lambda_{RTT}) \leq \rho \phi(t) \right] \leq \Psi_u, \quad \forall k \in K \quad (9e)$$

$$\sum_{v \in \mathcal{V}_k^e} \sum_{m \in \mathcal{M}} p_{k,m}^{e,v}(t) \leq P_{max}, \quad \forall k \in \mathcal{K} \quad (9f)$$

$$p_{k,m}^{e,v}(t) \geq 0, \quad \forall v \in \mathcal{V}^e, m \in \mathcal{M} \quad (9g)$$

$$\beta_{v,m}^k(t) \in \{0, 1\}, \quad \forall v \in \mathcal{V}^e, m \in \mathcal{M} \quad (9h)$$

$$\eta_{m,l}^{k,v}(t) \in \{0, 1\}, \quad \forall v \in \mathcal{V}^u, m \in \mathcal{M} \quad (9i)$$

where (9a) refers to the objective where we look to maximize the rate of eMBB service. Constraint (9b) defines the eMBB RB allocation constraint, ensuring that only one user is linked to a RB. Whereas (9c) ensures that no more than one URLLC

user can puncture a specific mini-slot on a particular RB at a given time slot. Constraint (9d) defines that the punctured mini-slots should be less than or equal to the total number of mini-slots. Constraint (9e) defines the URLLC reliability constraint. Constraints (9f) and (9g) refer to the eMBB power allocation constraints. Likewise, the constraints (9h) and (9i) specify the limitations on resource allocation visibility.

IV. PROPOSED DECENTRALIZED DRL-BASED FRAMEWORK

The DRL agents are positioned at the edges of the network, while a core training module operates from the regional cloud server for simplicity of implementation and enhanced stability. The central cloud server conducts training globally, utilising experiences accumulated from all agents. This method facilitates quicker convergence and improves overall performance. Despite sharing common learning parameters from the central cloud server, each agent independently takes decisions, with no awareness of the decisions made by others. Solving the mixed-integer programming challenge presented in (9) is typically classified as NP-hard. Furthermore, the scheduling parameter for URLLC is intricately linked with the RB and power allocation parameters, thereby increasing the intricacy of the optimisation challenge. We present a DRL-based framework to address the optimisation problem presented in (9), where we model it as a Markov decision process (MDP) for K agents. The agent selects an action from its action space based on the observations associated with the state, and the action is guided by the policy π .

State space: The set of state space can be defined as $S = \{S_1, S_2, \dots, S_k, \dots, S_K\}$. We assume that each edge cloud server acting as an agent exclusively gets its individual state, specifically the information of users within the same cell. This approach is adopted to minimise the overhead resulting from information exchange across cells. It can be represented as $s_k(t) = \{g_k^e(t), g_k^u(t), \phi_k(t), V_k^e, V_k^u\}$, where it comprises the channel information of both eMBB and URLLC users, along with the traffic information at time slot t .

Action: The set of action space can be defined as $A = \{A_1, A_2, \dots, A_k, \dots, A_K\}$. Each agent takes action on selecting eMBB RB β , eMBB power allocation P , and URLLC scheduling η .

Reward: We design the global reward function by considering the requirements of URLLC service. The reward function can be formulated as

$$r(t) = \underbrace{\left(\sum_{v \in \mathcal{V}_k^e} r_{k,v}^e \right)}_I - \Phi(t) \underbrace{\left(\sum_{v \in \mathcal{V}_k^u} r_{k,m}^{u,v} (t + \Lambda_{RTT}) - \rho \phi(t) \right)}_{II} \quad (10)$$

where we introduce the time-varying weight coefficient $\Phi(t)$ to ensure the URLLC reliability constraint. It can be updated as

$$\Phi(t+1) = \max \{ \Phi(t) + \Psi(t) - \Psi_u, 0 \}, \quad (11)$$

where $\Psi(t)$ refers to the achieved outage probability as mentioned in (8). In the first part, we aim to maximise the rate

of eMBB users, while the second part represents the constraint for URLLC. The objective of the agent is to choose an optimal policy (π) by balancing the trade-off between low outage probability and high eMBB data rate.

