IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 19, 2022, accepted April 11, 2022, date of publication April 25, 2022, date of current version May 16, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3170447

Refiner GAN Algorithmically Enabled Deep-RL for Guaranteed Traffic Packets in Real-Time URLLC B5G Communication Systems

ADEEB SALH[1,2], LUKMAN AUDAH[1], (Member, IEEE), KWANG SOON KIM[3], (Senior Member, IEEE), SAEED HAMOOD ALSAMHI[4,5], MOHAMMED A. ALHARTOMI[2], (Member, IEEE), QAZWAN ABDULLAH[1,6], (Member, IEEE), FARIS A. ALMALKI[7], AND HANEEN ALGETHAMI[8], (Senior Member, IEEE)

[1]Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, Johor 86400, Malaysia
[2]Department of Electrical Engineering, University of Tabuk, Tabuk 47512, Saudi Arabia
[3]School of Electrical & Electronics Engineering, Yonsei University, Seodaemun-gu, Seoul 03277, South Korea
[4]SRI, Technical University of the Shannon: Midlands Midwest, Athlone, N37 F6D7 Ireland
[5]Faculty of Engineering, IBB University, Ibb, Yemen
[6]Faculty of Electrical Engineering, Universiti Teknikal Malaysia Melaka, Durian Tunggal 76100, Malaysia
[7]Department of Computer Engineering, College of Computers and Information Technology, Taif University, Taif 21944, Saudi Arabia
[8]Department of Computer Science, College of Computers and Information Technology, Taif University, Taif 21944, Saudi Arabia

Corresponding authors: Lukman Audah (hanif@uthm.edu.my) and Qazwan Abdullah (gazwan20062015@gmail.com)

This work was supported by Universiti Tun Hussein Onn Malaysia through the Publication Fund E15216. The authors are grateful to the University of Tabuk in Saudi Arabia for funding this research work through the project number S-0237-1438, and the Deanship of Scientific Research at Taif University, Kingdom of Saudi Arabia for funding this project through Taif University Researchers Supporting Project No. TURSP-2020/265. The work of Saeed Hamood Alsamhi is supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 847577; and a research grant from Science Foundation Ireland (SFI) under Grant Number 16/RC/3918 (Ireland's European Structural and Investment Funds Programs and the European Regional Development Fund 2014–2020.

ABSTRACT Ultra-reliable and Low-latency Communications (URLLC) is expected to be one of the most critical characteristics Beyond fifth-Generation (B5G) cellular networks with stringent low latency and high-reliability requirements. The Deep Reinforcement Learning (deep-RL) framework has been applied to predict the optimization of a Resource Block (RB) and minimize Power Allocation (PA) to guarantee a high End-to-End (E2E) reliability and low E2E latency under rate constraints. This paper proposes a novel Policy Gradient-based Actor-Critic Learning (PGACL) algorithm to optimize the policy gradient for optimal rate allocation to solve the RB, minimize power, and guarantee a solution for URLLC scheduling. The purpose of a PGACL algorithm is to provide a good policy with a closer convergence rate and a low computational cost depending on the reduced action space for every user. URLLC systems need to operate in highly reliable systems and account for extreme network conditions. Therefore, we proposed the refiner Generative Adversarial Networks (GANs) that apply enough extreme events for the deep-RL agent to generate synthetic data with high reliability similar to real data based on the regulated number of extreme events in the dataset. This refiner GAN method enables a deep-RL approach to generate large amounts of data practically used in real-time operations. Simulation results showed that the proposed deep-RL for refiner-GAN can omit the transient training time and develop deep learning based on a controlled set of unlabeled real traffic at a relatively short time. Furthermore, the refiner GAN demonstrated 99.9999% reliability and E2E latency of less than 1.4ms.

INDEX TERMS URLLC, beyond fifth-generation, end-to-end, generative adversarial networks.

I. INTRODUCTION

With the rapid deployment of wireless networks to support diverse applications, e.g., smart city, and intelligent transportation, Beyond Fifth Generation (B5G) networks are required to provide seamless access and diverse services for a huge number of devices over a limited radio spectrum radio. In wireless networks, different devices have various Quality-of-Service (QoS) requirements. For QoS guarantee, in B5G wireless networks, Ultra-reliable and Low-latency

The associate editor coordinating the review of this manuscript and approving it for publication was Zhouyang Ren.

Communications (URLLC) is one of the most challenging services with stringent low latency and high-reliability requirements, i.e., in the 3rd Generation Partnership Project (3GPP), a general URLLC requirement of a one-way radio is 99.999% target reliability with 1ms latency [1], [2]. Nevertheless, one key challenge in the B5G system is how to design enough extreme events for the Deep Reinforcement Learning (deep-RL) agent to intelligently make decisions (such as Resource Allocation (RA), Resource Block (RB), energy management, and transmission scheduling) for wireless networks under different devices [3].

To accommodate a high data rate in cellular networks of B5G, real-time optimization is needed to control the generation of the real data from radio resources under time-varying network conditions. Recently, much attention has been paid to the study of URLLC, RA problems, and decision making in wireless networks such as [4], [5]. The optimization of the RA problem in [5] depends on the proposed two-phase framework, which includes the Enhanced Mobile Broadband (eMBB) RA and URLLC scheduling to maximize the data rate of Users (UEs), by considering the reliability of both eMBB and URLLC. RA for Orthogonal Frequency Division Multiple Access (OFDMA) can decrease the End-to-End (E2E) delay and achieve the stringent reliability of the uplink and Downlink (DL) transmission in mobile edge computing systems in [1], [6], [7]. This efficient RA algorithm is used to guarantee the maximum delay by offering a partial iteration between the uplink and the DL to reduce E2E latency [8]. B5G addresses several new service applications, such as drones, virtual reality, the advanced Internet of Things (IoT), Artificial Intelligence (AI), and autonomous driving. These applications require high reliability and low latency. Virtual environments such as videos and images need big data rates with ultra-reliability in real-time. These applications with a short packet size can provide high reliability and low latency, based on localization in time with Transmission Time Intervals (TTIs). Moreover, to achieve low latency and high reliability, the data package should be small, and the TTI should be short in the era of B5G[9]. The transmission latency can be decreased when the blocklength is short, and the relation between blocklength and error probability has been studied, as shown in [10]. This relation improves RA for short packet transmission in URLLC, as shown in [11], [12]. The RA problem is a power minimization implement the URLLC-B5G requirements, optimization algorithms cannot be performed by sacrificing QoS due to compromised reliability, latency, and rate limitations. Decreasing the total power consumption for a Base Station (BS) can be achieved by improving the bandwidth and Power Allocation (PA) by developing deep transfer learning for radio RA in real-time. However, minimizing power consumption of wireless UEs in cloud radio access networks cannot be executed [13]–[15] because they do not adopt the limitation of deep-RL when working with large action spaces. Another challenge is gathering real data for training deep learning.

A. RELATED WORKS

In order to achieve B5G service provisioning URLLC service. The problem of transmitting more packets and reducing power transmission levels is based on the decision policy's current state. The work in [16] proposed a model-based and data-driven unsupervised learning method for designing a burstiness-aware scheduling framework that reserves bandwidth for UEs to satisfy the ultrahigh reliability requirement. The authors of [17] propose a packet prediction technique to predict the future incoming packets based on the packets in the current queue. The author of [6] studied the RA problem for a mission-critical IoT system to achieve a data rate under short packet communications by jointly enhancing the bandwidth and PA. The authors of [4] proposed an actor-critic Reinforcement Learning (RL) that uses a new reward function for RB allocation and PA to enhance the learning efficiency by applying a learning policy and interacting with the environment. This actor-critic system supports a highly reliable and low latency in device-to-device-enabled vehicle-to-vehicle wireless networks. Many studies used training deep-RL to develop a novel artificial agent capable of learning and substantially enhancing prediction during the training process to collect more datasets, as shown in [15], [19], [20], [22]. If the deep-RL agent is utilized without training, the system will be lacked experience since the beginning, resulting in an unreliable system. In [6], [15], [18], the systems did not take into account the learning reliability based on the training dataset to accommodate extreme and critical events and, therefore, could not demonstrate the B5G URLLC that occurs in real wireless networks. Previous works, as shown in [18], [19], could not satisfy the requirement of B5G URLLC because the systems did not gain considerable experience through rigorous training to improve unusual traffic patterns, extreme events, and unforeseen network congestion. Therefore, the requirements of B5G URLLC can be fulfilled by adopting the large action space that the deep-RL agent can take. The deep-RL agent needs to address more action space. However, this large set of actions makes RB and reliability unsuitable for deep-RL frameworks [20]. Previous studies [5], [8], [18], [19], [22], and [27] were unable to handle the large amounts of data involved in URLLC and demonstrated a high order time complexity, making them unsuitable for real-time requirements. However, deep-RL has difficulty achieving a large label of a real dataset in real-time. Deep-RL for Generative Adversarial Networks (GANs) in the real-time setting is needed to achieve large arrival rates [21]–[24]to authorize the deep-RL agent to gain experience.

A large training sample is critical to handle the large amounts of data involved in URLLC to guarantee the training efficiency and improve the learning speed and learning stability toward the optimal policy [20]. A large real dataset in real-time achieves by using the reward clipping scheme to enhance the training constancy of GAN [23] by improving learning an approximate distribution of the state-action to obtain more stable and superior learning. Achieving a suitable

strategy for transmitting high packets transmission efficiency of different buffers through multiple channels depends on the transmission scheduling mechanism using deep learning [24]. The proposed GAN-powered deep distributional Q network decreases the size of the action space provides a good transmission packet, guarantees high reliability, and achieves the optimal RA [21]. To meet the target reliability and guarantee the desired arrival rate to every UE depends on minimizing the BS power.

B. MOTIVATION AND CONTRIBUTIONS

Motivated by the above issues, to satisfy the E2E delay requirements of network UEs, we have addressed the joint problem of power minimization with a rate constraint of UEs, ultrahigh reliability, and ultralow latency. To solve this problem, we proposed a deep-RL framework to measure the E2E reliability, and E2E delay latency for every UE based on using a dynamically predicted traffic model, jointly allocating RBs and power minimization UEs under the constraint rate and URLLC. Deep-RL uses a Deep Neural Network (DNN) to gather data through the learning process. Fig. 1 illustrates that deep learning combines Artificial Neural Networks (ANN) with an RL agent's experience to learn the best actions possible in a virtual environment. Both URLLC RB allocation and PA transmission relax into convex optimization problems, which become a non-deterministic polynomial-time problem, causing difficulty to obtain a closed-form solution. The system was proposed to obtain an optimal RA and extreme rate requirements of UEs.

- We proposed a deep-RL outline that uses two feedback inputs, transmits power and reliability, and updates the DNN in every time slot. Intelligent B5G-URLLC is used to schedule and guarantee ultra-reliability and minimize power transmission in every time slot based on the required feedback received in small time slots and the prediction of the significance of its actions in the future. The Lyapunov function solves the queue delay for the arriving packets waits for transmission. In addition, the Lyapunov function guarantees a minimized transmitting power and ultra-reliability with a time-varying system for every UE.
- Even though the deep-RL effectively adopts the large state space problem, it still has to guarantee the desired rate for every action by addressing the huge state space and action space. Therefore, we proposed a Policy Gradient-based Actor-Critic Learning (PGACL) that can provide a good policy with a closer convergence rate and a low computational cost based on reduction in action space by adapting the exact reliability and latency for every UE. Furthermore, the policy gradient based on training deep-RL for PGACL can select the optimal solution by iteratively leveraging the Bellman equation that provides the optimal distribution of actions and develops the estimation of action values in an integral randomness environment.
- Deep-RL for B5G URLLC uses a higher number of training samples to select the optimal state decision based on the historical data of traffic packs. Issues related to the high number of training samples appear at a sudden increase in the arrival rate of every UE with a long recovery time. In this case, the system requires a transient time, which is critical to perform URLLC. Furthermore, using only deep-RL learning makes it difficult to achieve large labels of a real dataset in real-time settings. We proposed deep-RL for refiner GANs to provide sufficiently accurate traffic packs in real-time settings by incorporating a large number of unlabeled training

TABLE 1. List of notations and abbreviations.

