Received 27 April 2025; accepted 12 May 2025. Date of publication 15 May 2025; date of current version 23 May 2025.

Digital Object Identifier 10.1109/OJCOMS.2025.3570444

Multi-Agent Quantum Reinforcement Learning for Adaptive Transmission in NOMA-Based Irregular Repetition Slotted ALOHA

WON JAE RYU [1] (Member, IEEE), JAE-MIN LEE [2] (Member, IEEE), AND DONG-SEONG KIM [2] (Senior Member, IEEE)

[1]ICT Convergence Research Center, Kumoh National Institute of Technology, Gumi 39177, Republic of Korea
[2]The IT Convergence Engineering, Kumoh National Institute of Technology, Gumi 39177, Republic of Korea

CORRESPONDING AUTHOR: D.-S. KIM (e-mail: dskim@kumoh.ac.kr)

This work was supported in part by the Ministry of Science and ICT (MSIT), South Korea, through the Innovative Human Resource Development for Local Intellectualization Program, supervised by the Institute for Information and Communications Technology Planning and Evaluation (IITP) under Grant IITP-2025-RS-2020-II201612 (25%); in part by the Priority Research Centers Program under Grant 2018R1A6A1A03024003 (25%); in part by the Information Technology Research Center (ITRC) under Grant IITP-2025-RS-2024-00438430 (25%); and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education, Science and Technology under Grant 2022R1I1A1A01069334 (25%).

ABSTRACT This paper proposes a Multi-Agent Quantum Deep Reinforcement Learning (MA-QDRL) framework to optimize uplink access in Irregular Repetition Slotted ALOHA with Non-Orthogonal Multiple Access (IRSA-NOMA) systems under practical constraints such as finite frame lengths and fading channels. IRSA enhances reliability by allowing devices to transmit multiple packet replicas across random time slots, while NOMA increases spectral efficiency through power-domain multiplexing with successive interference cancellation (SIC). As a benchmark, Contention Resolution Diversity Slotted ALOHA (CRDSA) improves traditional ALOHA through packet repetition and interference cancellation, maintaining solid performance under moderate network loads; however, its efficiency gradually declines under heavier traffic due to increased collisions and limited adaptability. Conventional multi-agent deep reinforcement learning (MA-CDRL) approaches have been explored to address coordination challenges in such environments. Nevertheless, these models often experience scalability limitations and unstable convergence as the number of agents increases. To overcome these challenges, MA-QDRL integrates variational quantum circuits into agents' policy networks to improve learning efficiency and convergence. Simulation results demonstrate that MA-QDRL reduces packet loss rate by 63.2% compared to classical DRL and by 39.2% compared to CRDSA-NOMA under high network load conditions ($G = 1.0$), confirming its effectiveness in highly congested IoT environments. In addition, it reduces overall computational cost compared to MA-CDRL. These results highlight the potential of MA-QDRL as a scalable and efficient solution for dynamic multi-agent wireless access in next-generation IoT networks.

INDEX TERMS Quantum neural network (QNN), multi-agent reinforcement learning, quantum deep reinforcement learning (QDRL), uplink random access, IRSA-NOMA Optimization, finite-length frame, adaptive repetition control, wireless IoT networks.

I. INTRODUCTION

IN THE Internet of Things (IoT) ecosystem, uplink random access plays a critical role in enabling efficient and reliable communication between a massive number of devices and the network infrastructure [1], [2], [3]. With the rapid growth of IoT applications, managing uplink traffic effectively has become essential to avoid collisions, reduce latency, and improve overall resource utilization [4], [5], [6], [7].

To address the limitations of traditional random access protocols under high device density, advanced techniques such as Irregular Repetition Slotted ALOHA (IRSA) [8] and Non-Orthogonal Multiple Access (NOMA) [9], [10]

have been extensively explored. IRSA enhances reliability by allowing devices to transmit multiple replicas of their packets across randomly selected time slots, following an irregular repetition pattern. This approach spreads the traffic load more evenly and enables the receiver to resolve collisions using successive interference cancellation (SIC) techniques. By carefully designing the repetition probabilities, IRSA improves system performance compared to traditional ALOHA-based schemes, particularly under high-density conditions. Similarly, NOMA addresses resource allocation challenges by enabling multiple devices to share the same time, frequency, or code resources. By separating users in the power domain with distinct power levels assigned to each device's signal, NOMA improves spectral efficiency and supports simultaneous connections for a large number of devices. At the receiver, SIC is used to decode the superimposed signals, making NOMA especially suitable for IoT networks [11].

Meanwhile, recent advances in quantum computing have sparked growing interest in applying quantum machine learning (QML) techniques to various IoT problems. For example, a quantum support vector machine-based intrusion detection system was proposed in [12], achieving high detection accuracy while requiring significantly less training data than conventional ML and DL models. A quantum-inspired optimization model for improving data accuracy and temporal efficiency in real-time IoT sensing environments was introduced in [13]. In addition, a quantum-inspired particle swarm optimization algorithm was developed in [14] for efficient IoT service placement in edge computing systems, optimizing throughput, delay, energy consumption, and load balancing. These studies collectively demonstrate the versatility and potential of quantum-inspired learning and optimization techniques in diverse IoT applications, from security and sensing to resource orchestration. Moreover, a recent study proposed and evaluated three quantum autoencoder (QAE)–based anomaly-detection frameworks that outperformed all classical baselines on IoT-traffic benchmark datasets, further underscoring the promise of QML for network-security applications [15].

While IRSA and NOMA provide significant improvements over traditional random access protocols, they still rely on static or probabilistic transmission strategies, which may not adapt efficiently to rapidly changing network conditions. Moreover, existing DRL-based approaches often suffer from slow convergence and require extensive training time, particularly in multi-agent scenarios where coordination among agents is essential [16]. In such settings, classical DRL methods can struggle with stability and sample inefficiency, especially when scaling to large numbers of agents and diverse network states. These challenges highlight the need for a more scalable, decentralized, and sample-efficient learning framework that can dynamically adapt transmission strategies in IRSA-NOMA systems.

Building on these foundations, this paper proposes an optimization framework for IRSA-NOMA systems under practical constraints such as finite frame length and fading channels. Leveraging the power of quantum computing, we design a Multi-Agent Quantum Deep Reinforcement Learning (MA-QDRL) approach based on Quantum Neural Networks (QNNs) [17], [18], aiming to improve learning efficiency, enhance convergence, and minimize packet loss in highly dynamic IoT environments.

The main contributions of this paper are as follows:

- We propose a novel MA-QDRL-based optimization framework for IRSA-NOMA with finite frame length, considering realistic system constraints such as buffer states, fading channel conditions, and packet losses.
- The proposed approach dynamically adjusts the number of packet repetitions based on real-time network conditions, unlike fixed probabilistic repetition in conventional IRSA.
- To support decentralized decision-making, MA-QDRL enables each node to learn and optimize its transmission strategy without centralized coordination.
- The proposed method reduces retransmissions and network congestion, outperforming classical DRL and baseline IRSA-NOMA approaches in terms of reliability and efficiency.

The remainder of this paper is structured as follows: Section I reviews the related work. Section III presents the system model. Section IV details the MA-QDRL approach. Section V describes the simulation results. Section VI concludes the paper and discusses future research directions.

II. RELATED WORK

This section reviews prior work on uplink access optimization using IRSA and NOMA, the application of learning-based methods, and recent advances in quantum learning techniques for wireless communications. A comprehensive comparison of the reviewed studies is summarized in Table 1.

A. IRSA-NOMA FOR UPLINK ACCESS

IRSA-NOMA integrates the irregular repetition strategy of IRSA with the power-domain multiplexing of NOMA, enabling multiple users to share the same time slot with distinct power levels and repeated transmissions. This approach has been investigated in recent works such as [19], [20], focusing primarily on enhancing collision resolution and throughput in high-density scenarios. Specifically, [19] prioritizes emergency device traffic in massive IoT networks, while [20] explores IRSA-NOMA's applicability to satellite communication systems. However, both studies assume ideal conditions with infinite frame lengths and do not consider practical limitations such as fading channels or finite frame resources.

TABLE 1. Comparison of related works on IRSA-NOMA and quantum learning for wireless systems.

Reference	Finite Frame Length	Channel Condition	AI-Based Traffic Control	QNN
[19]	X	X	X	X
[20]	X	X	X	X
[21]	✓	X	X	X
[22]	✓	X	✓	X
[23]	✓	X	X	X
[24]	✓	X	X	X
[25]	X	X	X	X
[26]	X	X	X	X
[27]	X	X	X	X
[28]	X	X	X	X
[29]	X	X	X	X
[30]	X	✓	X	X
[31]	X	X	X	X
[32]	X	X	X	X
[33]	X	X	X	X
[34]	X	X	X	X
[35]	X	✓	✓	X
[36]	X	✓	✓	X
[37]	X	X	△	✓
[38]	X	X	△	✓
[39]	X	X	△	✓
Proposed	✓	✓	✓	✓

B. IRSA WITH FINITE LENGTH AND PRACTICAL CONSTRAINTS

Recent studies have attempted to address practical limitations in IRSA systems. Studies such as [21], [22] explored retransmission and decoding strategies under finite frame lengths. However, these studies do not integrate NOMA or consider fading effects. A more recent effort [23] included short-packet decoding error analysis under fading but still assumed infinite frame length. A notable study [24] extended IRSA with multiuser detection (MUD) and conducted a density evolution analysis valid for finite frame lengths. This work investigated both frame-structured and frameless asynchronous access modes, incorporating bounded delay and memory constraints, and demonstrated that moderate values of MUD degree k could significantly reduce packet loss and decoding delay. However, this approach relies on multiuser detection at the receiver side, which may not be scalable or feasible for resource-constrained IoT deployments. Furthermore, while the analysis considers finite-length frames, it does not support adaptive or learning-based optimization strategies. NOMA integration is also not addressed, limiting its applicability to high-efficiency random access scenarios in practical wireless networks.