A. Thompson sampling-based DDPG framework

The DDPG agent consists of two main components: an actor and a critic. The actor is responsible for defining a policy function, which maps states to corresponding actions. The policy, denoted as $\pi = \pi_a^L$, can be characterised by the network state observed by the agent. The agent then performs actions corresponding to the number of punctured mini-slots L from each allocated RB. Subsequently, the reward is computed by the agent using equation (10), considering the decisions made. The updated state information of the network is then provided to the agent. The objective of the actor is to acquire a strategy that maximises the cumulative discounted reward. It can be denoted as $J = \sum_{t=1}^T \mu r(t)(s(t), a(t))$, where μ refers to the discount factor. To assess the effectiveness of the actor's chosen actions, the critic employs a state-action value function denoted as $Q(s(t), a(t))$. This function calculates the cumulative return over the long term by considering the immediate reward ($r(t)(s(t), a(t))$) and the estimated future returns ($\pi Q(s(t+1), a(t+1))$) resulting from taking action at in-state $s(t)$ according to the current policy. It utilises a Deep DNN with parameters Θ^Q , denoted as $Q(s(t), a(t)|\Theta^Q)$, to compute the Q-value for a given state-action pair ($s(t), a(t)$). To ensure a stable learning process, both the critic and actor utilise duplicates of their DNNs. The critic employs Q-networks, denoted as $\hat{Q}(s(t+1), a(t+1)|\Theta^{\hat{Q}})$, parameterized by $\Theta^{\hat{Q}}$, while the actor utilizes a policy network $\hat{\pi}(s(t+1)|\Theta^{\hat{\pi}})$, parameterized by $\Theta^{\hat{\pi}}$. The critic undergoes training to enhance the optimisation of Θ^Q and $\Theta^{\hat{Q}}$, ensuring accurate computation of the long-term return. The update of Θ^Q using gradient descent involves minimising the loss function as follows

$$\Upsilon(\Theta^Q) = (Q(s(t), a(t)|\Theta^Q) - \zeta(t))^2, \quad (12)$$

where $\zeta(t) = r(s(t), a(t)) + \mu \hat{Q}(s(t+1), a(t+1)|\Theta^{\hat{Q}})$ refers to the target action-value. The parameter $\Theta^{\hat{Q}}$ is updated as follows $\Theta^{\hat{Q}} = \varkappa \Theta^Q + (1 - \varkappa) \Theta^{\hat{Q}}$. The actor undergoes training to discover the most effective policy that maximizes the objective function J , essentially identifying the optimal values for Θ^π and $\Theta^{\hat{\pi}}$. The parameter Θ^π is updated using a gradient ascent algorithm along with the corresponding gradient as follows

$$\nabla_{\Theta^\pi} J = \nabla_\pi Q(s(t), a(t)|\Theta^Q) \nabla_{\Theta^\pi} \pi(s(t)|\Theta^\pi), \quad (13)$$

where $\nabla_\pi Q(s(t), a(t)|\Theta^Q)$ denotes the policy evaluation. In this paper, we use Thompson sampling to balance the exploration and exploitation. When an edge agent selects a hypothesis before interacting with the environment, uncertainty arises. Hence, achieving a balance between exploration and exploitation becomes imperative to identify the optimal policy. Each hypothesis, associated with parameter θ , undergoes training on a relevant subset D to shape the distribution over the posterior knowledge $\hat{P}(\theta|D)$. Approximating $\hat{P}(\theta|D)$ entails