LIST OF NOTATIONS

$\check{\mathrm{I}}, \mathcal{J}$	Set of URLLC UEs and RBs, respectively
i, j	Number of UEs and number of RBs
$z_{ij}^t, \mathcal{P}_{ij}^t$	RBs allocation and DL transmission PA, respectively for $J \in \mathcal{J}$ and $I \in \check{\mathrm{I}}$
$\hbar_{ij}$	Channel gain of the transmission from the BS to UE (i)
$\mathcal{N}_0$	Single-sided noise spectral density
B	RB bandwidth
R_i^t	The achievable rate depending on RB to every UE $i \in I$
ψ_i^t	Reliability for a UE (i)
ξ	Duration of transmission time intervals
θ_i^t	E2E latency for UE (i)
$\mathcal{C}_i^t$	Packet arrival rate in a time slot
a_t	Action
s_t	State
$\mathcal{R}$	Reward
$\mathcal{v}_i^t$	Time-varying weight to guarantee the URLLC reliability
π_{ij}^t	The conditional distribution of the RB (j) for the allocated UE (i) at any TTI
$\pi(a_t, s_t)$	Policy function
ϕ	The policy of a parameter vector
$\mathfrak{X}(s, a)$	Updates of the basis function vector
$\mathfrak{f}_t$	The critic that uses the temporal-difference learning
ℓ_c	Critic learning rate
$\mathfrak{Q}^*{}_{\mathbb{R}}$	Refiner neural network
$\mathfrak{Q}_{\mathbb{R}}, \mathfrak{Q}_{\mathbb{D}}$	Regular weight for the refining network and regular weights of the discriminator
$\mathfrak{p}_g(\mathbb{x})$	Global optimality distribution of synthetic data
$\mathfrak{y}$	The vector of Lagrangian dual variables
∇_ϕ	The gradient of the objective function

LIST OF ABBREVIATIONS

URLLCs	Ultra-reliable low-latency communications
B5G	Beyond fifth-generation
Deep-RL	Deep reinforcement learning
RB	Resource block
RA	Resource allocation
E2E	End-to-end
PGACL	Policy gradient-based actor-critic learning
GANs	Generative adversarial networks
QoS	Quality of service
3GPP	3rd Generation partnership project
eMBB	Enhanced mobile broadband
UEs	Users
PA	Power allocation
DL	Downlink
IoT	Internet of things
AI	Artificial intelligence
TTIs	Transmission time intervals
RL	Reinforcement learning
DNN	Deep neural network
ANN	Artificial neural networks
gNB	gNodeB
OFDMA	Orthogonal frequency division multiple access
NAK	Negative acknowledgement
RNN	Refiner neural network
TD	Temporal-difference

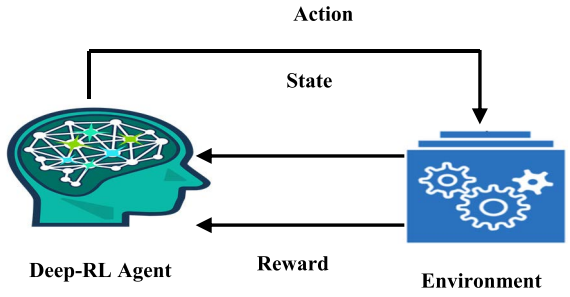

FIGURE 1. Agent-environment interaction in the deep-RL.

samples to support B5G networks. A more vital real data requires covering all unexpected situations and training periods by using a sufficient number of extreme events and controlling the level of extreme events when the refiner's output is similar to the refiner's input.

II. SYSTEM MODEL

We have considered the DL OFDMA scenario of a single BS, where the BS is located at the center of the cell and a set $\breve{I}$ of I UEs and has a set $\mathcal{J}$ of J available at RBs. The wireless RA consists of the following $I \times J$ RB allocation matrix and $I \times J$ PA matrix. The Shannon capacity rates cannot be used due to the small packet size in the URLLC traffic. Instead, the achievable rate can use the finite blocklength to define a URLLC UE i on RB j at any time slot t. The achievable URLLC rate based on finite blocklength is given by [25], [30]:

$$R_i^t = \xi \left[z_{ij}^t B \log_2 \left(1 + \frac{\hbar_{ij}^t \mathcal{P}_{ij}^t}{\mathcal{N}_0 B} \right) \right], \tag{1}$$

where $\mathcal{P}_{ij}^t$ is the transmission power of gNodeB (gNB) for URLLC user ($i \in I$), and $\hbar_{ij}$ is the Rayleigh fading channel gain from the BS to UE i on RB j at time slot t. ξ represents the duration of a TTI, z_{ij}^t is the RB allocation display with $z_{ij}^t = 1$ when RB j is allocated for UE i at any TTI of the time slot t; otherwise, $z_{ij}^t = 0$. B is the bandwidth of RB, and $\mathcal{N}_0$ is the single-sided noise spectral density. The achievable rate R_i^t depends on RB for every UE by calling TTI of the time slot t to serve URLLCs based on the reliability constraint.

From (1), the reliability of the URLLC decreases due to interference. Thus, for serving URLLC data rate, UEs are

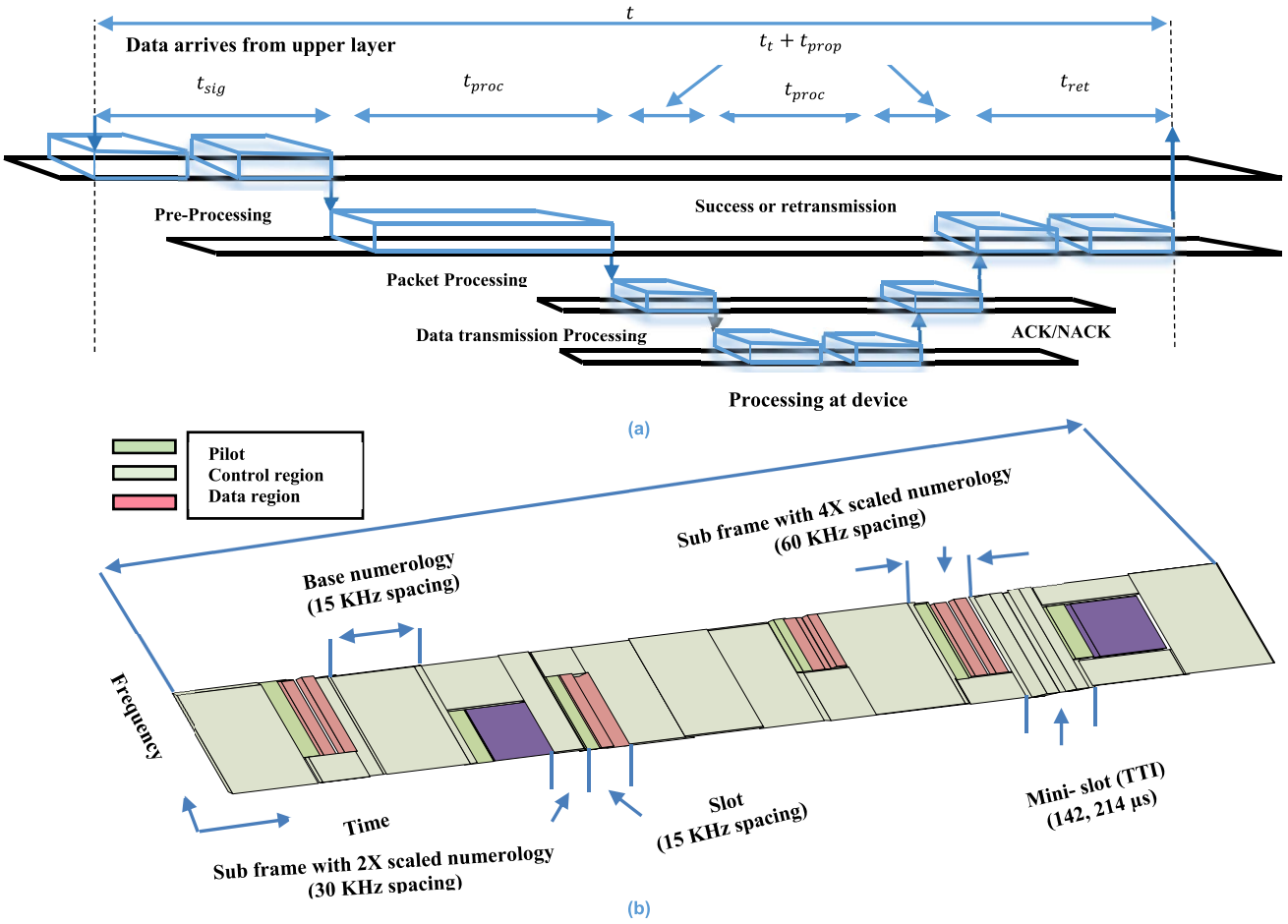

FIGURE 2. (a) Low latency components of URLLC, (b) Frame structure for URLLC-B5G [6].

based on concentrating a finite blocklength and error probability [25] to support B5G-URLLC scenarios. The achievable rate depends on RB to every UE $i \in I$ as shown (1). The probability of the E2E packet delay surpassing a predefined target E2E latency θ_i^t for UE i is represented as reliability Ψ_i^t. The transmission delay includes E2E packet delay and queue delay for the arriving packets. To meet the system reliably and latency requirements, the system must keep rate concerning the packet arrival rate [22], [30]., i.e.

$$R_i^t > \mho\left(\Lambda_i^t, \theta_i^t, \gamma_i^t, \Psi_i^t\right) > \Lambda_i^t \gamma_i^t, \tag{2}$$

where Λ_i^t represents the vector of current serving URLLC UEs for particular packet size, γ_i^t is the random number representing the arrival URLLC packet rate at UE i at TTI, and $\mho$ (.) refers to an unknown function and we consider it as implicitly approximate. A guarantee of high reliability depends on redesigning the physical layer and enabling technologies, including packet and frame structure, as shown in Fig. 2. The short packet transmission or efficient low-latency transmission over 0.1 ms depends on the control of signaling and scheduling information for a large portion of the transmission latency as $t = t_t + t_{prop} + t_{proc} + t_{ret} + t_{sig}$. Where t represent the time slot to decrease the processing latency, t_t is the latency, t_{prop} is signal propagation time, t_{proc} is the time to achieve precoding and decoding, t_{ret} is the time taken for retransmission, and t_{sig} represents pre-processing time. The success probability of negative acknowledgement (NAK) is sent by the UE if there is no acknowledgement (ACK) ACK/NAK, and Pr is the successful probability of ACK that guarantees high reliability of data packet when the UE sends NAK. The mini-slot-level (142, 241 μs), 2×, and 4× provide the base numerology for sub-frame and latency in the slot t_t, which improves the success probability of packet [9]. Allocation of RB is achieved by calling in any TTI of a time slot based on the reliability constraint. The target E2E latency for UE i for every packet loss probability can be written as

$$Pr\left[\sum_{i=1}^{I}\sum_{i=1}^{I} g\left(\theta_i^t, \gamma_i^t\right) \le 0\right] \ge 1 - \Psi_i^t \quad \forall\, i \in \check{\mathrm{I}}, \tag{3}$$

where Ψ_i^t represent the reliability for a UE i, and θ_i^t E2E latency for UE i. The achievable rate depends on the ability to ensure the URLLC when the Signal –to- Noise Ratio (SNR) ≥ 5 dB, as shown in [5], [26]–[28]. However, a reduction in the reliability of each UE occurs due to variations in the quality of the channel. The reliability Ψ_i^t and latency θ_i^t of URLLC depend on the ability to ensure that the outage probability of E2E instantaneous packet delay is more than the target E2E latency $g\left(\theta_i^t, \gamma_i^t\right)$ for UE i, as shown in (3). Designing reliable RA to keep the minimum required rate depends on the instantaneous arrival rate that satisfies the queue stability condition and the reliability and latency condition ($\theta_i^t < \gamma_i^t \le \Psi_i^t \ \forall\, t$). The average data arrival rate R_i^t of UE ($i \in \check{\mathrm{I}}$ at TTI) was needed to serve URLLC UEs, which is expressed as follows:

$$\frac{1}{t}\sum_{\tau=1}^{t}\sum_{i=1}^{I} \xi R_i^t \ge \mathcal{C}_i^t, \quad \forall\, i,\ t, \tag{4}$$

where $\mathcal{C}$ is the packet arrival rate in the time slot. The vector URLLC of UEs is expressed as $\boldsymbol{\theta_i^t} = 1$, when $\boldsymbol{i \in I}$ by next-generation gNB at TTI of the time slot $\boldsymbol{t}$; otherwise, $\boldsymbol{\theta_i^t} = 0$. Based on (3) and (4), the probability and latency ensure the URLLC.