In parallel, several enhancements originally developed for Contention Resolution Diversity Slotted ALOHA (CRDSA)—a precursor to IRSA with fixed repetition degrees—have inspired further advances in IRSA optimization [25]. One study [26] proposed physical-layer enhancements for CRDSA, optimizing throughput and decoding under high-density access. Another approach [27] introduced transmit power diversity, enabling better utilization of capture effects in random access scenarios. While these works focus on CRDSA, their techniques are conceptually extendable to IRSA, especially in scenarios involving SIC and power-domain differentiation. However, both studies assume idealized channel conditions and static repetition policies, limiting their applicability to dynamic wireless environments that require real-time adaptive control. A recent study [28] proposed an adaptive transmission control method for CRDSA/IRSA in satellite IoT systems by adjusting terminal-specific transmission probabilities based on the number of available channels. While this approach improves fairness under heterogeneous channel availability, its rule-based adaptation relies on predefined heuristics and lacks dynamic learning or real-time optimization, limiting its applicability in highly dynamic network environments.

Furthermore, IRSA under total transmit power constraints in satellite IoT systems has been investigated [29]. The results demonstrated that degree distribution can be optimized under power-limited environments. Yet, the proposed optimization was designed for fixed satellite topologies and does not accommodate stochastic channel variations or learning-based decision-making, which are essential in terrestrial IoT networks.

More recently, a Spatial Group-based NOMA-IRSA (SG-NOMA-IRSA) framework was proposed in [30], which introduces adaptive degree distribution and user grouping based on spatial locations to reduce access collisions and enhance SIC efficiency. While this approach improves throughput and latency in static WSN scenarios, it assumes a centralized grouping scheme and lacks decentralized adaptability, which, although effective in static WSN scenarios, limits its scalability under dynamic traffic and topology changes.

Other studies have explored throughput bounds of NOMA-ALOHA under ideal conditions [31], but they primarily focus on theoretical limits rather than adaptive protocol design. In addition, [32] investigates the throughput performance of NOMA-ALOHA under sparse active user scenarios in massive IoT networks. While it provides valuable insights into the asymptotic behavior and system scalability, it does not address learning-based adaptation or finite-frame constraints, which are critical in practical deployments.

To address reliability in MTC, several works have proposed transmission strategies such as prioritized burst repetition [33] and NOMA-based coded slotted ALOHA (CS-ALOHA) [34]. While these approaches reduce collisions and improve reliability, they rely on predefined repetition patterns or coding rules and lack adaptability to dynamic network conditions.

C. LEARNING-BASED OPTIMIZATION APPROACHES

Classical deep reinforcement learning (DRL) has been applied to improve random access and resource allocation in wireless networks [22]. DRL-based methods can learn

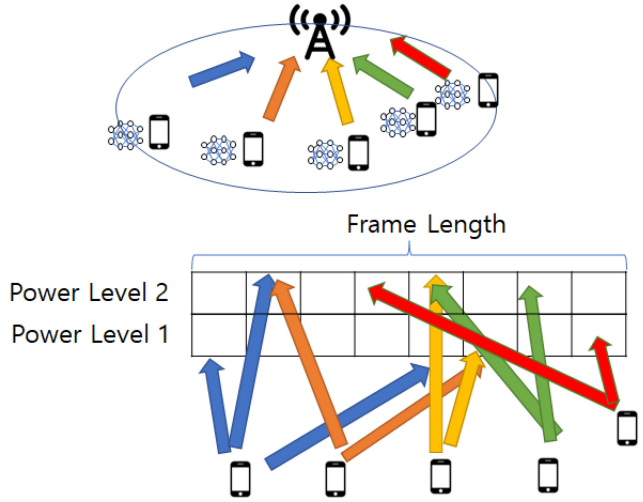

FIGURE 1. System model of IRSA-NOMA enhanced with MA-QDRL.

optimal repetition strategies in a decentralized setting but often suffer from convergence delays and scalability issues, particularly under dynamic and noisy channel conditions.

Recent work such as [35] applies DRL to NOMA-ALOHA under fading, learning the optimal transmit probabilities. Reference [36] further proposes distributed multi-agent reinforcement learning for heterogeneous NOMA-ALOHA systems, allowing decentralized policy adaptation. While these methods improve adaptivity, they rely on classical neural networks with high complexity and long training times.

These complementary extensions and their limitations reinforce the need for a learning-augmented, finite-frame IRSA-NOMA framework that supports power control, buffer dynamics, and real-time adaptability in dynamic wireless environments.

D. QUANTUM NEURAL NETWORKS FOR WIRELESS SYSTEMS

QNNs offer a promising approach to address scalability and expressivity challenges of classical learning methods. Prior works have explored QNNs for optimizing user pairing in downlink NOMA [37], resource allocation in MIMO-NOMA systems [38], and federated learning-based NOMA power control [39]. However, these approaches are largely confined to the downlink and overlook key aspects such as the parameter shift rule (PSR) [40], which enables efficient gradient computation for quantum models.

In contrast, our proposed framework introduces a MA-QDRL model for uplink IRSA-NOMA, considering finite frame length, fading, and buffer dynamics—critical yet underexplored challenges in prior QNN applications.

III. SYSTEM MODEL

This study focuses on the performance enhancement of IRSA-NOMA systems with finite frame lengths using the MA-QDRL approach. As illustrated in Fig. 1, the system comprises a base station (BS) and multiple decentralized uplink nodes, each equipped with a local QDRL agent. The system assumes a quasi-static Rayleigh fading channel, where the channel state remains constant within a frame and varies independently across frames. Each node estimates its local channel coefficient at the start of each frame using pilot signals from the BS. This enables frame-level decision-making with minimal overhead. For practical scalability to massive machine-type communication (mMTC), the MA-QDRL scheme operates in a fully decentralized fashion: each node observes only local state variables (e.g., buffer size, recent packet loss, and current channel gain) and optimizes its transmission strategy independently. This avoids centralized coordination and allows the scheme to scale efficiently with increasing node density. To estimate traffic load, each node indirectly infers contention levels through reinforcement signals such as packet loss frequency and reward feedback. Unlike traditional schemes that require explicit global load estimation, MA-QDRL leverages local observations to adapt transmission policies, making it suitable for highly dynamic and heterogeneous networks. Real-time channel coefficient updates are facilitated by periodic pilot exchanges at the start of each frame, which is practical in IoT systems with subframe-level scheduling (e.g., 0.25 ms in 5G NR with $\mu = 3$) [41]. While our current model assumes perfect CSI, future work will consider scenarios with delayed or noisy feedback.

In this study, we assume perfect channel state information (CSI) for both the BS and the uplink nodes, as well as perfect SIC at the BS. By assuming perfect CSI at all nodes, we eliminate uncertainties arising from channel estimation errors, enabling a clearer assessment of the QDRL-based optimization's impact on system performance. Similarly, the assumption of perfect SIC at the base station ensures that interference from multiple users is ideally canceled, thereby allowing us to focus on the inherent advantages of the proposed IRSA-NOMA framework. Although these idealized assumptions may not fully reflect real-world operating conditions, they provide a useful upper bound on system performance and serve as a baseline for future investigations that consider more practical scenarios. These assumptions are made to isolate and evaluate the effectiveness of the proposed MA-QDRL optimization under ideal conditions. While real-world systems inevitably suffer from imperfect CSI and non-ideal SIC, the current setup provides a controlled benchmark to measure the benefits of quantum-enhanced learning in decentralized uplink random access. Evaluating the system under imperfect assumptions remains an important direction for future work.

As described in Fig. 1, the frame structure consists of a finite frame length and two power levels. In accordance with URLLC requirements and the 3GPP 5G New Radio (NR) specifications, the transmission packet size is assumed to be 32 bytes (256 bits) [41], [42]. Moreover, in practical IoT deployments, devices often operate under strict energy and computational constraints, making smaller packet sizes, such

TABLE 2. Summary of notations used in this paper.

Symbol	Description
N	Number of uplink nodes (agents)
F	Frame size (number of timeslots per frame)
G	System load factor ($G = PN/F$)
P	Packet arrival rate
h	Instantaneous channel gain
h_{t1}, h_{t2}	Target channel gains for power levels 1 and 2
p_1, p_2	Power control coefficients at power levels 1 and 2
n_0	Normalized noise power
C_t	Target data rate (bits/symbol)
P_L	Packet loss (performance metric)
B	Packet buffer size per agent
s	State observed by an agent
a	Action selected (repetition factor)
$R(s, a)$	Reward for taking action a in state s
θ	Trainable parameter in the QNN
$\Delta\theta$	Gradient computed via parameter shift rule
L	Number of quantum layers in the QNN
N_{qubit}	Number of qubits in QNN
γ	Discount factor in reinforcement learning
τ	Soft update mixing coefficient
α	Learning rate
λ	Weight decay coefficient

as 32 bytes, preferable for reducing energy consumption and processing overhead. Although the proposed system does not fully adhere to stringent URLLC requirements, adopting a 32-byte packet size remains a reasonable choice, balancing efficiency and practical deployment in IoT networks.

Furthermore, each subcarrier in a timeslot is designed to contain 14 symbols, a structure consistent with the $\mu = 3$ configuration (i.e., a subcarrier spacing of 60 kHz) defined for 5G NR [41], [43]. To accommodate the transmission of a 32-byte packet within a single timeslot, eight subcarriers are allocated. This configuration ensures efficient communication while maintaining low latency and minimal resource utilization, which are critical factors in IoT applications. Consequently, the adoption of the 32-byte packet size in our model is justified not only by standardization but also by practical considerations, ensuring that the system remains both efficient and scalable in real-world scenarios.