capturing the posterior distribution, a challenge arising from the difficulty of precisely determining values. Instead, a variety of hypotheses is extracted, and their distribution is dictated by the prior distribution $P(\theta)$. We consider a set of actors $A \in \mathcal{A}$, each defined by specific parameters θ_i , where $i \in \mathcal{A}$. These parameters are sampled from the prior distribution $P(\theta)$. The hypothesis in question encapsulates this ensemble of actors, giving rise to the distribution $\hat{P}(\theta|D)$, which is influenced by the original distribution $P(\theta)$. To maintain the distribution, each actor undergoes personalised updates contingent on its specific sub-sample, as indicated by the mask value derived from a Bernoulli distribution [14]. This binary mask value serves to identify whether a given sample is part of the actor's subset or not. The implementation of Thompson sampling entails a stochastic selection from the ensemble of actors. This random selection process informs the formation of a hypothesis aligned with the posterior distribution. This adaptive strategy plays a pivotal role in striking a nuanced balance between exploration and exploitation, contributing to the formulation of an optimal policy for the eMBB resource block, power allocation and URLLC scheduling. Therefore, the target function is updated as follows

$$\zeta(t) = r(s(t), a(t)) + \mu \max_{i \in \mathcal{A}} \left[\hat{Q}_i(s(t+1), a(t+1) | \Theta^{\hat{Q}}) \right]. \quad (14)$$

A neural network model undergoes training offline at the Non-RT RIC. This controller is situated in the regional cloud server and utilizes data gathered from all agents. After the training process, the model is communicated to the DRL agents located at the Near-RT RICs on edge cloud servers. The training objective for the global model is to maximize the predefined global reward function outlined in (10).

TABLE I: Simulation Parameters

| Parameter | Value |
|-------------------------------|-----------------------------|
| Frame duration | 10 ms |
| No. of mini-slots in each TTI | 7 |
| sub-carrier spacing | 15 KHz |
| No. of OFDM symbols/TTI | 14 |
| OFDM symbols/mini-slot | 2 |
| Bandwidth | 20 MHz |
| URLLC packets length | 32 Bytes |
| RB Bandwidth | 180 kHz |
| Transmit power | 38 dBm |
| Pathloss Model | $120.8 + 37.5 \log_{10}(d)$ |
| Actor learning rate | 10^{-5} |
| Critic learning rate | 10^{-3} |

V. PERFORMANCE EVALUATION

We consider four BSs in our network model, where each BS provides coverage to an area of 250 square meters and handles eMBB users as well as URLLC users. The eMBB users generate constant full-buffer traffic, while URLLC users produce Poisson traffic with an arrival rate denoted as ϕ . This stochastic nature reflects the sporadic demands typical in URLLC applications, where real-time responsiveness and reliability are paramount. Within this framework, the network

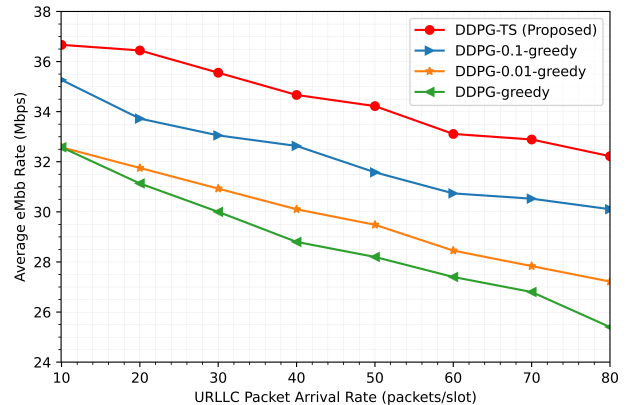
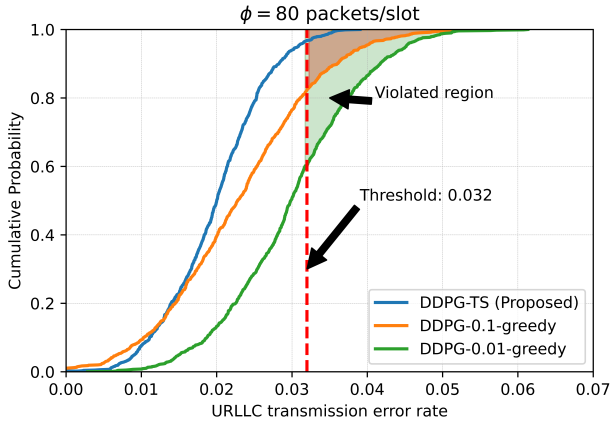