A. PROBLEM FORMULATION

The goal for the system is to allocate resources to minimize the average DL power while maintaining reliability, latency, and rate for the UEs. So, we pose this RA problem as a power minimization problem that is subject to a QoS constraint on maximizing the reliability of every packet and maintaining the required E2E latency and rate constraint of UEs are given in [2], [5], [22], and rate constraint of UEs can be formulated as:

$$\min_{z_{ij}, \mathcal{P}_{ij}} \frac{1}{t}\sum_{\tau=1}^{t}\sum_{i=1}^{I}\sum_{j=1}^{J} \mathcal{P}_{ij}^{\tau}, \tag{5}$$

$$s.t\ Pr\left[\sum_{i=1}^{I} g\left(\theta_i^t, \gamma_i^t\right) \le 0\right] < 1 - \Psi_i^t \ \ \forall\, i \in \check{\mathrm{I}}, \tag{5a}$$

$$\frac{1}{t}\sum_{\tau=1}^{t}\sum_{i=1}^{I} \xi R_i^t > \mathcal{C}_i^t, \quad \forall\, i,\ t, \tag{5b}$$

$$\theta_i^{\tau,t} < \gamma_i^t \le \Psi_i^t \quad \forall\, \tau,\ t, \tag{5c}$$

$$\sum_{i} z_{ij}^t = 1, \quad \forall\, j \in \mathcal{J},\ \forall\, t, \tag{5d}$$

$$\mathcal{P}_{ij}^t \ge 0,\ z_{ij}^t,\ \theta_i^t \in \{0,\ 1\}, \quad \forall\, i \in \check{\mathrm{I}},\ \forall\, j \in \mathcal{J},\ \forall t, \tag{5e}$$

$$\sum_{j \in J}\sum_{i \in I} z_{ij}^t \theta_i^t \le |K|, \quad \forall\, j \in \mathcal{J}. \tag{5f}$$

The optimization problem in (5) seeks to minimize the average power consumed by the BS. In the outage probability in (5a), the packet scheduling utilizes available radio resources efficiently, ensures fairness among scheduled UEs, and satisfies the QoS requirements depending on maximizing the reliability of every packet and maintaining the required E2E latency [2]. A higher value of traffic packets (5b) in the TTI lowers the chance of a UE getting scheduled, and therefore, it ensures fairness among the UEs over a certain time duration. The transmission short packet delay of all UEs and reliability constraints are preserved by (5a) and (5b). However, the reliability condition and ultralow latency in (5a) must ensure that the E2E delay is less than $\left(g\left(\theta_i^t, \gamma_i^t\right) \le 0\right)$ with minimum reliability of $1 - \Psi_i^t$ and g represent the minimum required R_i^t for the instantaneous arrival rate and minimum power $\mathcal{P}^t$. Both (5b) and (5c) ensure that the rate of each UE enables connection to the BS and determination of data transmission at every UE to guarantee target reliability Ψ_i^t at every time slot t. Moreover, constraints (5d) and (5e)

represent the orthogonality of RBs among the URLLC and PA. Resource restraint is given by constraint (5f). The latency or reliability cannot be sacrificed to decrease power, as shown in (5).

B. DECOMPOSITION AS A SOLUTION APPROACH FOR PROBLEM

The wholly achievable UE RB and joint optimization of PA depend on obtaining the best solution by searching for possible URLLC location TTIs in the space. The gNB allows URLLC UEs to obtain some RBs directly on TTI within every time slot t. At the beginning of the time slot t, gNB schedules each of its RBs, the URLLC traffic requests any τ of t to come in, and the scheduler tries to serve the requests in the next $t+1$. The portion of all RB z_{ij}^t is required for serving URLLC traffic overlaps at TTI. The reality of chance constraint in (5a) is still computationally expensive, and the combinatorial variable in (5) has difficulty reaching a globally optimal solution [3], [5], [22], [28]. Currently, it needs to adapt (5a) into deterministic form for explaining (5) by assuming $g\left(\theta_i^t, \gamma_i^t\right) = \sum_{i\in I} \theta_i^t - \gamma_i^t, \quad \forall\, t$. Thus, γ_i^t URLLC traffic arriving at gNB (any TTI of the time slot, t follows a Gaussian distribution, i.e., $\gamma_i^t \;\; N\left(\Gamma, \rho^2\right)$, where Γ and ρ^2 represent the mean and variance of γ.

$$g\left(\theta_i^t,\ \gamma_i^t\right) \le 0 = \Pr\left\{\sum_{i\in I} \theta_i^t - \gamma_i^t \le 0\right\}, \tag{6}$$

$$= \Pr\left\{\sum_{i\in I} \theta_i^t \le \gamma_i^t\right\} < 1 - \Psi_i^*, \tag{6a}$$

$$= 1 - \Pr\left\{\sum_{i\ \in I} \theta_i^t \ge \gamma_i^t\right\}, \tag{6b}$$

$$= 1 - \Pr\left\{\frac{\gamma_i^t - \Gamma}{\rho} \le \sum_{i\ \in I} \frac{\theta_i^t - \Gamma}{\rho}\right\}$$

$$= 1 - \mathfrak{F}_\gamma\left\{\sum_{i\in I} \theta_i^t\right\}, \tag{6c}$$

where $\mathfrak{F}_\gamma$ represents the cumulative distribution function of the instantaneous packet size of γ_i^t. From (6a), the reliability can reduce the E2E delay for the arrival rate of several URLLC packets at UE i at TTI.Therefore, constraints (6) can be rewritten as follows:

$$Pr\left[\sum_{i=1}^{I} \theta_i^t - \gamma_i^t \ \le 0\right] <\ 1 - \Psi_i^*, \tag{7}$$

$$\Pr\left\{\sum_{i\ \in I} \theta_i^t \le \gamma_i^t\right\} < 1 - \Psi_i^*, \tag{7a}$$

$$= \frac{1}{t}\sum_{\tau\ =1}^{t}\sum_{i=1}^{I} \xi R_i^t\ > \mathcal{C}_i^t, \quad \forall\, i,\ t, \tag{7b}$$

$$= \frac{1}{t}\sum_{\tau\ =1}^{t}\sum_{i=1}^{I} \xi R_i^t - \frac{1}{t}\sum_{\tau=1}^{t} \mathcal{C}_i^t < 0, \tag{7c}$$

$$= \frac{1}{t}\sum_{\tau=1}^{t} e^{(R_i^t - \mathcal{C}_i^t)\gamma_i^t}\ < 1 - \Psi_i^*, \tag{7d}$$

$$= \min_{z_{ij},\, \mathcal{P}_{ij}} \sum_{i=1}^{I}\sum_{j=1}^{J} \mathcal{P}_{ij}^t + \left(\sum_{i\ =1}^{I} R_i^t - \mathcal{C}_i^t\right) + \left(e^{(R_i^t - \mathcal{C}_i^t)\gamma_i^{\tau,t}} - \left(1 - \Psi_i^*\right)\right). \tag{7f}$$

At every time slot t, the target reliability Ψ_{i}^* depends on the rate of every UE. In addition, the minimization of transmit power depends on the PA and RBs of every UE. The (7) and (7c) show that the low latency and reliability depend on adapting the exact reliability with the smallest resource usage. From (7) - (7f) the fairness doctrine for this mission contributes stationary service quality enhances URLLC rate based on finite blocklength, and makes UEs more pleasant in the network [2]. Moreover, the PA is selected as the central resource for optimization problems (7a). Thus, the repeated form of (5a) and the problem of RB allocation, i.e., $\sum_{i\in I} z_{ij}^t \le |K|\,,\ \forall\, j \in J$ for any UE i and any RB j are still NP-hard due to the appearance of a combinatorial variable. The deep-RL agent must be trained with sufficient experience because the deep-RL gaining can take a long time to guarantee high reliability.

C. INTELLIGENT URLLC-B5G SCHEDULING: DEEP-RL

From the problem formulation the proposed deep-RL framework will use two feedback inputs $\mathcal{P}_{ij}$ and z_{ij} to estimate its performance and update its DNN in every time slot: The total power in DL BS for every time slot $\mathcal{P}(\tau) = \min_{z_{ij},\, \mathcal{P}_{ij}} \frac{1}{t}\sum_{\tau\ =1}^{t}\sum_{i=1}^{I}\sum_{j=1}^{J} \mathcal{P}_{ij}^{\tau}$ and the calculated reliability Ψ_i^t of every UE can be calculated from the problem formulation in (5). Using those two inputs, the deep-RL can determine $\mathcal{P}_{ij}$ and z_{ij} for all i and j. After iteratively assigning $\mathcal{P}_{ij}$ and z_{ij} and receiving the needed feedback in a few time slots. Determining the desired rate R_i^t for every UE i depends on the application of deep-RL at every time slot. We will use the derivation in Section II-D. An Action Space Reducer for RB and PA R_i^t to the OFDMA resources $\mathcal{P}_{ij}$ and z_{ij} for all $i \in \check{\mathrm{I}}, j\in\mathcal{J}$ while decreasing the power in (Section II-D). Therefore, we will use the derivation in (Section II-E) for PGACL algorithm, whereas every UE achieves the data rate R_i^t and attains a reward function as shown in (8) and transmits it as feedback to the deep-RL that uses this feedback and updates every UEs R_i^t accordingly PGACL algorithm (Section II-E). The deep-RL framework is formally defined by its action-value function $\mathcal{A}$, state-space $\mathcal{S}$, and reward $\mathcal{R}$. The deep-RL framework takes action $(a_t \in \mathcal{A})$ at every state $(s_t \in \mathcal{S})$ and receives reward function, i.e., $\mathcal{R}\,(a_t, s_t)$. We have considered the number of arriving URLLC packets γ_i^t, instantaneous packet length $\theta_i^{\tau,t}$ for every UE, and channel variation at every time slot. Thus, the state of a time slot is defined as $s_t = \left(\gamma_i^t, \theta_i^t,\ \hbar_{ij}^t\right), \forall\ i \in \check{\mathrm{I}}, j\in\mathcal{J}$, the action space represents the DL transmission power, and the number of TTI of every RB allocation j for any UE i is expresed as

$a_t = \left(\mathcal{P}_{ij}^t,\ z_{ij}^t\right),\ \forall\, i,\ j$. We formulated the reward function to guarantee that the requirement of URLLC based on training its DNN and action space function is satisfied as follows:

$$\mathcal{R}\left(a_t, s_t\right) = -\sum\nolimits_{i\in I} \mathcal{v}_i^t \left(1 - \Psi_i^t\left(a_t, s_t\right) - \Omega\mathcal{P}\left(a_t\right)\right), \quad (8)$$

where $\mathcal{v}_i^t$ is a time-varying weight that guarantees the URLLC reliability when $\Psi_i^t < \Psi_i^*$ throughout the time slots, Ω represents the weight factor of power, and the total transmit power $\mathcal{P}\left(a_t\right) = \sum_{i=1}^{I}\sum_{j=1}^{J}\mathcal{P}_{ij}^t$ is a casual variable depending on the status of the channel gain. The delay priority in (5) is non-convex due to the non-convex function created from two inputs, $\mathcal{P}_{ij}^t$ and γ_i^t. $\mathcal{v}_i^t$, that use deep-RL to evaluate the weights of the power and the reliability constraint to define the time-varying weight $\mathcal{v}_i^t$ at step, $t+1$, as follows:

$$\mathcal{v}_i^{t+1} = \max\left\{\mathcal{v}_i^t + \Psi_i^* - \Psi_i^t\ , 0\right\}, \quad (9)$$

where Ψ_i^* represents the target reliability at time slot t, which is defined in (7). The reliability is accurate when the time-varying value increased $\mathcal{v}_i^t$, when the reliability is $\Psi_i^t < \Psi_i^*$. The maximum data arrival rate is stable and can satisfy $\mathcal{v}_i^{t+1} - \mathcal{v}_i^t \le \sum_{i=1}^{I} R_i^t - \mathcal{C}$. The number of bits to be transmitted in every TTI depends on the packet size of each UE. Deep-RL appropriated depends on whether the BS maximizes the reward function in (8), and the reliability for every UE must be guaranteed at the fixed point when $\mathcal{v}_i^{t+1} = \mathcal{v}_i^t$, also when the reliability is $\Psi_i^t \ge \Psi_i^*$. The convergence deep-RL algorithm first assumes that the time-varying value $\mathcal{v}_i^t$ converges to $\mathcal{v}_i^*$. This convergence of deep-RL can minimize the allocated power $\mathcal{P}_{ij}^t$, R_i^tmaximizing a data arrival rate and z_{ij}^t for every UE i and j of the time-varying value $\mathcal{v}_i^t$. By defining the Lyapunov function for every BS as $\left\|\mathcal{v}_i^{t+1} - \mathcal{v}_i^t\right\|^2 = \left\|\max\left\{\mathcal{v}_i^t + \Psi_i^* - \Psi_i^t, 0\right\} - \mathcal{v}_i^t\right\|^2$, the original optimization in (5b) is equivalent to (7d) to satisfy (7b). Therefore, if this constraint in (7b) is not satisfied in any time slot, the queue delay will depend on the number of arriving packets in the queue waiting for transmission, followed by the use of Lyapunov optimization to solve the RA problem related to time variation [22], [27]. The deep-RL guarantees the reliability for every UE when $\left\|\Psi_i^* - \Psi_i^t\right\|^2 \le 2\left(\mathcal{v}_i^* - \mathcal{v}_i^t\right)^T\left(\Psi_i^* - \Psi_i^t\right)$ and if the initial conditions of queue delay are finite. Then, high reliability is guaranteed for every UE based on the fixed point for the equal time variance ($\mathcal{v}_i^{t+1} = \mathcal{v}_i^t$), followed by $\mathcal{v}_i^t + \Psi_i^* - \Psi_i^t \le \max\left\{\mathcal{v}_i^t + \Psi_i^* - \Psi_i^t, 0\right\} - \mathcal{v}_i^t$. Additionally, the latency for every UE is guaranteed when the BS maximizes the reward in (8) when $\mathcal{v}_i^t + \Psi_i^* - \Psi_i^t \le \mathcal{v}_i^t$, and $\Psi_i^t \ge \Psi_i^*$. From (7f), the value in every time slot t shows more complexity, making it difficult to improve the cellular networks. The long order of time complexity and difficulty handling more active space in URLLC [5], [28]. The deep-RL algorithm must be able to address more action space in real-time. To solve this problem, a novel algorithm was proposed to reduce the size of the action space without limiting it.