Since a timeslot consists of 8 subcarriers and each subcarrier contains 14 symbols, the total number of symbols per timeslot is given by $N_{\text{symbols}}^{\text{timeslots}} = 8 \times 14 = 112$. Given the packet size of 32 bytes (i.e., 256 bits), the required data rate per symbol, denoted by C_t, is computed as $C_t = \frac{256\,\text{bits}}{112} \approx 2.29\,\text{bits}$ per symbol. Thus, under the given assumptions, each symbol must carry approximately 2.29 bits, which corresponds to the target data rate C_t, to successfully transmit the 32-byte packet within one timeslot. To account for fading channel effects, a quasi-static Rayleigh fading model is adopted. Since the channel state remains constant over the duration of a frame, the uplink nodes can decide whether to transmit based on the CSI available at the beginning of the frame. Specifically, if the instantaneous channel gain between the BS and an uplink node falls below the target channel gain, the transmission is suspended, and the packet is buffered at the corresponding uplink node. This mechanism ensures that only packets meeting the target channel conditions are transmitted, thereby mitigating interference.

To facilitate understanding, Table 2 provides a list of symbols and their corresponding definitions used in this study.

Then, the target channel gains h_{t1} and h_{t2} for power levels 1 and 2, respectively, are calculated based on the Shannon capacity formula:

$$C_t = \log_2\left(\frac{h_{t1}^2}{n_0} + 1\right), \quad C_t = \log_2\left(\frac{h_{t2}^2}{h_{t1}^2 + n_0} + 1\right), \tag{1}$$

where n_0 represents the normalized noise power. Solving for the target channel gains yields

$$h_{t1} = \sqrt{(2^{C_t} - 1)n_0}, \quad h_{t2} = \sqrt{(2^{C_t} - 1)(h_{t1}^2 + n_0)}. \tag{2}$$

Based on (2), the target channel gains h_{t1} and h_{t2} are determined. In the IRSA-NOMA system, to enable successful decoding via SIC, the received powers at the base station (BS) for packets transmitted at power levels 1 and 2 must be equalized. Consequently, if the actual channel gains h_1 and h_2 exceed the corresponding target values, they are adjusted as:

$$h_{t1} = p_1 h_1, \quad h_{t2} = p_2 h_2, \quad 0 < p_1 \leq 1, \quad 0 < p_2 \leq 1, \tag{3}$$

$$p_1 = \frac{h_{t1}}{h_1}, \quad p_2 = \frac{h_{t2}}{h_2}. \tag{4}$$

In the proposed IRSA-NOMA system, we adopt a simplified two-level NOMA configuration for uplink access. At the BS, SIC is applied by first decoding the stronger signal (power level 2), subtracting it from the composite signal, and then decoding the weaker one (power level 1). SIC is successful only if the signal-to-interference-plus-noise ratio (SINR) at each stage exceeds a predefined threshold. A packet is lost if either stage fails due to insufficient SINR or residual interference from collisions. By applying the scaling factors p_1 and p_2 at the uplink nodes, all signals within each power level are aligned to a common received power, thereby simplifying SIC at the BS. To enable tractable analysis and implementation, we assume uplink users perform power control to align the received power of their transmissions with the target level corresponding to the selected power level. That is, each user adjusts its transmit power so that its received power at the BS equals a predefined value per power level. This approach assumes perfect CSI at the start of each frame, enabling each node to compute the appropriate transmit scaling. While this equalization assumption may appear somewhat artificial, it facilitates receiver-side decoding, mitigates the near-far problem in dense networks, and contributes to energy efficiency by allowing users with favorable channel conditions to transmit at lower power levels.

To accommodate delayed transmissions, we assume that each node is equipped with a buffer capable of storing up to four untransmitted packets. If the buffer reaches its capacity, any newly arriving packet will cause the earliest stored packet to be automatically dropped, resulting in packet loss. Although the proposed system does not strictly adhere to the stringent constraints of Ultra-Reliable Low-Latency Communication (URLLC), it is designed to reflect key characteristics of low-latency systems. Given that a subframe duration is 0.25 ms for μ=3 (60 kHz subcarrier spacing), this buffer size allows packets to be retained for up to 1 ms before reaching the storage limit. This design choice ensures a balance between practicality and low-latency performance, preventing excessive buffering delays that could degrade real-time communication by increasing packet loss and transmission inefficiency.

In summary, the proposed IRSA-NOMA system operates in a time-slotted manner with finite frame length, leveraging NOMA with two power levels and an adaptive transmission strategy. Uplink nodes communicate through a quasi-static Rayleigh fading channel, where power control ensures efficient SIC at the base station. To support real-time IoT applications, packets are constrained to 32 bytes, aligning with 5G NR specifications, while the buffer size is optimized to limit excessive queuing delays. The proposed framework assumes perfect CSI and SIC to establish an upper bound on system performance, providing a foundation for further investigations under practical deployment constraints.

IV. PROPOSED METHOD

This study proposes the MA-QDRL for the IRSA-NOMA system. Based on the MA-QDRL, the IRSA-NOMA repetition will be decided. The primary goal of this study is to minimize packet loss P_L among uplink nodes in the IRSA-NOMA system. However, due to the decentralized nature of the system, each uplink node operates independently as an autonomous agent without sharing information with other nodes. This makes traditional centralized optimization approaches impractical.

To address this challenge, we employ the MA-QDRL framework, where each node learns its own transmission policy based solely on its local observations. The proposed framework is designed to reduce packet loss by maximizing the expected reward. The reward function incentivizes successful transmissions while penalizing packet loss, thereby guiding the learning process toward minimizing communication failures. The reward function is designed to incentivize successful transmissions while penalizing packet loss. The resulting optimization problem is formulated as follows:

$$\textbf{Maximize} \quad J(\mathbf{a}) = \sum_{n=1}^{N} \mathbb{E}[R_n(s_n, a_n)], \tag{5}$$

where $J(\mathbf{a})$ denotes the objective function, which represents the expected cumulative reward across all N nodes. The vector $\mathbf{a} = (a_1, a_2, \ldots, a_N)$ consists of the actions taken by each node, where a_n is the action selected by node n. The goal is to find the optimal action set $\mathbf{a}^*$ that maximizes the total expected reward.

The reward function R_n is designed as follows:

$$R_n(s_n, a_n) = \begin{cases} R^+, & \text{if the transmission is successful (i.e., no packet loss)}, \\ R^-, & \text{if the transmission fails (i.e., packet loss occurs)}, \\ 0, & \text{if no transmission occurs (e.g., buffering state)}. \end{cases} \tag{6}$$

where R^+ and R^- denote the reward values assigned for successful and failed transmissions, respectively.

Thus, by maximizing the cumulative reward, the system inherently learns to minimize packet loss while ensuring efficient transmissions.

STATE DEFINITION

For each node n, the state s_n is represented as:

$$s_n = (P_L^{\text{local}}(n), h_n, b_n), \tag{7}$$

where:

- $P_L^{\text{local}}(n)$ is the number of packet losses observed over the most recent 10 transmissions at node n,
- h_n denotes the current channel gain at node n,
- b_n represents the number of packets stored in the buffer at node n.

CONSTRAINTS

The optimization is subject to the following constraints:

- *Packet loss measurement over recent transmissions:*

$$P_L^{\text{local}}(n) = \sum_{j=t-9}^{t} \delta_{(\text{packet loss occurred at node } n,\ \text{transmission } j)}. \tag{8}$$

  where $\delta_{(\cdot)}$ is an indicator function that takes the value of 1 if a packet loss occurred at node n during transmission j, and 0 otherwise. The summation provides a historical record of packet losses over the most recent 10 transmissions, serving as feedback for reinforcement learning.

- *Action selection based on Q-learning [44]:*

$$a_n = \arg\max_a Q(s_n, a). \tag{9}$$

  Each node selects its transmission repetition factor a_n independently based on the learned Q-value function, aiming to maximize the expected reward given its current state s_n.

- *Action space restriction:*

$$a_n \in \{1, \ldots, 8\}. \tag{10}$$

  The selected transmission repetition factor must lie within a predefined discrete set of values, ensuring computational feasibility and adherence to practical system constraints.

These constraints define the learning process and decision-making mechanism for each node. The first constraint

TABLE 3. Quantum gates explanations.

Operation	Permutation Matrix
Hadamard gate $\mathbf{H}$	$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$
Pauli Z gate $\mathbf{CZ}$	$\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & -1\end{bmatrix}$
Rotation X gate $\mathbf{RX}$	$\begin{bmatrix}\cos(\theta/2) & -i\sin(\theta/2)\\ -i\sin(\theta/2) & \cos(\theta/2)\end{bmatrix}$
Rotation Y gate $\mathbf{RY}$	$\begin{bmatrix}\cos(\theta/2) & -\sin(\theta/2)\\ \sin(\theta/2) & \cos(\theta/2)\end{bmatrix}$

ensures that each node maintains a record of packet losses, enabling informed decision-making based on recent transmission history. The second constraint governs the action selection process, where each node optimizes its transmission strategy using Q-learning without centralized coordination. Finally, the action space restriction enforces practical limits on the transmission repetition factor, maintaining efficiency and feasibility within the reinforcement learning framework.

A. QUANTUM NEURAL NETWORK

This study proposes the MA-QDRL framework based on QNN to enhance the performance of the IRSA-NOMA system. The QNN is constructed using variational quantum circuits (VQCs), also known as parameterized quantum circuits, which allow for adaptive parameter optimization during training.