Fig. 3: Impact of URLLC traffic

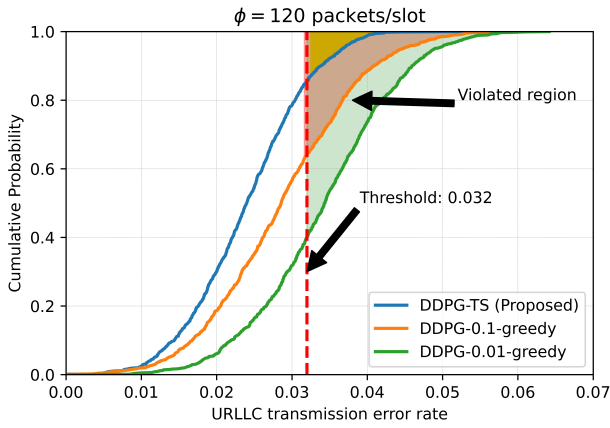
must balance the differing requirements of eMBB and URLLC traffic, optimizing resource allocation and scheduling algorithm to meet the diverse needs of both user types. We define the simulation parameters in Table I. The proposed approach undergoes training by varying the network environment.

Impact on eMBB rate: We investigate the impact of puncturing on the eMBB data rate and conduct a comparative analysis with alternative methods across different loads of incoming URLLC traffic. In Fig. 3, we illustrate the impact of incoming URLLC traffic on the eMBB rate. This influence arises from prioritizing URLLC service users, leading to the allocation of additional radio resources to meet the stringent latency requirements associated with URLLC services. Our proposed method strikes an effective balance between exploration and exploitation, allowing the algorithm to explore the solution space effectively while exploiting the knowledge gained. On the other hand, greedy-based methods where ϵ value is varied often prioritize the exploitation of known strategies rather than exploring new possibilities. This can lead to sub-optimal solutions when the environment is complex. However, as the influx of URLLC traffic intensifies, the average data rate begins to decline, yet the proposed approach consistently maintains a higher level compared to greedy approaches.

URLLC reliability: In Fig. 3, we examine the performance in terms of URLLC reliability by plotting the CDF of the transmission error probability. Fig. 4a shows that in more than 98% of instances, the proposed approach ensures that the transmission error rate remains within acceptable limits. When ϕ is 80 packets per time-slot, the proposed method manages to uphold a high level of accuracy and reliability in transmitting URLLC packets. However, it is crucial to note that in Fig. 4b, increasing URLLC traffic rates can pose challenges to URLLC reliability. Our proposed method still maintains reliability in transmitting URLLC packets compared to varied ϵ values of the greedy method. Unlike the greedy method, which relies on fixed ϵ values and can become trapped in sub-optimal choices, Thompson sampling continually explores and exploits new strategies, ensuring that the system can respond effectively to



(a) CDF of URLLC transmission error probability when $\phi = 80$ packets/slot



(b) CDF of URLLC transmission error probability when $\phi = 120$ packets/slot

Fig. 4: CDF of URLLC transmission error probability for different ϕ .

changing traffic patterns and network states. This adaptability is crucial for URLLC, where even minor delays or packet losses can have significant repercussions. By efficiently balancing the exploration of new transmission strategies with the exploitation of known successful ones, Thompson sampling minimizes the risk of prolonged failure modes. It is evident from the results that Thompson sampling improves performance by introducing a probabilistic element that helps deal with uncertainties compared to fixed learning rates.