D. ACTION SPACE REDUCER FOR RB AND PA

To improve the scalable hierarchical framework for the whole RB allocation z_{ij}^t and PA $\mathcal{P}_{ij}^t$ of the problem, the allocation solution with the smallest power was selected. The RA of the action space adopts the actions $\mathbb{r}^J$ for diverse integer representing the action space sizes, i.e., $\mathbb{O}\left(J^I\right) \times \mathbb{r}^J$ with dimensional RB $(I \times J)$ and PA $(I \times J)$. Reducing the action space can improve the rate of every UE by adjusting exactly the reliability and latency [29]. The optimization problem related to the action space function can be defined to obtain z_{ij}^t and $\mathcal{P}_{ij}^t$ from the optimization variable in (5).

$$\min_{z_{ij},\,\mathcal{P}_{ij}} \sum_{i=1}^{I}\sum_{j=1}^{J}\mathcal{P}_{ij}^t, \quad (10)$$

$$s.t\ R_i^t = \mathcal{C}_i^t \quad (10a)$$

$$(4a), (4b), (4c), (4d), (4e), \text{ and } (4f), \quad (10b)$$

where $\mathcal{C}_i^t = \left[R_1^t,\ R_2^t, \ldots.., R_I^t\right] \in \mathbb{r}^J$ represents the desired rate for every UE. The optimization problem in (10) is equivalent to decreasing the BS power and guaranteeing the rate R_i^t by collecting the desired rate $\mathcal{C}_i^t$ for every UE. The constraint (10a) guarantees that every UE can be performed with a required rate $R_i^t \ge \mathcal{C}_i^t$, and minimization of BS power in (10), which can be solved with constraint (10a) a form of inequality constraint $R_i^t \ge \mathcal{C}_i^t$, will be fulfilled in the form of equivalence [30].The convex optimization problem in (10)–(10b) can be solved by the augmented Lagrangian algorithm [28]. To reduce the state space dimensions, the rate R_i^t for every UE is defined by collecting the desired rate $\mathcal{C}_i^t$ for every UE. The deep-RL can reduce the error caused by the action space reducer and conventional RA by using the Lagrangian for the problem (10) while recognizing relaxation of the inequality constraint $R_i^t \ge \mathcal{C}_i^t$ by introducing auxiliary variables as follows:

$$\max_{\mathfrak{y}} \min_{z_{ij},\,\mathcal{P}_{ij}} \sum_{i=1}^{I}\sum_{j=1}^{J}\mathcal{P}_{ij}^t - \sum_{i=1}^{I}\mathfrak{y}\left(R_i^t - \mathcal{C}_i^t\right). \quad (11)$$

Moreover, the problem in (11) is dual decomposable for every RB, growing $\mathcal{C}_i^t$ will growth $\mathfrak{y}$, which can be rewritten as:

$$\begin{aligned}\min_{\mathcal{P}_{ij}} &\sum\nolimits_{i=1}^{I}\sum\nolimits_{j=1}^{J}\mathcal{P}_{ij}^t - \sum\nolimits_{i=1}^{I}\mathfrak{y}R_i^t \\ &= \min_{\mathcal{P}_{ij}} \sum\nolimits_{i=1}^{I}\sum\nolimits_{j=1}^{J}\mathcal{P}_{ij}^t \\ &\quad - \sum\nolimits_{i=1}^{I}\mathfrak{y}B\,log_2\left(1 + \frac{\hbar_{ij}^t\mathcal{P}_{ij}^t}{\mathcal{N}_0 B}\right), \quad \forall\, j \in \mathcal{J}, \end{aligned} \quad (12)$$

where $\mathfrak{y}$ is a vector of Lagrangian dual variables and $\mathcal{L}\left(\mathcal{P},\ z,\ \mathfrak{y}\right)$ is a convex function and has a closed-form solution. The optimal power $\mathcal{P}$ for a given $\mathfrak{y}$ is denoted as optimal $\mathcal{P}_{ij}^*$. The optimal PA is obtained when the equality in (5b) and (5e) is controlled. By utilizing the softmax function as the activation function in the output layer, we can guarantee

that $\mathcal{P}_{ij}^{t}\geq$. This problem in (12) can be derived in terms of $\mathcal{P}_{ij}^{t}$ as follows:

$$1-\mathfrak{y}B\left(\frac{\hbar_{ij}^{t}}{\left(\mathcal{N}_0\,B+\hbar_{ij}^{t}\mathcal{P}_{ij}^{*}\right)log\,2}\right)=0,\quad \forall\, i\ \in \check{\mathrm{I}}. \quad (13)$$

The optimal PA can be expressed as:

$$\mathcal{P}_{\mathrm{ij}}^{*}=\left[\frac{\mathfrak{y}\,\mathrm{B}}{\log 2}-\frac{\mathcal{N}_0\mathrm{B}}{\hbar_{\mathrm{ij}}^{\mathrm{t}}}\right]^{+},\quad \forall\, i\ \in \check{\mathrm{I}}. \quad (14)$$

Since the $\hbar_{ij}^{t}$ channel gain of the transmission is required to guarantee the rate of every UE, which decreases with the transmit PA to UE, then every RB can be allocated only to one UE, and the optimal solution of problem (12) is selected to allocate RB j to UE i where is expressed as:

$$i_j=\arg\min_{i}\mathcal{P}_{ij}^{*}-\mathfrak{y}B\log_2\left(1+\frac{\hbar_{ij}^{t}\mathcal{P}_{ij}^{*}}{\mathcal{N}_0B}\right),\quad \forall\, j\ \in \mathcal{J}. \quad (15)$$

Allocating the RB to every UE is based on the optimal PA $\boldsymbol{\mathcal{P}_{ij}^{*}}$ required in (14) and (15) by minimizing the system PA. The greedy allocation method is taken into account, and the UE for the optimal PA is selected in every RB $\boldsymbol{arg\,min_{i}\,\mathcal{P}_{ij}^{*}}$. According to [31], only the deep-RL complexity is affected by the increase in the number of UEs; the complexity of our algorithm is $\mathbb{O}\left(\boldsymbol{JI^3}\right)$ based on the chosen actions and will provide the new training state information to the agent in (12). Moreover, when reducing the action space, the desired rate for every action becomes $\boldsymbol{\mathcal{C}_i^t}=\left[\boldsymbol{R_1^t},\ \boldsymbol{R_2^t},\ldots..,\boldsymbol{R_I^t}\right]\in\mathbb{r}^{\boldsymbol{J}}$. This action space is still not scalable due to the N-dimensional excitation in $\mathbb{r}^{\boldsymbol{J}}$, as shown in [32]. The proposed deep-RL problem using a policy gradient is shown in the new subsection.

E. PGACL ALGORITHM FOR POLICY GRADIENT FOR OPTIMAL RATE ALLOCATION

In this section, the PGACL algorithm is explained to improve the performance of the policy gradient for optimal rate allocation and analyze its convergence. The agent aims to select a policy gradient algorithm such as the PGACL algorithm that can control the desired rate $\mathcal{C}_i^t$ or every action of the optimization problem in (12). Power allocation $\mathcal{P}_{ij}^{*}$ based on deep-RL can learn to find the optimal policy π^* through maximizing expected reward. The policy function can be described as $\pi\left(a_t,s_t\right)=\left\{\pi_{ij}^{t},\ \forall\, i\in\check{\mathrm{I}},j\in\mathcal{J}\right\}$, where π_{ij}^{t} is a conditional distribution of RB j, which is allocated for UE i at any TTI of the time slot t. The goal of deep-RL is to obtain the optimal policy π^* by selecting the desired rate and power according to the current state based on the decision policy and generating the maximal $\partial\left(s,a\right)$ for each state space s and action space a. Let $\pi^{*}=\arg\max_{\pi}\partial^{\pi}\left(s,a\right)\ \forall a\ \in$. With given policy π, the policy action value, i.e., $\partial^{\pi}\left(a_t,s_t\right)$ is a cumulative discounted reward at a given π as shown in Fig. 3, which is used to update the deterministic policy to achieve a policy that exploits the expected return of the algorithm in (16) and can be expressed as

$$\partial^{\pi}\left(s,a\right)=\mathbb{E}_{\pi}\left[\sum\nolimits_{t=0}^{\infty}\mu_t\mathcal{R}\left(s_t,a_t\right)\middle|\, s_0=s,\pi\right], \quad (16)$$

where $\mu_t\in(0,1]$ is a discount factor, and $\mathbb{E}$ is expectation. The cumulative discounted reward iteratively applying $\partial^{\pi}\left(s_{t+1},\ a_{t+1}\right)$ leads to junction $\partial^{\pi}\left(s,a\right)=\mathbb{E}\left[\mathcal{R}\left(s_t,\ a_t\right)+\mu_t\partial^{\pi}\left(s_{t+1},\ a_{t+1}\right)-\partial^{\pi}\left(s_t,a_t\right)\right]$. The best solution obtained by iteratively leveraging the Bellman equation in [16], is used to improve its ability and estimate action values in an integral randomness environment. The policy that maximizes $\mathbb{J}\left(\pi\right)$ of the DNN can be obtained and explained in [7]:

$$\mathbb{J}\left(\pi\right)=\int_{\mathcal{S}}\int_{\mathcal{A}}\pi\left(s\ ,\ a\right)\partial^{\pi}\left(s,a\right)\ \ dsda. \quad (17)$$

Based on (17), policy optimization π proposed a policy gradient PGACL algorithm that can provide a good policy with a closer convergence rate and a low computational cost by combining policy learning and value learning. The PGACL algorithm consists of two parts: a) The first part is called an actor part that can update the policy in DNN depending on the policy gradient. This policy is created based on proposing a parameter vector ϕ, as $\pi_{\phi}\left(s,\ a\right)=Pr\left(a\,|s,\ \phi\right)$. The agent performs the Bellman optimality in (16) on every transition of the selected action and obtains the target action value. This actor part can achieve incredible performance on deep-RL problems according to (17) in respect of ϕ as $\nabla_{\phi}\mathbb{J}\left(\pi_{\phi}\right)=\int_{\mathcal{S}}\int_{\mathcal{A}}\nabla_{\pi\phi_i}\partial^{\pi_{\phi}}\left(s,a\right)\ \ ds\ da$, where, ∇_{ϕ} is the gradient of objective function. Additionally, this policy of a parameter vector ϕ can provide the optimal distribution of actions $\mathcal{C}_{\mathrm{i}}^{\mathrm{t}}$ for every UE $i\ \in\check{\mathrm{I}}$ based on regularizing the actor's learning to correct the error and improve stability as large steps in the actor update. The parameterized policy can control the policy by the Gibbs distribution as $\pi_{\phi}\left(s,a\right)=e\left(\phi\Gamma\left(s,a\right)\right)/\sum_{\grave{\mathrm{a}}\in\mathcal{A}}\left(\phi\Gamma\left(\grave{s},a\right)\right)$, based on using the gradient function in (17) as $\phi_{t+1}=\phi_t+as\,\varsigma_a\nabla_{\phi}\mathbb{J}\left(\pi_{\phi}\right)$, where $\Gamma\left(s,a\right)$ is the feature vector and ς_a represents the learning rate of the actor. b) The second part is called the critic part, which is used to learn the correct scheduler mechanism to be fulfilled at every TTI to maximize the rate allocation for the policy gradient. The function estimator in [33] is applied to estimate the value function $\mathrm{Ч}\left(s,a\right)$ of agent $\ddagger$, which is expressed as:

$$\mathrm{Ч}\left(s,a\right)=\mathrm{ч}^{T}\mathrm{Ж}\left(s,a\right)=\sum\nolimits_{\ddagger\in\mathcal{S}}\mathrm{Ч}_{\ddagger}\mathrm{Ж}_{\ddagger}\left(s,a\right), \quad (18)$$

where $\mathrm{Ж}=\left[\mathrm{Ж}_1\left(s,a\right),\ldots\ldots,\mathrm{Ж}_{\mathcal{S}}\left(s,a\right)\right]^{T}$ is the basis function vector. The critic utilities are used to calculate the error between the estimated and real values to achieve the performance gain in terms of the temporal-difference (TD) method as $\mathfrak{f}_t=\mathcal{R}_{t+1}+\mathrm{Ч}\left(s_{t+1}\right)-\mathrm{Ч}\left(s_t,a_t\right)$. The linear function estimator in (18) is used to update the weight vector, i.e.,