To better understand how the QNN functions within our framework, we first provide a brief overview of the fundamental principles of quantum computing.

Qubits exist in a state that can be represented using any selected orthogonal basis, typically corresponding to the states $|0\rangle$ and $|1\rangle$. The state of a qubit, denoted as $|\psi\rangle$, can be expressed as a state vector [45]. The amplitudes of each orthogonal state define the components of the state vector $|\psi\rangle$. In the computational basis $\{|0\rangle, |1\rangle\}$, the quantum state of a single qubit is given by:

$$|\psi\rangle = \begin{bmatrix}a\\ b\end{bmatrix} = \alpha|0\rangle + \beta|1\rangle, \tag{11}$$

where the probability amplitudes satisfy the normalization condition:

$$\alpha^2 + \beta^2 = 1. \tag{12}$$

Here, the first element of the state vector corresponds to the amplitude of the $|0\rangle$ state, while the second element represents the amplitude of the $|1\rangle$ state.

If the amplitudes α and β are equal, the quantum state $|\psi\rangle$ can be written as:

$$|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle. \tag{13}$$

For a system of N qubits, the overall quantum state is represented as the tensor product of individual qubit states:

$$|\Psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_N\rangle. \tag{14}$$

A quantum measurement is performed to extract classical values from quantum states. Unlike classical bits, which hold deterministic values of either 0 or 1, qubits exist in a superposition of states until they are measured. The measurement process collapses the quantum state into one of the computational basis states, $|0\rangle$ or $|1\rangle$, with probabilities determined by the squared magnitudes of their respective probability amplitudes.

Mathematically, quantum measurement is associated with a *quantum observable*, denoted as Υ, which corresponds to a Hermitian operator. The quantum observable can be expressed as:

$$\Upsilon = \sum_{\upsilon} \delta_\upsilon \alpha_\upsilon, \tag{15}$$

where δ_υ represents the eigenvalues of Υ and α_υ is a projector onto the corresponding eigenspace.

When a quantum system is in a specific state $|\phi\rangle$, the probability of obtaining a measurement outcome δ_υ is given by:

$$p(\delta_\upsilon) = \langle\phi|\alpha_\upsilon|\phi\rangle. \tag{16}$$

The expectation value of the quantum observable Υ in state $|\phi\rangle$ is then given by:

$$\langle\Upsilon\rangle_\phi = \langle\phi|\Upsilon|\phi\rangle. \tag{17}$$

These measurement principles are fundamental in quantum computing, as they determine how quantum information is retrieved and used in computations. The probabilistic nature of quantum measurement introduces uncertainty, but it also enables quantum parallelism, which is leveraged in quantum algorithms and machine learning models such as QDRL.

The architecture of the proposed QNN is illustrated in Fig. 2. The QNN follows a layer-wise structure, where each layer processes three input values using quantum rotations and entanglement operations [46]. The network consists of a total of eight qubits, which evolve through four layers of quantum transformations. Each layer applies rotation gates to encode input features and trainable parameters, while entanglement is introduced through controlled operations. After all layers, the final quantum state is measured on all eight qubits, producing an 8-qubit output representation.

Since the VQC follows a layer-wise design, each layer comprises three RX gates to encode input features and eight RY gates to represent trainable weight parameters.

Four types of quantum gates are utilized to construct the QNN, as detailed in Table 3:

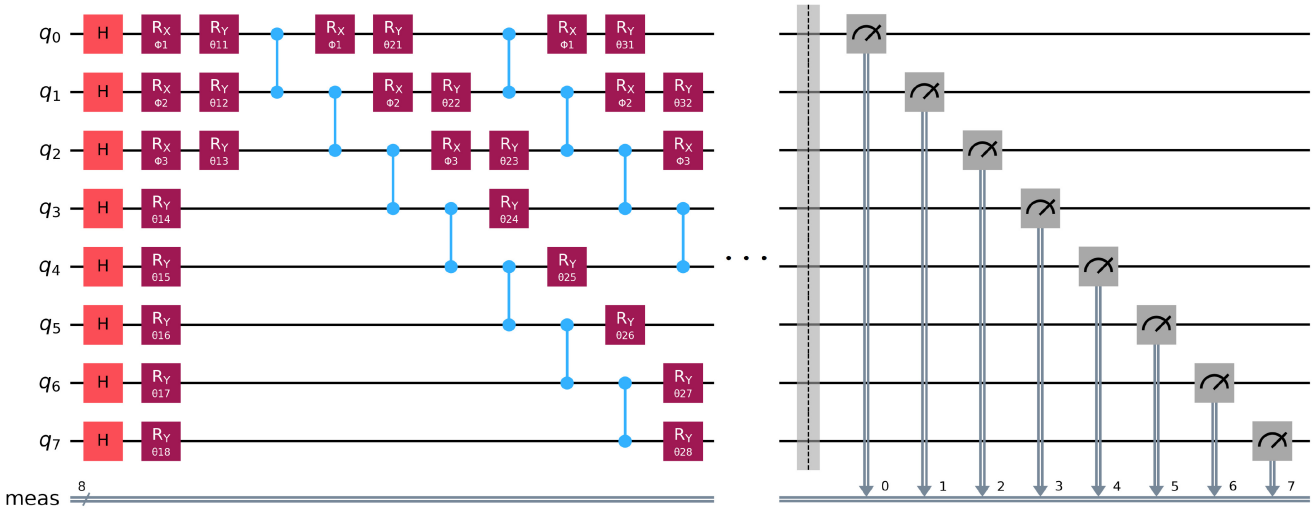

FIGURE 2. Proposed quantum circuit (layer-wise structure used in the QNN of MA-QDRL).

- The Hadamard gate (**H**) is applied at the beginning to transform qubits into an equal superposition state, enabling quantum parallelism.
- The Controlled-Z gate (**CZ**) introduces entanglement between qubits, ensuring quantum correlations are leveraged during computation.
- The Rotation-X gate (**RX**) encodes the classical input vector $\mathbf{x}$ by applying rotations around the X-axis of the Bloch sphere.
- The Rotation-Y gate (**RY**) parametrizes trainable weights by rotating qubits around the Y-axis, allowing for adjustable transformations during optimization.

In quantum computing, encoding operations transform classical values into quantum states, allowing quantum neural networks (QNNs) to process classical input data in a quantum framework. Given an input vector $\mathbf{x}$ with N elements, the encoding operation maps $\mathbf{x} = \{x_n\}_{n=1}^{N}$ (where x_n is the nth input element) to a corresponding quantum state representation.

To ensure proper encoding, the classical values in $\mathbf{x}$ must be scaled into an appropriate range for quantum operations. The quantum rotation gates used in the encoding process operate within a phase range of $[-\pi, \pi]$. Therefore, each input element x_n is scaled as follows:

$$\phi_n = -\pi + \frac{2\pi (x_n - x_{\min})}{x_{\max} - x_{\min}}. \tag{18}$$

The input encoding is performed using *RX* gates, which apply rotations around the X-axis of the Bloch sphere. Since the QNN processes three input values while utilizing a total of eight qubits, only the first N_{inputs} qubits undergo rotation transformations, while the remaining $N_{\text{outputs}} - N_{\text{inputs}}$ qubits remain unchanged. To ensure this, identity operations are applied to the unused qubits. Therefore, the encoding operation is expressed as:

$$S_x = \bigotimes_{n=1}^{N_{\text{inputs}}} \mathbf{RX}(\phi_n) \otimes \bigotimes_{n=N_{\text{inputs}+1}}^{N_{\text{outputs}}} \mathbf{I}, \tag{19}$$

where $\mathbf{RX}(\phi_n)$ rotates the nth qubit by an angle ϕ_n based on the scaled input value, while $\mathbf{I}$ represents the identity operation applied to qubits that do not participate in the encoding process.

In addition to input encoding, the trainable parameters of the QNN, denoted as weight parameters θ in *RY* gates, are randomly initialized within the range $[-\pi, \pi]$. These parameters are iteratively adjusted during training to optimize the QNN's performance. The corresponding quantum operation for applying these parameters is given by:

$$S_y^{(j)} = \bigotimes_{m=1}^{N_{\text{outputs}}} \mathbf{RY}(\theta_m^{(j)}), \tag{20}$$

where $\theta_m^{(j)}$ represents the trainable parameter associated with the mth qubit in the jth layer. The parameters $\theta_m^{(j)}$ are trainable and are updated throughout the learning process to optimize the performance of the QNN.

The first layer is expressed as follows:

$$|\Psi_1\rangle = \prod_{i=1}^{N_{\text{outputs}}-1} \mathbf{CZ}(i, i+1) \cdot S_y^{(1)} \cdot S_x \bigotimes_{i=1}^{N_{\text{outputs}}} H \cdot |0\rangle^{\otimes N_{\text{outputs}}}. \tag{21}$$

As shown in Fig. 2, the input qubits are initialized in the state $|0\rangle^{\otimes N_{\text{outputs}}}$. A Hadamard gate (**H**) is applied to each qubit, creating a uniform superposition state to prepare them for further processing. **CZ** gates are applied between adjacent qubits to introduce entanglement, creating correlations between qubits. This enhances the representational power of the quantum state. The entanglement spans all N_{outputs}

qubits, with each qubit participating in at least one entangling operation. Following entanglement, S_x is applied to encode the input data, and $S_y^{(1)}$ is used to apply trainable weight parameters in the first layer, as defined in (19) and (20).

The quantum state of the first layer, $|\Psi_1\rangle$, results from applying the **H** gates, **RX** gates for classical data encoding, **RY** gates for parameterization, and **CZ** gates for entanglement. This quantum state serves as the input to the subsequent layers of the QNN, enabling quantum-based optimization in the QDRL framework.