VI. CONCLUSION

This study has addressed the resource allocation challenges in multi-cell wireless systems catering to eMBB and URLLC users. Through the formulation of an optimization problem and the development of a distributed learning framework, particularly leveraging a Thompson sampling-based DRL approach, we have presented a solution capable of making online resource allocation decisions. The proposed approach aligns with the evolving O-RAN network architectures, allowing for

efficient learning over wireless networks. Our simulation results demonstrate the effectiveness of the algorithm in meeting the QoS requirements for both eMBB and URLLC users. This research contributes valuable insights into optimizing resource utilization in dynamic and heterogeneous wireless environments, paving the way for enhanced network performance and user satisfaction across diverse communication services.

ACKNOWLEDGMENT

This work was supported by EPSRC projects, CHEDDAR EP/X040518/1 and CHEDDAR Uplift EP/Y037421/1.

REFERENCES

- [1] O. Alliance, "O-ran use cases and deployment scenarios," *White Paper*, Feb, 2020.
- [2] S. T. Shah, S. F. Hasan, J. Gu, and M. Y. Chung, "Sc-fdma-based resource allocation and power control scheme for d2d communication using lte-a uplink resource," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, pp. 1–15, 2015.
- [3] S. T. Shah, J. Gu, S. F. Hasan, and M. Y. Chung, "Proportional fair scheduling using water-filling technique for sc-fdma based d2d communication," *Physical Communication*, vol. 59, p. 102108, 2023.
- [4] M. Zangoeei, M. Golkarifard, M. Rouili, N. Saha, and R. Boutaba, "Flexible ran slicing in open ran with constrained multi-agent reinforcement learning," *IEEE Journal on Selected Areas in Communications*, vol. 42, no. 2, pp. 280–294, 2024.
- [5] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5g wireless network slicing for embb, urllc, and mmcc: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55 765–55 779, 2018.
- [6] R. M. Sohaib, O. Onireti, Y. Sambo, R. Swash, S. Ansari, and M. A. Imran, "Intelligent resource management for embb and urllc in 5g and beyond wireless networks," *IEEE Access*, vol. 11, pp. 65 205–65 221, 2023.
- [7] R. M. Sohaib, O. Onireti, Y. Sambo, and M. A. Imran, "Network slicing for beyond 5g systems: An overview of the smart port use case," *Electronics*, vol. 10, no. 9, p. 1090, 2021.
- [8] M. Alsenwi, N. H. Tran, M. Bennis, S. R. Pandey, A. K. Bairagi, and C. S. Hong, "Intelligent resource slicing for embb and urllc coexistence in 5g and beyond: A deep reinforcement learning based approach," *IEEE Transactions on Wireless Communications*, vol. 20, no. 7, 2021.
- [9] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5g wireless network slicing for embb, urllc, and mmcc: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55 765–55 779, 2018.
- [10] C. Sun, C. She, C. Yang, T. Q. Quek, Y. Li, and B. Vucetic, "Optimizing resource allocation in the short blocklength regime for ultra-reliable and low-latency communications," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 402–415, 2018.
- [11] G. Pocovi, B. Soret, K. I. Pedersen, and P. Mogensen, "Mac layer enhancements for ultra-reliable low-latency communications in cellular networks," in *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2017, pp. 1005–1010.
- [12] N. Strodthoff, B. Göktepe, T. Schierl, C. Hellge, and W. Samek, "Enhanced machine learning techniques for early harq feedback prediction in 5g," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 11, pp. 2573–2587, 2019.
- [13] N. Strodthoff, B. Göktepe, T. Schierl, W. Samek, and C. Hellge, "Machine learning for early harq feedback prediction in 5g," in *2018 IEEE Globecom Workshops (GC Wkshps)*, 2018, pp. 1–6.
- [14] I. Osband, C. Blundell, and B. Van Roy, "Deep exploration via bootstrapped dqn," *Advances in neural information processing systems*, vol. 29, 2016.