$\chi(s, a)$ by using the gradient descent method as:

$$\begin{aligned} \mathrm{Ч}(s_{t+1}, a_{t+1}) &= \mathrm{Ч}((\mathrm{s_t}), (\mathrm{a_t})) + \ell_c \, \mathrm{ғ_t} \nabla_{\mathrm{Ч}} \mathrm{Ч}(\mathrm{s}, \mathrm{a}) \\ &= ((\mathrm{s_t}), (\mathrm{a_t})) + \ell_c \, \mathrm{ғ_t} \, \chi(\mathrm{s}, \mathrm{a}), \end{aligned} \tag{19}$$

where $\mathrm{ғ}_t$ is the critic that uses the TD method, and ℓ_c is the critical learning rate. The value function in (18) has updated the DNN at every TTI by applying critic learning to the value of $\mathrm{Ч}(s, a)$ in (19). The PGACL algorithm trains DNN by testing random tuples in the experience pool based on selected action, next state (s_{t+1}), the current reward $\mathcal{R}_t$ and supplies the experience tuple. When $\mathrm{ғ}_t \in (0, 1)$ starts from any $\mathrm{Ч}(s, a)$ iteratively applying the operator $\mathrm{Ч}(s_{t+1}, a_{t+1})$, the iterative process starts to satisfy the Bellman optimality equation, as shown in [34].The value of time-varying $\mathscr{v}_i^t$ is updated according to (9) in the system to meet the target reliability while reducing the aggregate BS power. The issue can occur from a sudden increase in the arrival rate of each UE over a long recovery time, causing the system to require a transient time, which is critical to perform URLLC B5G. To overcome this issue, refiner GANs were proposed to evaluate real data and synthetic data sets by controlling the generation of real data that operate in real-time. This proposed method can generate realistic traffic flows at the packet level and guarantee reliability for RA in the long term. Furthermore, the proposed deep-RL has been emphasized for refining the GAN solution to generate high-reliability synthetic data similar to real data based on the regulator of the number of great actions in the dataset. Previous research has not introduced this number of great actions in the dataset [22], [36].

III. PROPOSED DEEP-RL FOR REFINER GAN IN REAL-TIME

Deep-RL for refiner GANs in real-time is needed to achieve large arrival rates created from a (synthetic data) generator and a (data) discriminator. The guarantee of a level of great actions in the generated datasets depends on when the Refiner Neural Network (RNN) output is similar to the input of the RNN. The optimal RNN $\mathfrak{Q}^*_{\mathbb{R}}$ is trained to obtain the output of the refiner indiscernible from a real dataset based on using a discriminator neural network over with an RNN [35], [37] as:

$$\begin{aligned} \mathfrak{Q}^*_{\mathbb{R}} &= \arg \min_{\mathfrak{Q}_{\mathbb{R}}} \max_{\mathfrak{Q}_{\mathbb{D}}} \mathcal{F}(\mathfrak{Q}_{\mathbb{R}}, \mathfrak{Q}_{\mathbb{D}}) \\ &= \arg \min_{\mathbb{R}} \mathcal{F}\left(\mathfrak{Q}_{\mathbb{R}}, \ \mathfrak{Q}^*_{\mathbb{D}}(\mathfrak{Q}_{\mathbb{R}})\right), \end{aligned} \tag{20}$$

where $\mathfrak{Q}_{\mathbb{R}}$ denotes the regular weight for the refining network, $\mathfrak{Q}_{\mathbb{D}}$ represents the regular weights of the discriminator, and $\mathcal{F}$ is a cross-entropy as the loss function. The optimal RNN $\mathfrak{Q}^*_{\mathbb{R}}$ is trained to minimize an objective prediction function when the discriminator of predicting networks attempt to discriminate the real data from the refined synthetic data, as shown in Fig. 4. The RNN controls the generation of the real-like data $\mathbb{Z}$ obtained from a distribution $\mathfrak{p}_g(\mathbb{Z})$ and generates real data as $\mathbb{x} = (\mathbb{Z}, \ \mathfrak{Q}^*_{\mathbb{D}}(\mathfrak{Q}_{\mathbb{R}})) \ \mathfrak{p}_g(\mathbb{Z})$. The discriminator $\mathfrak{Q}_{\mathbb{D}}$ at the time of training recognizes between the actual data $\mathfrak{p}(\mathbb{x})$ and the data arriving from the refined, GAN- distribution traffic data $\mathfrak{p}_g(\mathbb{Z})$ by training a function. The DNN is trained with a backpropagation function by using cross-entropy as the loss function, as follows:

$$\begin{aligned} &\min_{\mathfrak{Q}_{\mathbb{R}}} \max_{\mathfrak{Q}_{\mathbb{D}}} \mathcal{F}(\mathfrak{Q}_{\mathbb{R}}, \mathfrak{Q}_{\mathbb{D}}) \\ &\quad = \mathbb{E}_{\mathbb{x}\, \mathfrak{p}(\mathbb{x})} \left[\log \mathbb{D}(\mathbb{x}, \ \mathfrak{Q}_{\mathbb{D}})\right] \\ &\quad + \mathbb{E}_{\mathbb{Z}\, \mathfrak{p}_g(\mathbb{Z})} \left[\log\left(1 - \mathcal{F}\left(\mathbb{Z}, \ \mathfrak{Q}^*_{\mathbb{D}}(\mathfrak{Q}_{\mathbb{R}})\right)\right)\right], \end{aligned} \tag{21}$$

where $\mathfrak{p}(\mathbb{x})$ represents real distributed data and $\mathfrak{p}_g(\mathbb{Z})$ represents refined simulated data. The notation $\mathbb{E}_{\mathbb{x}\, \mathfrak{p}(\mathbb{x})}$ denotes an expectation, dependent on the output of the discriminator parametrized by $\mathfrak{Q}_{\mathbb{D}}$, when the unrefined synthetic input $\mathbb{x}$ is given, and $\mathbb{Z}$ is a real data sample from the distribution, i.e., $\mathfrak{p}_g(\mathbb{Z})$ and $(\mathbb{Z}, \ \mathfrak{Q}^*_{\mathbb{D}}(\mathfrak{Q}_{\mathbb{R}}))$. According to (21), the first term represents the discriminator's ability to learn real data distribution. The second term represents the discriminator's ability to train the coming from the refiner generator. By training the discriminator, the generative refiner can guarantee the output of the RNN as $\mathcal{F}(\mathfrak{Q}_{\mathbb{R}}) = \mathbb{E}_{\mathbb{Z}\, \mathfrak{p}_g(\mathbb{Z})} \log(\mathcal{F}(\mathbb{Z}, \ \mathfrak{Q}^*_{\mathbb{D}}(\mathfrak{Q}_{\mathbb{R}}))/(1 - \mathcal{F}(\mathbb{Z}, \ \mathfrak{Q}^*_{\mathbb{D}}(\mathfrak{Q}_{\mathbb{R}}))))$ and minimize the average correct predictions [8]. From (21), in practice, the output of the RNN cannot be controlled for sufficient training for $\mathfrak{Q}_{\mathbb{R}}$ to learn well. When $\mathfrak{Q}_{\mathbb{R}}$ is reduced, $\mathfrak{Q}_{\mathbb{D}}$ can reject training with great confidence because the output becomes incomparable to the real data obtained during the data training rather than a training $\mathfrak{Q}_{\mathbb{R}}$ used to minimize $\min_{\mathfrak{Q}_{\mathbb{R}}} = \log(1 - \mathcal{F}(\mathbb{Z}, \ \mathfrak{Q}^*_{\mathbb{D}}(\mathfrak{Q}_{\mathbb{R}})))$. The dynamics of $\mathfrak{Q}_{\mathbb{R}}$ and $\mathfrak{Q}_{\mathbb{D}}$ Provide stronger real data by controlling the level of extreme events when the refiner's output is the same as its input. The $\mathfrak{Q}^*_{\mathbb{D}}$ the network is trained to make the distribution of the real data and generated data the same by applying the convergence discriminator $\mathfrak{Q}^*_{\mathbb{D}} = \mathfrak{p}(\mathbb{x})/\left(\mathfrak{p}(\mathbb{x}) + \mathfrak{p}_g(\mathbb{x})\right)$, where $\mathfrak{p}_g(\mathbb{x})$ is the global optimality distribution of synthetic data [37]. From (21), if the output of the RNN is similar to its input, the synthetic dataset can easily be recognized by the convergence discriminator. Therefore, the global optimality of the virtual environment can prepare the trained agent and help to control the level of real extreme events when $\mathfrak{p}_g(\mathbb{x}) = \mathfrak{p}(\mathbb{x})$. The optimal discriminator $\mathfrak{Q}^*_{\mathbb{D}}$ of the real wireless environment can be written as:

$$\mathfrak{Q}^*_{\mathbb{D}}(\mathfrak{Q}_{\mathbb{R}}) = \max_{\mathfrak{Q}_{\mathbb{D}}} \mathcal{F}(\mathfrak{Q}_{\mathbb{R}}, \mathfrak{Q}_{\mathbb{D}}). \tag{22}$$

The training refiner GAN in real-time for the discriminator $(\mathfrak{Q}_{\mathbb{D}})$ maximizes any $(m, n) \in \mathbb{r}^2 \setminus \{0, 0\}$, and the function $\mathbb{y} \to m \log \mathbb{y} + n \log(1 - \mathbb{y})$ attains the maximum in the set $\mathbb{y} \in [0, \ 1]$ at $\frac{m}{m+n}$.

The training objective $\mathfrak{Q}_{\mathbb{D}}$ can be explained as the maximization of the log-likelihood for estimating the discrimination of the real data from the refined synthetic training data, which can be expressed as follows:

$$\begin{aligned} &\max_{\mathfrak{Q}_{\mathbb{D}}} \mathcal{F}(\mathfrak{Q}_{\mathbb{R}}, \mathfrak{Q}_{\mathbb{D}}) \\ &\quad = \int_{\mathbb{x}} \log \mathbb{D}(\mathbb{x}, \ \mathfrak{Q}_{\mathbb{D}}) \, \mathfrak{p}(\mathbb{x}) \, d_{\mathbb{x}} \end{aligned}$$

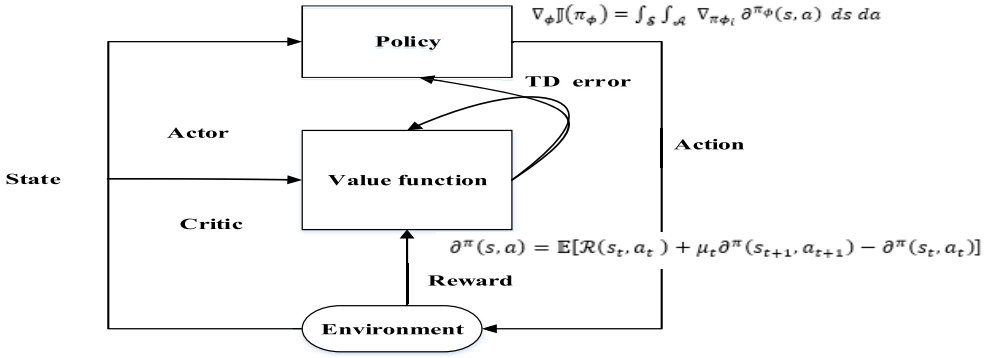

FIGURE 3. The basic structure of PGACL Actor - critic for URLLC.