The jth layer between the input and output layers is expressed as follows:

$$|\Psi_{\mathrm{j}}\rangle = \prod_{i=1}^{N_{\mathrm{outputs}}-1} \mathbf{CZ}(i, i+1) \cdot S_y^{(j)} \cdot S_x \cdot |\psi_{\mathrm{j\text{-}1}}\rangle. \quad (22)$$

As the proposed QNN model follows a layer-wise structure, the jth layer also encodes the scaled input values ϕ using $\mathbf{R_X}$ gates, as defined in (19). However, unlike the first layer (21), the key difference lies in the initialization process. Specifically, the Hadamard gates (**H**) and state preparation $|0\rangle$ are applied only in the first layer and are omitted in subsequent layers. Apart from this, the operations in the jth layer remain identical to those in the first layer.

The final layer is expressed as follows:

$$|\Psi_{\mathrm{final}}\rangle = S_y^{(\mathrm{final})} \cdot s_x \cdot |\psi_{\mathrm{final\text{-}1}}\rangle. \quad (23)$$

The quantum state of the output layer, $|\psi_{\mathrm{final}}\rangle$, is calculated as shown in (23). The main difference between the output layer and the jth hidden layer is the absence of **CZ** gates. Since the output layer does not require entanglement between qubits, **CZ** gates are omitted. All other operations, including input encoding and trainable weight application, remain the same as in the jth layer. These are represented using S_x and S_y, which encapsulate the transformations performed by the R_X and R_Y gates, respectively, as previously defined in (19) and (20).

The unified expression for the QNN model, encompassing (21)–(23), is as follows:

$$|\Psi_{\mathrm{final}}\rangle = \left[\prod_{j=1}^{N_{\mathrm{layers}}} \left(\mathbf{U_{CZ}}^{(j)} \cdot S_y^{(j)} \cdot S_x\right)\right] \cdot \bigotimes_{i=1}^{N_{\mathrm{outputs}}} H \cdot |0\rangle^{\otimes N_{\mathrm{outputs}}}, \quad (24)$$

where

$$\mathbf{U_{CZ}}^{(j)} = \begin{cases} \mathbf{I}, & \text{if } j = N_{\mathrm{layers}}, \\ \prod_{i=1}^{N_{\mathrm{outputs}}-1} \mathbf{CZ}(i, i+1), & \text{otherwise.} \end{cases} \quad (25)$$

Equation (24) describes the iterative application of the layers, starting from the initialized state $|0\rangle^{\otimes N_{\mathrm{outputs}}}$, applying H gates, and then progressing through each layer j.

B. QUANTUM DEEP REINFORCEMENT LEARNING

In this section, we design a learning algorithm to train the QDRL model. In IRSA-NOMA systems, packet loss is influenced by multiple interdependent factors, including channel conditions, buffer states, and contention among nodes, making the optimization problem highly complex and non-convex. Traditional algorithms struggle to efficiently adapt to these dynamic conditions. Unlike conventional optimization techniques, such as brute-force search or convex optimization, which require exhaustive computation or explicit mathematical modeling of the system, reinforcement learning is well-suited for this problem due to its ability to handle dynamic environments and learn optimal policies without prior knowledge of the system model [47]. While Q-learning enables decision-making based on learned policies, it lacks scalability in large-scale and non-stationary environments. In contrast, CDRL, which utilizes traditional neural networks, enables each node to continuously learn and adapt its transmission strategy based on real-time observations, making it well-suited for dynamic environments. However, despite its advantages, CDRL-based optimization still faces challenges such as slow convergence and high computational complexity, particularly in large-scale networks [48].

The proposed MA-QDRL framework is inherently scalable to massive IoT scenarios. Since each uplink node operates as an independent agent using only local observations, no global coordination is required. Moreover, the use of compact QNN models reduces computational overhead, allowing efficient training and inference even with a large number of nodes. This decentralized architecture is especially well-suited for massive machine-type communication environments, where centralized scheduling is often infeasible.

To further enhance learning efficiency and address computational challenges, we propose the MA-QDRL framework based on QNNs. By leveraging quantum computing's ability to process high-dimensional data efficiently, the QDRL framework accelerates training and improves the overall performance of IRSA-NOMA optimization.

DOUBLE DEEP Q-LEARNING (DDQN)

The double deep Q-network (DDQN) structure [49] is applied to the QDRL model, as described in Fig. 3. Unlike the standard deep Q-network (DQN), which tends to overestimate Q-values, DDQN mitigates this bias by using two separate networks: a policy network Q_p (online Q-network) for action selection and a target network Q_t for Q-value evaluation. Q_p selects the optimal action, while Q_t estimates its value, leading to more stable learning. Experience replay is used to train the model by randomly sampling past transitions, ensuring better generalization and reducing correlation between updates. This structure improves the stability and accuracy of Q-value estimation in reinforcement learning.

In the proposed QDRL framework, a QNN is adopted instead of a traditional neural network. Q_p predicts the Q-values online for selecting the action a, expressed as:

$$a = \arg\max_a Q_p(s, a; \theta), \quad (26)$$

where s represents the state, and θ denotes the parameters of Q_p. The state s is defined as a vector consisting of:

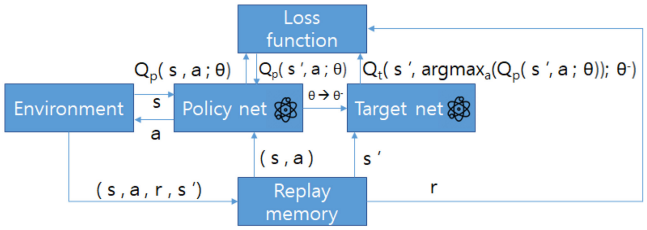

FIGURE 3. Parameter Update with quantum neural networks.

- Channel gain h,
- Packet loss count p_L, representing the number of packet losses over the most recent 10 transmissions,
- Buffer state b, indicating the number of packets stored in the buffer.

The action a represents the number of repeated transmissions, ranging from 1 to 8. For example, if the output index of the QNN is 2, it implies that two repeated transmissions will be performed.

The target network Q_t computes the target Q-value during the Bellman update [49]:

$$y = r + \gamma Q_t(s', \arg\max_a Q_p(s', a; \theta); \theta'), \tag{27}$$

where r is the reward, s' is the next state, and θ' denotes the parameters of Q_t.

To train Q_p, the loss function $L(\theta)$ for optimizing θ is defined as:

$$L(\theta) = \mathbb{E}_{(s,a,r,s')\sim D}\left[(y - Q_p(s, a; \theta))^2\right], \tag{28}$$

where D is the experience replay buffer storing past transitions (s, a, r, s'), and $\mathbb{E}$ denotes the expectation taken over the distribution of sampled experiences.

As shown in Fig. 3, experience tuples E (s, a, r, s') are collected from the environment and stored in replay memory. These experience tuples E are randomly sampled to train Q_p. Q_t is periodically updated by replacing its parameters θ' with those of Q_p.

In classical deep learning, parameters θ are typically optimized using gradient-based methods. However, VQCs commonly adopt the parameter shift rule (PSR) for optimization [40]. Using PSR, the gradients $\Delta\theta$ are computed as:

$$\Delta\theta_m^{(j)} = \frac{L(\theta_m^{(j)} + \pi/2) - L(\theta_m^{(j)} - \pi/2)}{2}. \tag{29}$$

The computed gradient $\Delta\theta$ is then used to update the parameters:

$$\theta_m^{(j)} = \theta_m^{(j)} - \alpha\Delta\theta_m^{(j)}. \tag{30}$$

Based on (29) and (30), the mth parameter θ_m in the jth layer is updated accordingly.

To improve convergence stability and mitigate overfitting, the AdamW optimizer [50] is employed for parameter updates. AdamW introduces weight decay regularization,

Algorithm 1: Learning Scheme for the Proposed QDRL Model

Input: Experience tuples E of N transitions randomly sampled from the replay memory D

$\Delta\theta$ = **zeros**(Size(θ), N)
for j=*1:N* **do**
  $(s, a, r, s') = E_j$
  $y = r + \gamma\ Q_t(s', \arg\max_a Q_p(s', a; \theta); \theta')$
  for i=*1:Size(θ)* **do**
    $\theta_i^+ = \theta_i + \pi/2$
    $\theta_i^- = \theta_i$ - $\pi/2$
    $l^+ = (y$ - $Q_p(s, a; \theta^+))^2$
    $l^- = (y$ - $Q_p(s, a; \theta^-))^2$
    $\Delta\theta_i[j] = \Delta\theta_i[j] + (l^+$ - $l^-)/2$
for i=*1:Size(θ)* **do**
  $\theta_i = \theta_i$ - α **mean**($\Delta\theta_i$[:])
$\theta' = \tau\theta + (1\text{-}\tau)\theta'$

TABLE 4. Simulation environment.

CPU	Intel(R) Core(TM) i9-10940X CPU @ 3.30GHz
RAM	128 GB
GPU	NVIDIA GeForce RTX 3090
OS	Ubuntu 20.04.6 LTS
Python Version	3.8.18
Qiskit Version	1.2.0

which enhances training efficiency by preventing parameter growth in VQCs. The AdamW update rule is given by:

$$\theta_m^{(j)} = \theta_m^{(j)} - \alpha \cdot \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} - \lambda\theta_m^{(j)}, \tag{31}$$

where $\hat{m}_t$ and $\hat{v}_t$ denote bias-corrected first and second moment estimates, respectively, λ represents the weight decay factor, and ϵ is a small constant for numerical stability.