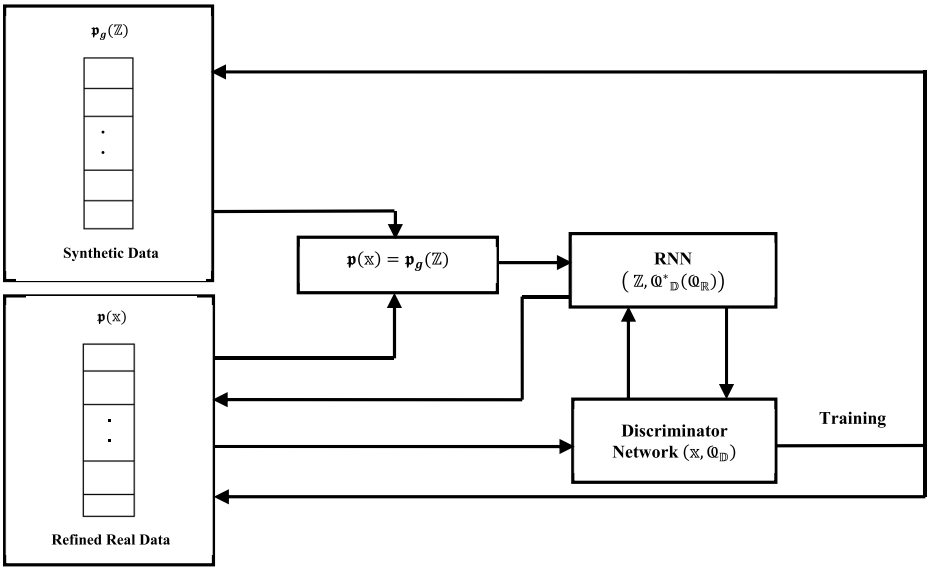

FIGURE 4. The network structure of refiner GAN.

$$+\int_{\mathbb{Z}}[\log(1-\mathcal{F}(\mathbb{Z},\ \mathbb{Q}^{*}{}_{\mathbb{D}}(\mathbb{Q}_{\mathbb{R}})))]\mathfrak{p}_{g}(\mathbb{Z})\,d\mathbb{Z}$$
$$=\int_{\mathbb{x}}\log\mathbb{D}(\mathbb{x},\mathbb{Q}_{\mathbb{D}})\mathfrak{p}(\mathbb{x})$$
$$+\ \mathfrak{p}_{g}(\mathbb{x})\log\left(1-\left(\mathbb{x},\ \mathbb{Q}^{*}{}_{\mathbb{D}}(\mathbb{Z})\right)\right)d\mathbb{Z}. \tag{23}$$

The optimal discriminator consequently tries to maximize $\max_{\mathbb{Q}_{\mathbb{D}}}\mathcal{F}(\mathbb{Q}_{\mathbb{R}},\ \mathbb{Q}_{\mathbb{D}})$ by training to distinguish between $\mathbb{Q}_{\mathbb{R}}$ and $\mathbb{Q}_{\mathbb{D}}$. The minimax RNN in (21) can be maximized by generating the best realistic samples of real data that aim to fool the best-trained offline, such as the $\mathbb{Q}_{\mathbb{D}}$, and can be reformulated as follows:

$$\max_{\mathbb{Q}_{\mathbb{D}}}\mathcal{F}(\mathbb{Q}_{\mathbb{R}},\mathbb{Q}_{\mathbb{D}})=\mathbb{E}_{\mathbb{x}\sim\mathfrak{p}(\mathbb{x})}\left[log\mathbb{D}^{*}(\mathbb{x},\mathbb{Q}_{\mathbb{D}})\right]$$
$$+\ \mathbb{E}_{\mathbb{Z}\sim\mathfrak{p}_{g}(\mathbb{Z})}\left[log\left(1-\mathcal{F}\left(\mathbb{Z},\ \mathbb{Q}^{*}{}_{\mathbb{D}}(\mathbb{Q}_{\mathbb{R}})\right)\right)\right]$$
$$\max_{\mathbb{Q}_{\mathbb{D}}}\mathcal{F}(\mathbb{Q}_{\mathbb{R}},\mathbb{Q}_{\mathbb{D}})=\mathbb{E}_{\mathbb{x}\sim\mathfrak{p}(\mathbb{x})}\left[log\mathbb{D}^{*}(\mathbb{x})\right]$$
$$+\ \mathbb{E}_{\mathbb{x}\sim\mathfrak{p}_{g}(\mathbb{x})}\left[log\left(1-\mathcal{F}\left(\mathbb{Q}^{*}{}_{\mathbb{D}}(\mathbb{Z})\right)\right)\right]$$
$$=\mathbb{E}_{\mathbb{x}\sim\mathfrak{p}(\mathbb{x})}\left[log\frac{\mathfrak{p}(\mathbb{x})}{\mathfrak{p}(\mathbb{x})+\mathfrak{p}_{g}(\mathbb{x})}\right]$$
$$+\ \mathbb{E}_{\mathbb{x}\sim\mathfrak{p}_{g}(\mathbb{x})}\left[log\frac{\mathfrak{p}(\mathbb{x})}{\mathfrak{p}(\mathbb{x})+\mathfrak{p}_{g}(\mathbb{x})}\right]. \tag{24}$$

The generator function achieves the convergence discriminator $\mathbb{Q}^{*}{}_{\mathbb{D}}$ as in (21), which satisfies $[(\mathbb{Q}^{*}{}_{\mathbb{D}}(\mathbb{Z})=\mathbb{x})\ =\ (\mathbb{D}^{*}(\mathbb{x})=\mathbb{x})]$. In addition, if and only if $\mathfrak{p}(\mathbb{x})=\mathfrak{p}_{g}(\mathbb{x})$, the discriminator $\mathbb{Q}_{\mathbb{D}}$ is allowed to minimize the average correct predictions, as shown in (24). The algorithm generates real data that operate in real-time refiner GAN training for developing deep learning.

Algorithm I handled the large amounts of data involved in URLLC and demonstrated a high order time complexity, making them suitable for real-time. The algorithm of refiner GAN training can address more action space with low complexity depending on removing the transient training time with great confidence of regular weights of the discriminator $\mathbb{Q}_{\mathbb{D}}$ because the output becomes incomparable to the real data in real-time. The computational complexity is proportional to the number of actions at every decision epoch. After an update to the RNN, the gradient of the output of the discriminator was parametrized by $\mathbb{Q}_{\mathbb{D}}$. The network that discriminates between the refined data $\mathcal{F}(\mathbb{Z},\ \mathbb{Q}^{*}{}_{\mathbb{D}}(\mathbb{Q}_{\mathbb{R}}))$, and real data are

Algorithm I: Refiner GAN Training for Enabled Deep-RL for Guarantee Traffic Packs in Real-Time

1. **For** the number of training, iterations **do**
2. **For** $\tau = 1$ to t, where t **is** the number of alternating training iterations to apply to the discriminator.
3. Initialize a dueling generation refined simulated data distributed $\mathbb{Z}$, and real data $_{\mathbb{x}}$ obtained from a distribution $\mathfrak{p}_g(\mathbb{Z})$ and the actual data $\mathfrak{p}\,(_{\mathbb{x}})$ for the RNN.
4. Initialize a dueling generator $\mathbb{R}$ and a discriminator $\mathbb{D}$ with random weights $\mathfrak{Q}_{\mathbb{R}}$ and $\mathfrak{Q}_{\mathbb{D}}$.
5. Maximize the discriminator $\mathfrak{Q}_{\mathbb{D}}$ for any $\left(\mathfrak{p}(_{\mathbb{x}}), \mathfrak{p}_g\,(_{\mathbb{x}})\right) \in \mathbb{r}^2 \setminus \{0, 0\}$,
6. Achieve the highest in the set $\mathbb{y} \in [0,\ 1]$ at $\frac{\mathfrak{p}(_{\mathbb{x}})}{\mathfrak{p}(_{\mathbb{x}})+\mathfrak{p}_g(_{\mathbb{x}})}$, and convergence discriminator $\mathfrak{Q}^*{}_{\mathbb{D}} = \frac{\mathfrak{p}(_{\mathbb{x}})}{(\mathfrak{p}(_{\mathbb{x}})+\mathfrak{p}_g(_{\mathbb{x}}))}$,
7. Update the dynamics of $\mathfrak{Q}_{\mathbb{R}}$ and $\mathfrak{Q}_{\mathbb{D}}$ with only regularization loss.
8. Achieve the optimal discriminator $\mathfrak{Q}^*{}_{\mathbb{D}}$ of the real wireless environment as $\mathfrak{Q}^*{}_{\mathbb{D}}\,(\mathfrak{Q}_{\mathbb{R}}) = \max\limits_{\mathfrak{Q}_{\mathbb{D}}} \mathcal{F}\,(\mathfrak{Q}_{\mathbb{R}}, \mathfrak{Q}_{\mathbb{D}})$,
9. **end for**
10. **if** $\mathfrak{p}(_{\mathbb{x}}) = \mathfrak{p}_g\,(_{\mathbb{x}})$
11. Minimizing the average correct predictions as shown in (24).
12. **end if**
13. Update the training $\mathfrak{Q}_{\mathbb{R}}$ and $\mathfrak{Q}_{\mathbb{D}}$ in the deep-RL network by minimizing $\min\limits_{\mathfrak{Q}_{\mathbb{R}}} = \log\left(1 - \mathcal{F}\left(\mathbb{Z},\ \mathfrak{Q}^*{}_{\mathbb{D}}\,(\mathfrak{Q}_{\mathbb{R}})\right)\right)$ and maximize the log-likelihood for estimating the discrimination of the real data, as shown in (23),
14. Perform the best-trained offline as in (24),
15. **end for**
16. Perform deep learning based on updated RNN.

more likely to be classified as real data. After several steps of training, if $\mathfrak{Q}_{\mathbb{R}}$ and $\mathfrak{Q}_{\mathbb{D}}$ generate more realistic real data that reach a point at which both cannot improve because $\mathfrak{p}_g\,(_{\mathbb{x}}) = \mathfrak{p}\,(_{\mathbb{x}})$. The discriminative neural network has to allocate training data to establish efficient decisions for refiner networks. Let L, C_0, and C_l denote the training layers in trained deep-RL models, proportional to the number of hidden layers and dimensions of the output utilized in deep-RL, respectively. The complexity in every training for every agent is computed by $\mathcal{O}\left(\mathrm{C}_0\mathrm{C}_l + \sum_{l=1}^{L-1} \mathrm{C}_l\mathrm{C}_{l+1}\right)$ at every training procedure. In real data training, every TTI has epochs $\mathbb{N}^{epoch}$ with every epochs being time slot t, and every trained model is finished over iterations. Therefore, the convergence and the network has 3 agents with 3 trained deep-RL models reached. Hence, the total complexity is $\mathcal{O}\left(3\mathbb{N}^{epoch}t + \sum_{l=1}^{L-1} \mathrm{C}_l\mathrm{C}_{l+1}\right)$. The high deep-RL training complexity phase is achieved offline and the number of actions for a limited number of epochs at a powerful unit as the BS [38], [39].

IV. SIMULATION RESULTS

In this section, B5G URLLC is evaluated by achieving large labels of the real dataset in real-time and the number of packets generated during the interarrival time. In addition, the number of arriving packets from the dataset, which is comparable in length and interarrival time for every UE is proposed. The proposed algorithms are evaluated using different benchmark metrics. These metrics are commonly used to evaluate the performance of existing approaches to performance deep-RL for refiner GANs [15] [22] [30]. The main simulation parameters are listed in Table 2.

TABLE 2. Simulation parameters.

Parameter	Value
i	20
B	180 KHz
$\mathcal{N}_0$	-173.9 dBm/Hz
$\mathcal{P}$	4 W
θ	10 ms
URLLC packet size	64
TTI length	1 ms
Carrier frequency	2 GHz
Total bandwidth	40 MHz
Cell radius	500

A. INTELLIGENT URLLC-B5G SCHEDULING AND ACTION SPACE REDUCER FOR RB AND PA

The intelligent agent can improve smart packet transmission scheduling for URLLC. The average transmission delay is very sensitive to the size of the data packet. There is high reliability of short packets as the reliability decreases with an increase in the average packet size, as shown in Fig. 5. The proposed deep-RL ensured the reliability of the UEs by keeping a limited blocklength close to 1 at R_{min} = 2.5Mbps, and the URLLC is more reliable in increasing the data rate when the packet size is small because the delay increases with the increase in the packet size. In Fig. 5, it can be observed that the proposed deep-RL for refiner GAN can provide high reliability and low latency with a high data rate based on the decreasing packet size due to limited bandwidth. The proposed deep-RL for refiner GAN can randomly reduce the packet loss generated by different multiuser arrival at a BS. Compared with [30], Fig.4, packet size can provide more reliability for traffic with shorter packet sizes. The interarrival time between packets and queueing delay violation depends on the performance of E2E in terms of reliability. The deep-RL can decrease errors quickly based on the average training loss and validation loss for the ANN. The proposed deep-RL for refiner GAN could keep the URLLC reliability higher than 97% at R_{min} = 2.5 Mbps, while the average achievable rate denotes failure to maintain suitable reliability, which fell to a value lower than 60%. From Fig. 6, the deep-RL algorithm for reducing transmission power relies on the current state of the decision policy to obtain intelligent transmission for every UE. The training sample time is insufficient if the packet arrival rate is high. Therefore, training time requires the AI in URLLC to transmit more packets in real-time. The proposed deep-RL reduces the total power while holding the lowest average rate of

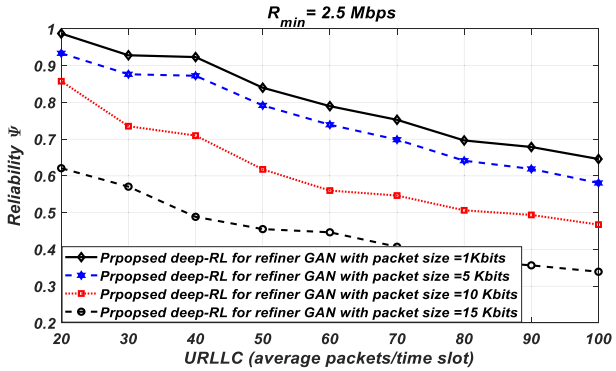

FIGURE 5. Reliability for an average number of arriving URLLC packets.

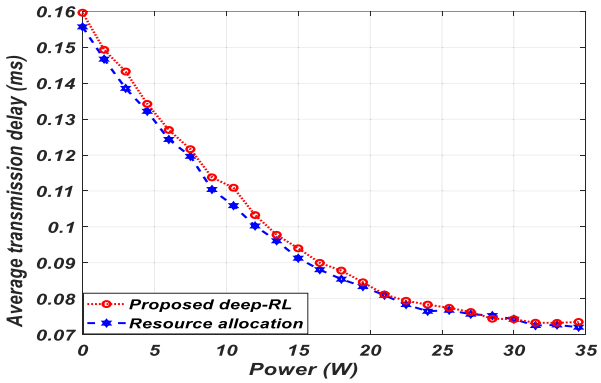

FIGURE 6. Average transmission delay for total transmission power.

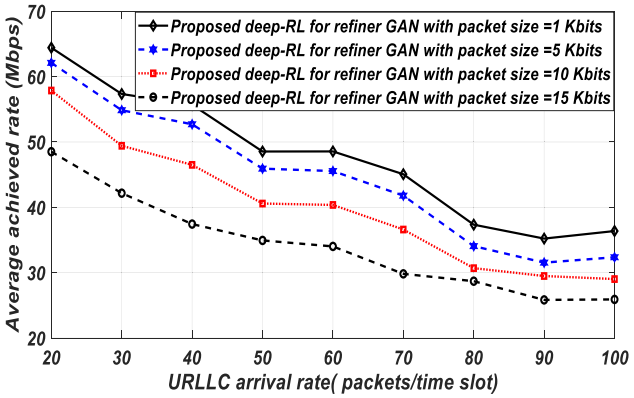

FIGURE 7. The average achieved rate for different arrival packets.

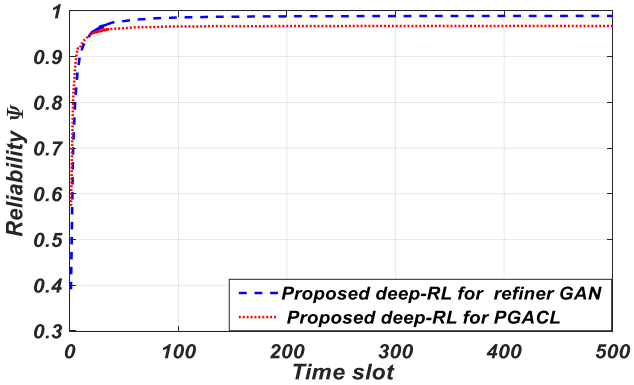

FIGURE 8. Evaluation of the reliability of URLLC for a time slot.

every UE and controls the packet transmission for a maximum delay. The deep-RL achieved the minimum average transmission delays as low as 0.16 ms, as shown in Fig.6, by finite transmit power to many URLLC UEs. As shown in Fig. 6, the optimal RA can provide the minimum transmission delay while minimizing transmit power or providing the same power as proposed in the deep-RL. The minimum average delay obtained by the amount of power is =35 W, compared with [22], Fig.7, where the minimum average delay with the power becomes more significant than 50 W. The average transmission delay is nearly flat in refiner GAN when the transmission power level increases. The training of deep-RL with the PGACL algorithm can control the transmission duration of each packet by ensuring a minimized transmit power to more URLLC.

B. OPTIMAL RATE ALLOCATION FOR POLICY GRADIENT

The policy gradient is significant for selecting an optimal rate allocation to solve the RB, improving transmission packet to guarantee high E2E reliability, and guaranteeing a good policy with a closer convergence rate, which depends on the reduced action space for every UE. Fig. 7 shows the impact of the URLLC arrival rate versus the achieved average rate. The average data rate depends on the training of the deep-RL based on the real data distribution. The proposed deep-RL for the refiner GAN could grow up to 65 Mbps for every 20 UE. The average rate of 65 Mbps was achieved when the average URLLC load was 20 (packet/time slot). The rate decreased to 48 Mbps when the average URLLC load was increased to 100 packets/time slot.

The packet size is large, and the rate will decrease because delay violations have caused the packet loss probability. The proposed deep-RL of URLLC for refiner GAN becomes more reliable by generating the best realistic real data samples to fool the best-trained offline. Fig.7, the delay increases with increasing packet size. When the packet size is increased, the average delay metric is not suitable in real-time services (delay of packets would lie in a small range). However, the proposed deep-RL for refiner GAN has achieved the average achievable data rate that varied from 28 Mbps to 48 Mbps by growing the average URLLC load from 20 to 100 packets/time slot. Figure 8 evaluates the reliability with the total number of UEs in the system at the time slot. The reliability did not decrease with an increase in the running time from zero to 500-time slots but slowly increased due to its strict E2E QoS requirement. From Fig. 8, the refiner GAN can achieve 0.999 with around 400 epochs. While in [15] Fig.6, only 400 epochs are needed to achieve the 0.98 accuracies. As a result, the system provides high reliability and a higher data rate based on localizing in time with TTI. From (11), the data package size was too small to achieve high reliability, and the TTI was short. From Fig. 8, the proposed deep-RL for the refiner GAN provides good performance by enabling fast convergence, achieving a better response, and improving time by controlling the generation of real data during refiner GAN

training for the first time slot in real-time. When the reliability in the PGACL is smaller than that in the refiner GAN due to the use of PGACL, the issue of a sudden increase in the arrival rate of each UE with a long recovery time occurred, and the system needed a transient time, which is critical to perform B5G URLLC.

C. DEEP-RL FOR REFINER GAN IN REAL-TIME

In this subsection, the conditional refiner GAN for large iterative training can provide a desirable action in each decision epoch, reduce the order of time complexity and control the great action space involved in URLLC in real-time. Figure 9 shows the relation between E2E target latency versus the delay reliability in terms of the effect of the maximum bandwidth. From Fig. 9, the high reliability and low latency were achieved at a high rate following allocating the higher bandwidth to the system. The E2E latency increases with increasing reliability because of the tradeoff between latency and reliability. When the bandwidth increased from 35 MHz to 45 MHz, the latency decreased, making it difficult to guarantee the latency and reliability. Moreover, the minimum 35 MHz bandwidth could increase the rate of each UE as per requirement without sacrificing the E2E latency and reliability of URLLC. The curve in Fig. 9, will increase the reliability because the URLLC data rate based on finite block-length provides more reliability to the traffic with shorter packet sizes and protects UEs at bad channel states to satisfy the looser reliability requirements. The proposed deep-RL for the refiner GAN can achieve a reliability of 99.9999% and a latency of less than 1.4ms.

Figure 10 shows the training GANs and presents the actual refined training data. The discriminator $\mathfrak{G}_{\mathbb{D}}$ and refiner losses decreased to a stability loss based on the real dataset in terms of the training loss. The optimal refiner GAN ($\mathfrak{G}^*_{\mathbb{R}}$) has been trained to provide stable loss over time for discriminator loss and refiner loss. The loss did not decrease (loss was slowly decreasing) over time per training number of epochs due to control over the output of the RNN for sufficient training so $\mathfrak{G}_{\mathbb{R}}$ can learn well. Only 2000 epochs (training samples) are needed for the deep-RL refiner GAN to achieve stable loss. From Fig.10, the deep-RL for refiner GAN trains data samples to minimize an accurate prediction, as shown in (21), and generates the best realistic samples of real data that aim to provide the stable loss and generate high-reliability synthetic data similar to real data when the prediction tries to discriminate between the two sets of data.

V. CONCLUSION

This paper uses the proposed deep-RL framework to determine the E2E reliability and E2E latency for every UE based on a dynamically predicted traffic model, jointly allocating RBs and power minimization under the constraint arrival rate B5G URLLC in the DL of an OFDMA system. The joint problem of minimization power was formulated with rate constraints of UEs, ultrahigh reliability, and ultralow latency to operate in highly reliable systems. Using those predictions

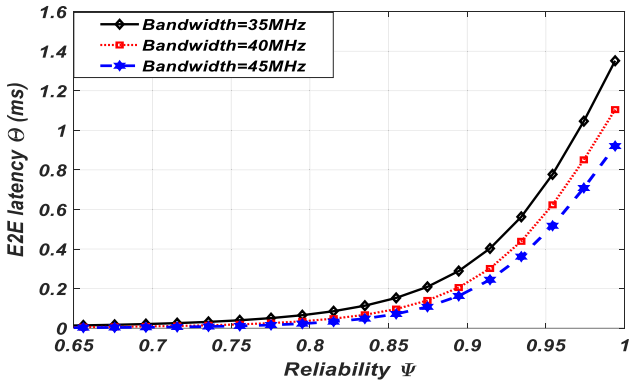

FIGURE 9. Effect of bandwidth on E2E latency and reliability of URLLC.

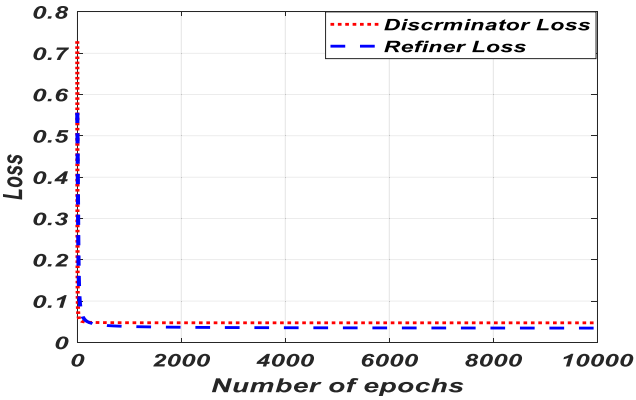

FIGURE 10. Loss progress for GAN vs. several training epochs.

in the RA process, the proposed deep-RL can predict UEs' traffic. To solve this problem, it is necessary to guarantee the desired rate for every action by addressing the large state space and large action space based on the proposed PGACL that can provide a good policy with a closer convergence rate and a low computational cost. Finally, to improve highly reliable systems, the E2E reliability and latency of every UE were used as feedback on the proposed refiner GANs to provide sufficient traffic packs in real-time by avoiding leaving a large number of training samples unlabeled.As a result, the proposed deep-RL for the refiner GAN can minimize unlabeled real traffic, which needs to learn faster and has a shorter transition period. From the simulation results, the proposed deep-RL verifies that refiner GANs can satisfy the stringent requirements of URLLC and high rate based on the omission of the untrained agent and the synthetically trained agent, which takes a longer transient training time. Our future work will investigate the improved intelligent smart packet transmission scheduling and fairness of UEs for the internet of everything in URLLC-B5G.

REFERENCES

[1] S. F. Abedin, A. K. Bairagi, M. S. Munir, N. H. Tran, and C. S. Hong, "Fog load balancing for massive machine type communications: A game and transport theoretic approach," *IEEE Access*, vol. 7, pp. 4204–4218, 2019.

[2] A. K. Bairagi, M. S. Munir, M. Alsenwi, N. H. Tran, S. S. Alshamrani, M. Masud, Z. Han, and C. S. Hong, "Coexistence mechanism between eMBB and uRLLC in 5G wireless networks," *IEEE Trans. Commun.*, vol. 69, no. 3, pp. 1736–1749, Mar. 2021.

[3] M. Alsenwi, N. H. Tran, M. Bennis, A. K. Bairagi, and C. S. Hong, "EMBB-URLLC resource slicing: A risk-sensitive approach," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 740–743, Apr. 2019.

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

[5] 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 Trans. Wireless Commun.*, vol. 20, no. 7, pp. 4585–4599, Jul. 2021.