After the parameter update, the soft update is applied to synchronize the target network [51]:

$$\theta' = \tau\theta + (1 - \tau)\theta'. \tag{32}$$

The overall procedure of QDRL is outlined in Algorithm 1. Experience tuples E are randomly sampled from the replay buffer D with a batch size B. For each experience E_j, state s, action a, reward r, and next state s' are extracted and used to train Q_p. The target Q-value y is computed using (27), and the policy network Q_p is updated using (29). The gradient $\Delta\theta_i$ for each parameter θ_i is calculated using the batch, followed by updates via (30) and (32), as described in Algorithm 1.

V. SIMULATION RESULTS

In this section, the proposed MA-QDRL solution applied to IRSA-NOMA with finite frame length is evaluated.

The simulation environment used in this study is summarized in Table 4. The experiments were conducted on a computing system with an Intel Core i9-10940X CPU,

TABLE 5. Simulation parameters.

Frame Size	8, 14, 20
Number of Subcarriers	8
Number of Symbols	14
Packet Size (Bytes)	32
Transmit SNR (dB)	30
Channel Model	Quasi-Static Rayleigh Fading (σ=1)
NOMA Power Levels	2
Packet Buffer Size for Each Agent	4
Iteration (times)	1000

TABLE 6. QDRL agent parameter.

Structure (Layer-wise structure to encode 3 inputs)	$8 \times 8 \times 8 \times 8$
Replay Memory Size	1000
Batch Size	10
Learning Rate	0.1
Optimizer	AdamW
Gradient Estimation Method	Parameter Shift Rules
Target Update Method	Soft Update (τ=0.005)
Discount Factor (Γ)	0.99

128 GB RAM, and an NVIDIA RTX 3090 GPU. The system operated on Ubuntu 20.04.6 LTS, using Python 3.8.18 and Qiskit 1.2.0 for quantum simulations.

The simulation parameters are described in Table 5. The frame size is set to 8, 14, and 20. In 5G NR, a slot consists of 14 OFDM symbols under the μ=3 configuration (subcarrier spacing of 60 kHz) [41]. Since a slot is the minimum scheduling unit, it is natural to set the frame size as a multiple of 8. However, to evaluate system performance under diverse conditions, non-multiples of 8, such as 14 and 20, are also considered. To transmit a packet of 32 bytes within one time slot, the number of subcarriers and symbols is set to 8 and 14, respectively. Therefore, each symbol should have a capacity of at least 2.29 bits/symbol. The transmit SNR is set to 30 dB, and the channel model follows a quasi-static Rayleigh fading channel, where the channel gain varies but remains constant within a frame. The NOMA system operates with two power levels, and each agent has a packet buffer size of 4. If the buffer exceeds this limit, the oldest packet is dropped.

The QDRL agent parameter is described in Table 6. The QNN structure is described in (24), which is composed of 4 layers, each consisting of 8 qubits. This structure follows a layer-wise design, where 3 input features are encoded into the quantum circuit at each layer and processed to produce 8 output qubits. The layer-wise encoding and transformation enable the gradual evolution of quantum states across all 4 layers.

The configuration of the training parameters for the proposed MA-QDRL model is as follows:

- *Replay Memory Size:* The replay memory is set to a size of 1000, enabling the storage of a sufficient number of experience tuples to enhance learning stability and mitigate overfitting.
- *Batch Size:* A batch size of 10 is used during training, allowing efficient gradient updates based on small subsets of the replay memory.
- *Learning Rate:* The learning rate is set to 0.1, controlling the step size for weight updates during the optimization process.
- *Optimizer:* The AdamW optimizer is employed to enhance convergence by introducing weight decay to reduce overfitting.
- *Gradient Estimation Method:* The parameter shift rule is utilized for gradient estimation, a method tailored for quantum neural networks, ensuring accurate gradient calculations.
- *Target Update Method:* A soft update strategy is adopted for the target network with $\tau = 0.005$, ensuring smooth updates and reducing instability during training.
- *Discount Factor* (Γ)*:* A discount factor of 0.99 is applied to balance the importance of immediate and future rewards, promoting long-term reward optimization.
- *Decentralized Learning:* Each agent operates independently without sharing information with other agents. The learning process is entirely decentralized, meaning that each agent optimizes its transmission strategy based solely on its local observations and experience.

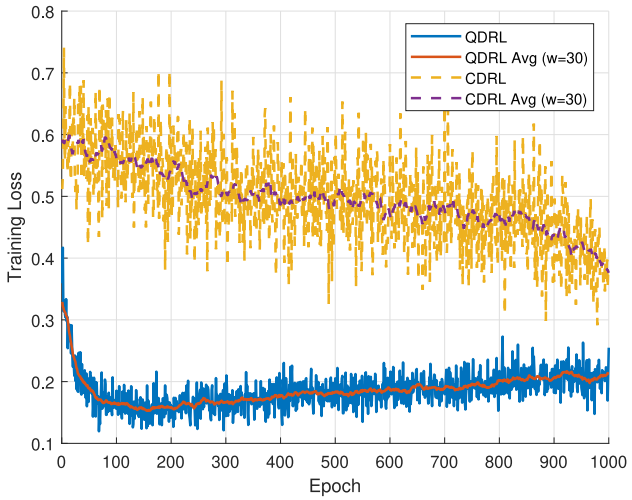

FIGURE 4. Training loss (comparison between the proposed QDRL and CDRL using MSE loss).

This configuration is designed to achieve efficient and stable training in the MA-QDRL model.

Fig. 4 illustrates the training loss of MA-QDRL compared to a MA-CDRL model, which employs a standard neural network with an architecture of $3 \times 8 \times 8 \times 8$. The proposed MA-QDRL model demonstrates faster and more stable convergence than MA-CDRL, as it leverages a QNN that utilizes PSR instead of backpropagation. By eliminating iterative cost function minimization, QDRL enables faster training convergence, making it well-suited for real-time scheduling tasks [52].

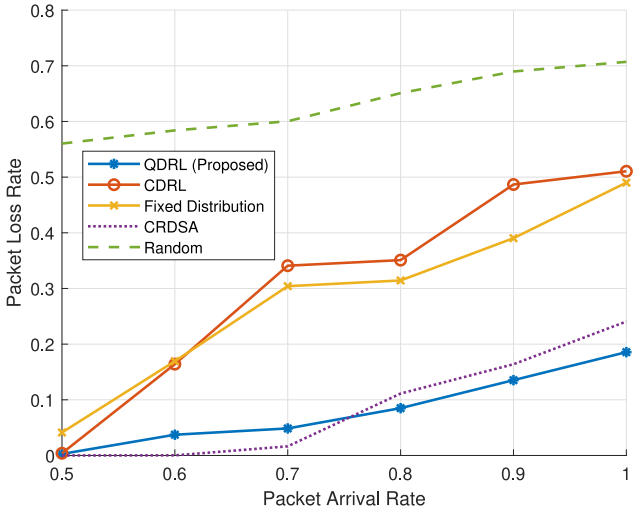

FIGURE 5. Varying packet arrival rate (frame size = 8, number of nodes = 8).

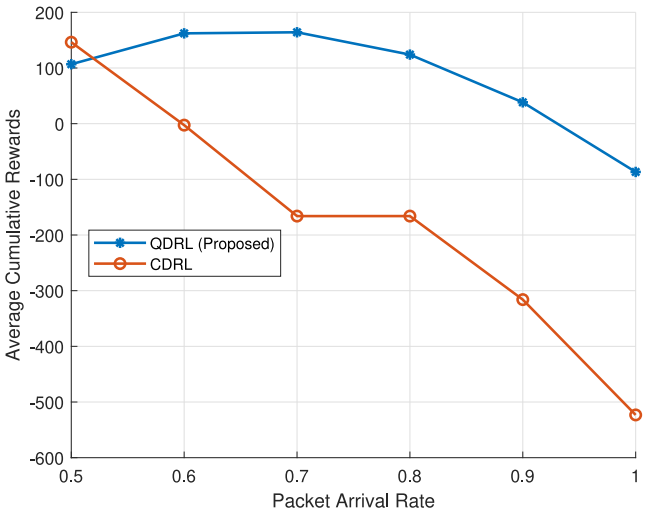

FIGURE 6. Average cumulative reward per user vs. packet arrival rate (Frame Size = 8, Number of Nodes = 8).

Fig. 5 presents the packet loss rate comparison as the packet arrival rate P varies from 0.5 to 1.0 in an environment where the frame size F and the number of nodes N are both fixed at 8. The proposed QDRL model is evaluated against CDRL, a fixed transmission distribution ($0.5112x^2 + 0.266x^3 + 0.2228x^8$ [20]), CRDSA and a random transmission scheme. When the P is 0.5, MA-QDRL, MA-CDRL, the fixed distribution, and CRDSA all exhibit low packet loss rates. However, as the P increases beyond 0.6, the differences between the methods become more pronounced. Notably, CRDSA outperforms MA-QDRL up to P of 0.7, showing its effectiveness in low-to-moderate traffic conditions. However, starting from P=0.8 MA-QDRL consistently outperforms CRDSA and all other methods due to its faster convergence and higher scalability under heavier traffic conditions. In contrast, MA-CDRL suffers from higher packet loss rates, as shown in Fig. 4, due to its slow convergence rate.

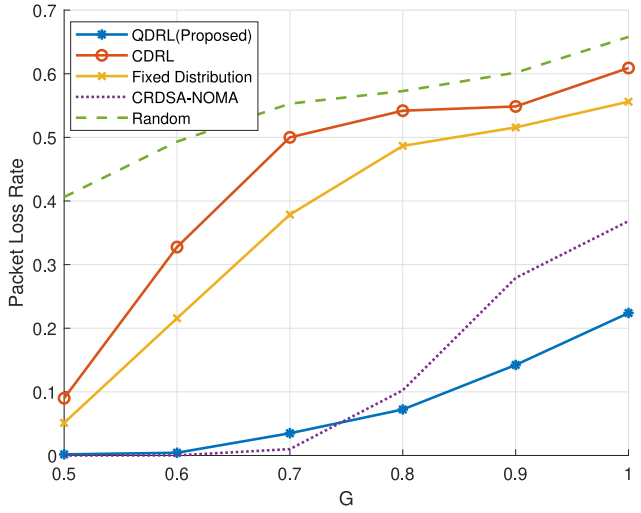

FIGURE 7. Varying G (frame size = 8, packet arrival rate = 0.5).

Fig. 6 shows the average cumulative reward per user after 1000 training epochs. At $P = 0.5$, the MA-CDRL model achieves a slightly higher reward. However, as the packet arrival rate increases ($P \geq 0.6$), its performance degrades significantly. In contrast, MA-QDRL maintains relatively stable rewards under high traffic conditions. Even at $P = 1.0$, where MA-QDRL begins to lose reward due to system congestion, it still outperforms MA-CDRL. These results demonstrate the superior adaptability of MA-QDRL in handling heavy traffic loads in IRSA-NOMA systems.

Table 7 presents the learned degree distributions under varying packet arrival rates P, with fixed frame size $F = 8$ and number of nodes $N = 8$, as shown in Fig. 5. As the network traffic load increases with higher P, MA-QDRL dynamically adapts its transmission strategy by learning appropriate degree distributions tailored to each traffic condition. While CRDSA maintains competitive performance under low-to-moderate traffic conditions (e.g., $P \leq 0.7$), the relative performance advantage of MA-QDRL becomes more pronounced as the traffic load increases, surpassing CRDSA under high arrival rates (e.g., $P \geq 0.8$). Notably, the learned distributions tend to assign high probability to degree-1 transmissions across all conditions, indicating that excessive repetition is suboptimal in short-frame settings. This suggests that, when the frame size is limited, minimizing transmission collisions takes precedence over redundancy, and the learning-based approach effectively captures this trade-off. These results highlight the adaptability and scalability of the proposed learning-based framework in handling diverse network congestion scenarios.

Fig. 7 presents the packet loss rate comparison as the system load factor G varies from 0.5 to 1.0. P and F are fixed at 0.5 and 8, respectively, while the system load factor G is adjusted by varying the number of nodes. G is defined as:

$$G = P\frac{N}{F}. \tag{33}$$

TABLE 7. Learned degree distributions for $F = 8$, $N = 8$, under varying packet arrival rate P.

P	Learned Degree Distribution
0.6	$0.8391x + 0.0766x^2 + 0.0397x^3 + 0.0322x^4 + 0.0101x^5 + 0.0015x^6 + 0.0006x^7 + 0.0002x^8$
0.7	$0.8916x + 0.0515x^2 + 0.0352x^3 + 0.0133x^4 + 0.0066x^5 + 0.0007x^6 + 0.0009x^7 + 0.0002x^8$
0.8	$0.8546x + 0.0676x^2 + 0.0401x^3 + 0.0236x^4 + 0.0109x^5 + 0.0024x^6 + 0.0004x^7 + 0.0004x^8$
0.9	$0.8815x + 0.0464x^2 + 0.0391x^3 + 0.0161x^4 + 0.0107x^5 + 0.0036x^6 + 0.0006x^7 + 0.0020x^8$
1.0	$0.8789x + 0.0723x^2 + 0.0227x^3 + 0.0202x^4 + 0.0039x^5 + 0.0013x^6 + 0.0007x^7$

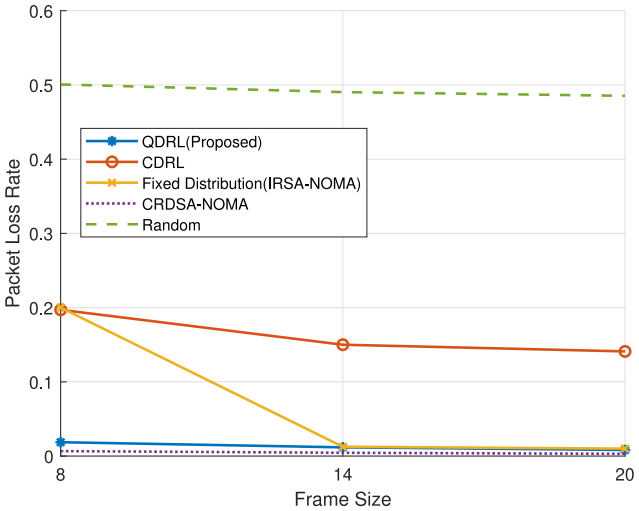

FIGURE 8. Varying frame size (packet arrival rate = 0.5, G = 0.6).

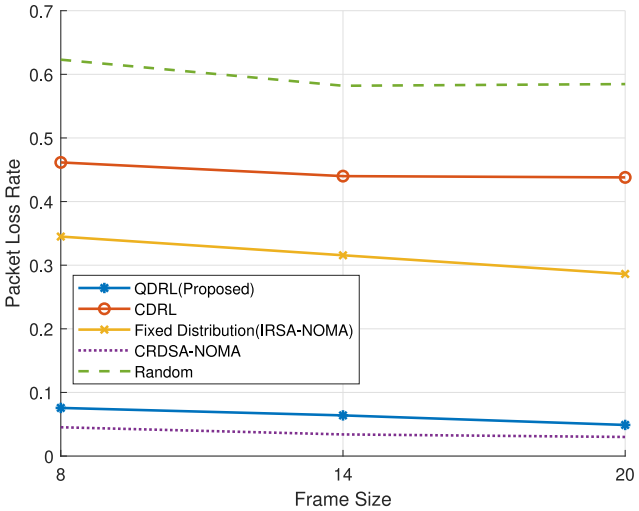

FIGURE 9. Varying frame size (packet arrival rate = 0.5, G = 0.7).

As G increases, all algorithms experience higher packet loss due to increased network congestion. CRDSA-NOMA initially shows competitive performance at lower values of G, however its packet loss rate rapidly increases, becoming higher than MA-QDRL as G surpasses approximately 0.8. However, as the traffic load increases beyond $G \approx 0.8$, the proposed MA-QDRL model consistently achieves the lowest packet loss, demonstrating superior performance over the other approaches under high network congestion conditions. At $G = 1.0$, MA-QDRL achieves a packet loss rate of 0.2239, compared to 0.3684 for CRDSA-NOMA and 0.6091 for MA-CDRL. This corresponds to a reduction of approximately 39.2% and 63.2%, respectively. The fixed transmission distribution exhibits the third-best performance, followed by MA-CDRL. The MA-CDRL approach struggles with higher packet loss due to its slow convergence rate, requiring more training epochs to reach optimal performance. As network size increases, the scalability limitations of CDRL become more evident, as its training inefficiency makes it difficult to adapt to large-scale, high-complexity environments. This suggests that CDRL may not be the most suitable multi-agent solution for IRSA-NOMA systems, particularly under high network load conditions.

Fig. 8 shows the packet loss rate performances among the algorithms when the frame size varies 8, 14 and 20. P and G are fixed at 0.5 and 0.6. CRDSA-NOMA consistently achieves the lowest packet loss among the compared methods at this relatively low load condition G=0.6. MA-CDRL exhibits the worst performance, aside from random transmission, as higher values of G correspond to an increased number of deployed nodes. The presence of more nodes significantly increases the search complexity, making it more challenging for CDRL to effectively optimize transmission strategies. Therefore, CDRL's learning convergence cannot reach the appropriate points. In contrast, although MA-QDRL demonstrates a relatively higher packet loss rate compared to CRDSA-NOMA at this specific load ($G = 0.6$), its performance is not significantly worse. Moreover, as previously shown in Fig. 7, MA-QDRL outperforms CRDSA-NOMA when the network load increases beyond $G = 0.8$, highlighting MA-QDRL's superior capability under higher network congestion conditions. Additionally, as the frame size increases, the fixed distribution exhibits a lower packet loss rate. This is because the fixed distribution was originally derived based on the assumption of an infinite frame length. Consequently, a larger frame size allows it to better approximate its theoretical performance, leading to improved packet loss reduction.

Fig. 9 shows the packet loss rate performances among the algorithms when the frame size varies at 8, 14, and 20. P and G are fixed at 0.5 and 0.7, respectively. At $G = 0.7$, MA-CDRL's performance deteriorates due to the increased complexity and convergence challenges caused by a higher number of deployed nodes. Notably, MA-CDRL exhibits the poorest performance, except for random transmission, highlighting its limitations under moderate-to-high network

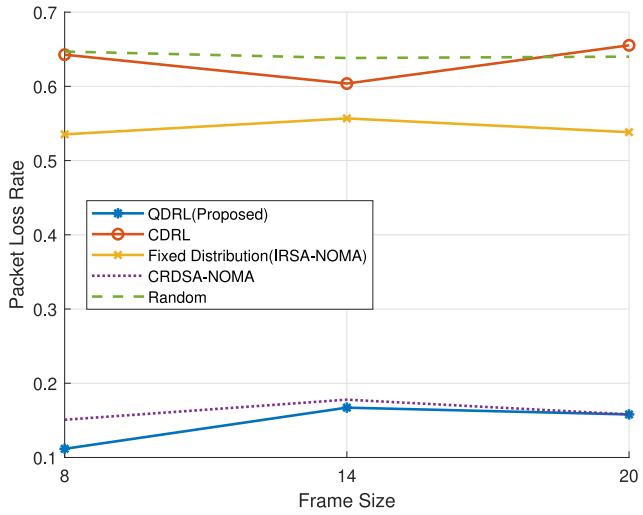

FIGURE 10. Varying frame size (packet arrival rate = 0.5, G = 0.8).