[6] H. Ren, C. Pan, Y. Deng, M. Elkashlan, and A. Nallanathan, "Resource allocation for secure URLLC in mission-critical IoT scenarios," *IEEE Trans. Commun.*, vol. 68, no. 9, pp. 5793–5807, Sep. 2020.

[7] W. R. Ghanem, V. Jamali, Q. Zhang, and R. Schober, "Joint uplink-downlink resource allocation for OFDMA-URLLC MEC systems," in *Proc. IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–7.

[8] Q. He, Y. Hu, and A. Schmeink, "Resource allocation for ultra-reliable low latency communications in sparse code multiple access networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2018, no. 1, pp. 1–9, Dec. 2018.

[9] A. Salh, L. Audah, N. S. M. Shah, A. Alhammadi, Q. Abdullah, Y. H. Kim, S. A. Al-Gailani, S. A. Hamzah, B. A. F. Esmail, and A. A. Almohammedi, "A survey on deep learning for ultra-reliable and low-latency communications challenges on 6G wireless systems," *IEEE Access*, vol. 9, pp. 55098–55131, 2021.

[10] W. Yang, G. Durisi, T. Koch, and Y. Polyanskiy, "Quasi-static multiple-antenna fading channels at finite blocklength," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4232–4265, Jul. 2014.

[11] S. Xu, T.-H. Chang, S.-C. Lin, C. Shen, and G. Zhu, "Energy-efficient packet scheduling with finite blocklength codes: Convexity analysis and efficient algorithms," *IEEE Trans. Wireless Commun.*, vol. 15, no. 8, pp. 5527–5540, Aug. 2016.

[12] C. Sun, C. She, C. Yang, T. Q. S. Quek, Y. Li, and B. Vucetic, "Optimizing resource allocation in the short blocklength regime for ultra-reliable and low-latency communications," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 402–415, Jan. 2019.

[13] C. Sun, C. She, and C. Yang, "Unsupervised deep learning for optimizing wireless systems with instantaneous and statistic constraints," May 2020, *arXiv:2006.01641*.

[14] H. Yang, K. Zhang, K. Zheng, and Y. Qian, "Joint frame design and resource allocation for ultra-reliable and low-latency vehicular networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3607–3622, May 2020.

[15] R. Dong, C. She, W. Hardjawana, Y. Li, and B. Vucetic, "Deep learning for radio resource allocation with diverse quality-of-service requirements in 5G," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2309–2324, Apr. 2021.

[16] Z. Hou, C. She, Y. Li, T. Q. Quek, and B. Vucetic, "Burstiness-aware bandwidth reservation for ultra-reliable and low-latency communications in tactile internet," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2401–2410, Nov. 2018.

[17] W. K. Lai and C. L. Tang, "QoS-aware downlink packet scheduling for LTE networks," *Comput. Netw.*, vol. 57, no. 7, pp. 1689–1698, May 2013.

[18] J. Li, H. Gao, T. Lv, and Y. Lu, "Deep reinforcement learning based computation offloading and resource allocation for MEC," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Barcelona, Spain, Apr. 2018, pp. 1–6.

[19] A. Azari, M. Ozger, and C. Cavdar, "Risk-aware resource allocation for URLLC: Challenges and strategies with machine learning," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 42–48, Mar. 2019.

[20] H. Yang, Z. Xiong, J. Zhao, D. Niyato, C. Yuen, and R. Deng, "Deep reinforcement learning based massive access management for ultra-reliable low-latency communications," *IEEE Trans. Wireless Commun.*, vol. 20, no. 5, pp. 2977–2990, May 2021.

[21] Y. Hua, R. Li, Z. Zhao, X. Chen, and H. Zhang, "GAN-powered deep distributional reinforcement learning for resource management in network slicing," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 2, pp. 334–349, Feb. 2020.

[22] A. T. Z. Kasgari, W. Saad, M. Mozaffari, and H. V. Poor, "Experienced deep reinforcement learning with generative adversarial networks (GANs) for model-free ultra reliable low latency communication," *IEEE Trans. Commun.*, vol. 69, no. 2, pp. 884–899, Feb. 2021.

[23] F. Naeem, S. Seifollahi, Z. Zhou, and M. Tariq, "A generative adversarial network enabled deep distributional reinforcement learning for transmission scheduling in internet of vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4550–4559, Jul. 2021.

[24] J. Zhu, Y. Song, D. Jiang, and H. Song, "A new deep-Q-learning-based transmission scheduling mechanism for the cognitive Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2375–2385, Aug. 2018.

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

[26] S. Schiessl, H. Al-Zubaidy, M. Skoglund, and J. Gross, "Delay performance of wireless communications with imperfect CSI and finite-length coding," *IEEE Trans. Commun.*, vol. 66, no. 12, pp. 6527–6541, Dec. 2018.

[27] B. Fehrman, B. Gess, and A. Jentzen, "Convergence rates for the stochastic gradient descent method for non-convex objective functions," *J. Mach. Learn. Res.*, vol. 21, no. 136, pp. 1–48, 2020.

[28] Q. Huang, X. Xie, and M. Cheriet, "Reinforcement learning-based hybrid spectrum resource allocation scheme for the high load of URLLC services," *EURASIP J. Wireless Commun. Netw.*, vol. 2020, no. 1, pp. 1–21, Dec. 2020.

[29] A. M. Koushik, F. Hu, and S. Kumar, "Intelligent spectrum management based on transfer actor-critic learning for rateless transmissions in cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 5, pp. 1204–1215, May 2018.

[30] A. T. Z. Kasgari and W. Saad, "Model-free ultra reliable low latency communication (URLLC): A deep reinforcement learning framework," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Shanghai, China, May 2019, pp. 1–6.

[31] T. Terlaky and J. Zhu, "Comments on 'dual methods for nonconvex spectrum optimization of multicarrier systems,'" *Optim. Lett.*, vol. 2, no. 4, pp. 497–503, Aug. 2008.

[32] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep reinforcement learning based resource allocation for V2V communications," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3163–3173, Apr. 2019.

[33] E. F. Morales and J. H. Zaragoza, "An introduction to reinforcement learning," in *Decision Theory Models for Applications in Artificial Intelligence*. Hershey, PA, USA: IGI Global, 2012, pp. 63–80.

[34] J. N. Tsitsiklis and B. Van Roy, "Feature-based methods for large scale dynamic programming," *Mach. Learn.*, vol. 22, nos. 1–3, pp. 59–94, 1996.

[35] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, Oct. 2020.

[36] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 2242–2251.

[37] T. Zhao, "Enhanced experience generation for reinforcement learning pre-training in telecommunication systems," M.S. thesis, Dept. Comput. Sci., School Elect. Eng. Comput. Sci., KTH Roy. Inst. Technol., Stockholm, Sweden, Tech. Rep., 2020.

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

[39] A. M. Seid, G. O. Boateng, B. Mareri, G. Sun, and W. Jiang, "Multi-agent DRL for task offloading and resource allocation in multi-UAV enabled IoT edge network," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 4, pp. 4531–4547, Dec. 2021.

ADEEB SALH received the Bachelor of Electrical and Electronic Engineering from IBB University, Ibb, Yemen, in 2007, and the master's and Ph.D. degrees in electrical and electronic engineering from the University Tun Hussein Onn Malaysia, in 2015 and 2020, respectively. From 2007 to 2012, he worked as a Lecturer Assistant with the Yareem Community College. He is currently a Postdoctoral Researcher at the Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia. His research interests include 5G, 6G wireless communications, massive MIMO, artificial intelligence (AI), and the Internet of Things (IoT).

LUKMAN AUDAH (Member, IEEE) received the Bachelor of Engineering degree in telecommunications from the Universiti Teknologi Malaysia, in 2005, and the M.Sc. degree in communication networks and software and the Ph.D. degree in electronic engineering from the University of Surrey, U.K. He is currently a Senior Lecturer with the Communication Engineering Department, Universiti Tun Hussein Onn Malaysia. His research interests include wireless and mobile communications, internet traffic engineering, network system management, data security, and satellite communications.

KWANG SOON KIM (Senior Member, IEEE) received the B.S. *(summa cum laude)*, M.S.E., and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in February 1994, February 1996, and February 1999, respectively. From March 1999 to March 2000, he was a Postdoctoral Researcher with the Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA, USA. From April 2000 to February 2004, he was a Senior Member of Research Staff with the Mobile Telecommunication Research Laboratory, Electronics and Telecommunication Research Institute, Daejeon. Since March 2004, he has been with the Department of Electrical and Electronic Engineering, Yonsei University, Seoul, South Korea, where he is currently a Professor. His research interests include signal processing, communication theory, information theory, and stochastic geometry applied to wireless heterogeneous cellular networks, wireless local area networks, wireless D2D networks, wireless ad hoc networks, and new radio access technologies for 5G. He was a recipient of the Postdoctoral Fellowship from Korea Science and Engineering Foundation (KOSEF) in 1999. He received the Outstanding Researcher Award from the Electronics and Telecommunication Research Institute (ETRI) in 2002, the Jack Neubauer Memorial Award (Best System Paper Award, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY) from IEEE Vehicular Technology Society in 2008, and the LG Research and Development Award: Industry-Academic Cooperation Prize, LG Electronics, in 2013. From 2006 to 2012, he served as an Editor for the *Journal of the Korean Institute of Communications and Information Sciences* (KICS). From 2013 to 2016, he served as the Editor-in-Chief for the *Journal of KICS*. Since 2008, he has been serving as an Editor for the *Journal of Communications and Networks* (JCN). From 2009 to 2014, he served as an Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.

SAEED HAMOOD ALSAMHI received the B.Eng. degree from the Department of Electronic Engineering (Communication Division), IBB University, Yemen, in 2009, and the M.Tech. degree in communication systems and the Ph.D. degree from the Department of Electronics Engineering, Indian Institute of Technology (Banaras Hindu University), IIT (BHU), Varanasi, India, in 2012 and 2015, respectively. In 2009, he worked as a Lecturer Assistant with the Engineering Faculty, IBB University. After that, he held a postdoctoral position with the School of Aerospace Engineering, Tsinghua University, Beijing, China, in optimal and smart wireless network research and its applications to enhance robotics technologies. Since 2019, he has been an Assistant Professor. He is currently a MSCA SMART 4.0 Fellow at the Athlone Institute of Technology, Athlone, Ireland. He has published more than 80 articles in high reputation journals in IEEE, Elsevier, Springer, Wiley, and MDPI publishers. His research interests include B5G, green communications, the green Internet of Things, QoE, QoS, multi-robot collaboration, blockchain technology, and space technologies (high altitude platform, drone, and tethered balloon technologies).

MOHAMMED A. ALHARTOMI (Member, IEEE) received the Ph.D. degree in electronic and electrical engineering from Leeds University, U.K., in 2016. He is currently an Assistant Professor with the Department of Electrical Engineering, University of Tabuk. His research interests include wireless and mobile communications, signal processing, optical wireless systems design, and visible light communications.

QAZWAN ABDULLAH (Member, IEEE) was born in Taiz, Yemen. He received the bachelor's and Master of Science degrees in electrical and electronic engineering from the Universiti Tun Hussein Onn Malaysia (UTHM), in 2013 and 2015, respectively. He has more than 40 scientific publications. He is currently a Research Assistant with the UTHM. His research interests include control theory, adaptive fuzzy logic controller, mobile communication (5G/6G), fuzzy logic control and its applications, motor drive, electric vehicle, and antenna filter design.

FARIS A. ALMALKI received the B.Sc. degree in computer engineering from Taif University, the M.Sc. degree in broadband and mobile communication networks from Kent University, and the Ph.D. degree in wireless communication networks from Brunel University London. He is currently an Associate Professor of wireless communications and drones with the Computer Engineering Department, Taif University, a Research Fellow with the Department of Electronic and Computer Engineering, Brunel University London. His research interests include unmanned aerial vehicles (UAVs) and satellites and their application in ad hoc wireless networks. Besides, topics related to artificial intelligence, the Internet of Healthcare Things, machine learning, encrypted wireless communications, and emerging trends and applications. He is a member of the IEEE Communication Society. He is a reviewer in many respected journals and publishers, including Springer, IEEE, Elsevier, and Oxford Press.

HANEEN ALGETHAMI (Senior Member, IEEE) received the B.Sc. degree from Taif University, Saudi Arabia, in 2006, and the M.Sc. degree in advanced computing science and the Ph.D. degree in computer science from the University of Nottingham, U.K., in 2012 and 2017, respectively. Since 2018, she has been an Assistant Professor with the Computer Science Department, Taif University. Her research interests include real-world applications of combinatorial problems while using search algorithms and optimization techniques.