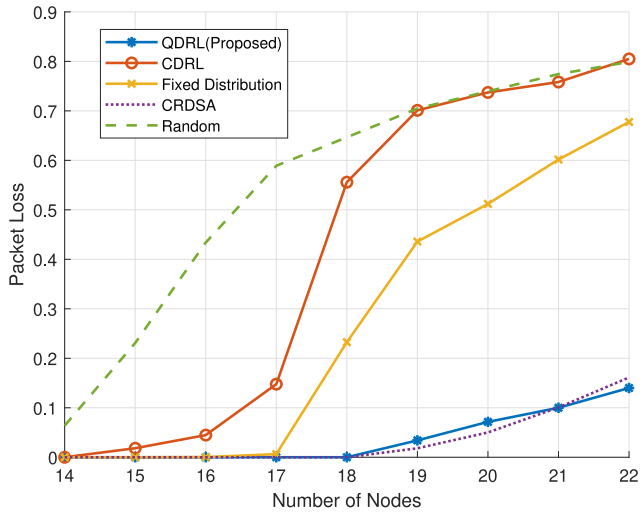

FIGURE 11. Varying number of nodes (frame size = 14, packet arrival rate = 0.5).

load conditions. At $G = 0.7$, CRDSA-NOMA achieves the lowest packet loss rate among all methods, clearly demonstrating its effectiveness under moderate network load conditions. Although the proposed MA-QDRL model has a slightly higher packet loss rate compared to CRDSA-NOMA under these conditions, it consistently outperforms the fixed distribution, MA-CDRL, and random transmission methods.

Fig.10 presents the packet loss rate comparison among the algorithms when the frame size varies at 8, 14, and 20, with P and G fixed at 0.5 and 0.8, respectively. This experiment is designed to validate that the proposed MA-QDRL scheme consistently outperforms the other approaches under high network load conditions. As previously demonstrated in Fig.7, MA-QDRL again achieves the best performance in Fig. 10, confirming its robustness and scalability in congested network scenarios. To further investigate the performance degradation trend of MA-CDRL and the effectiveness of MA-QDRL under increasing network load, we incrementally increased the number of nodes while fixing $F = 14$ and $P = 0.5$, as shown in Fig. 11. This configuration allows us to explore a continuous range of system load factors $G = P\frac{N}{F}$ from moderate to high. For instance, when $N = 19$, the system load factor becomes $G \approx 0.68$, and when $N = 22$, $G \approx 0.79$. As the number of nodes increases, all methods experience growing packet loss. CRDSA-NOMA shows strong performance across most conditions and achieves the lowest packet loss rate until $N = 21$. However, at $N = 22$, the proposed MA-QDRL scheme surpasses CRDSA-NOMA, demonstrating better robustness under severe network congestion. This transition highlights that while CRDSA-NOMA is effective under moderate loads, MA-QDRL exhibits superior scalability and adaptability in highly congested environments. In particular, MA-CDRL's convergence inefficiency becomes increasingly pronounced, reaffirming its unsuitability in large-scale multi-agent IRSA-NOMA systems. Meanwhile, MA-QDRL maintains its superior performance and scalability, demonstrating robustness even as network congestion grows.

Beyond empirical performance, computational efficiency is a critical factor in real-time multi-agent systems. The proposed MA-QDRL model adopts a VQC architecture with a computational complexity of $\mathcal{O}(N \cdot L_Q)$, where N and L_Q denote the number of qubits and quantum circuit layers, respectively. In contrast, the classical MA-CDRL model is based on a fully connected multilayer perceptron (MLP) with L_C hidden layers of size H, resulting in a complexity of $\mathcal{O}(L_C \cdot H^2)$. Specifically, the CDRL model follows an architecture of $I \to H \to H \to O$, where I, H, and O represent the number of input, hidden, and output neurons, respectively. The total number of multiply-accumulate (MAC) operations per forward pass is $\mathcal{O}(I \cdot H + H^2 + H \cdot O)$, which is dominated by $\mathcal{O}(H^2)$ when I and O are small. With two hidden layers ($L_C = 2$) and $H = 8$, the overall complexity becomes $\mathcal{O}(2 \cdot 8^2) = \mathcal{O}(128)$. In comparison, each layer of the proposed QDRL's VQC includes I RX gates for input encoding, N RY gates for parameterized rotations, and $N - 1$ CZ gates for entanglement. The total gate count across L_Q layers scales as $\mathcal{O}(L_Q \cdot (I + 2N - 1)) = \mathcal{O}(N \cdot L_Q)$. For $N = 8$ and $L_Q = 4$, the resulting complexity is $\mathcal{O}(32)$. Therefore, MA-QDRL achieves lower computational cost than MA-CDRL under typical settings, while maintaining competitive learning performance. This result highlights the scalability and efficiency of MA-QDRL for resource-constrained, real-time wireless access environments.

VI. CONCLUSION

This study proposed a Multi-Agent Quantum Deep Reinforcement Learning (MA-QDRL) framework to optimize IRSA-NOMA systems with finite frame lengths. By leveraging Quantum Neural Networks (QNNs) and parameter shift rules (PSR), the proposed method significantly enhances

convergence speed and scalability in reinforcement learning-based random access optimization. Unlike traditional IRSA-NOMA, which relies on fixed probabilistic repetition, MA-QDRL enables adaptive transmission control based on real-time network conditions, significantly reducing packet loss and optimizing system efficiency. Simulation results show that the proposed MA-QDRL achieves up to 63.2% lower packet loss compared to conventional CDRL and 39.2% compared to CRDSA-NOMA under high system load ($G = 1.0$). The key advantages of MA-QDRL are as follows. First, it achieves faster and more stable convergence compared to CDRL approaches, enabling more efficient training in reinforcement learning. Second, it exhibits high scalability, maintaining efficiency even as the number of network nodes increases. Third, MA-QDRL achieves lower packet losses compared to CDRL, fixed transmission schemes, and random transmission strategies. Lastly, the decentralized decision-making mechanism allows each node to optimize its transmission strategy independently without requiring coordination with other agents. These findings highlight the feasibility of integrating quantum-enhanced reinforcement learning into practical IRSA-NOMA systems, particularly under finite frame lengths and dynamic channel conditions. Future work will focus on extending this approach to more realistic channel estimation scenarios, hybrid QNN-ANN architectures, and hardware-aware quantum model design.

REFERENCES

[1] M. El-Tanab and W. Hamouda, "An overview of uplink access techniques in machine-type communications," *IEEE Netw.*, vol. 35, no. 3, pp. 246–251, May/Jun. 2021.

[2] F. J. Dian and R. Vahidnia, "Random access process enhancements for cellular Internet of Things (CIoT)," in *Proc. IEEE 12th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON)*, Vancouver, BC, Canada, 2021, pp. 0547–0552.

[3] A. Musaddiq, R. Ali, S. W. Kim, and D.-S. Kim, "Learning-based resource management for low-power and lossy IoT networks," *IEEE Internet Things J.*, vol. 9, no. 17, pp. 16006–16016, Sep. 2022.

[4] L. Iiyambo, G. Hancke, and A. M. Abu-Mahfouz, "A survey on NB-IoT random access: Approaches for uplink radio access network congestion management," *IEEE Access*, vol. 12, pp. 95487–95506, 2024.

[5] N. Jiang, Y. Deng, X. Kang, and A. Nallanathan, "Random access analysis for massive IoT networks under a new spatio-temporal model: A stochastic geometry approach," *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5788–5803, Nov. 2018.

[6] N. Jiang, Y. Deng, A. Nallanathan, X. Kang, and T. Q. S. Quek, "Analyzing random access collisions in massive IoT networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6853–6870, Oct. 2018.

[7] D.-S. Kim, T.-D. Hoa, and H.-T. Thien, "On the reliability of Industrial Internet of Things from systematic perspectives: Evaluation approaches challenges and open issues," *IETE Tech. Rev.*, vol. 39, no. 6, pp. 1–32, 2022.

[8] G. Liva, "Graph-based analysis and optimization of contention resolution diversity slotted ALOHA," *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 477–487, Feb. 2011.

[9] N. Iswarya and L. Jayashree, "A survey on successive interference cancellation schemes in non-orthogonal multiple access for future radio access," *Wireless Pers. Commun.*, vol. 120, no. 2, pp. 1057–1078, 2021.

[10] W. J. Ryu, J.-W. Kim, S.-Y. Shin, and D.-S. Kim, "On the performance evaluations of cooperative retransmission scheme for cell-edge users of URLLC in multi-carrier downlink NOMA systems," *Sensors*, vol. 21, no. 21, p. 7052, 2021.

[11] M. S. Ali, H. Tabassum and E. Hossain, "Dynamic user clustering and power allocation for uplink and downlink non-orthogonal multiple access (NOMA) systems," *IEEE Access*, vol. 4, pp. 6325–6343, 2016.

[12] R. Kumar and M. Swarnkar, "QuIDS: A quantum support vector machine-based intrusion detection system for IoT networks," *J. Netw. Comput. Appl.*, vol. 234, Feb. 2025, Art. no. 104072.

[13] M. Bhatia and S. K. Sood, "Quantum computing-inspired network optimization for IoT applications," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5590–5598, Jun. 2020.

[14] M. Bey, P. Kuila, B. B. Naik, and S. Ghosh, "Quantum-inspired particle swarm optimization for efficient IoT service placement in edge computing systems," *Expert Syst. Appl.*, vol. 236, Feb. 2024, Art. no. 121270.

[15] M. Hdaib, S.Rajasegarar, and L.Pan, "Quantum deep learning-based anomaly detection for enhanced network security," *Quant. Mach. Intell.*, vol. 6, p. 26, May 2024.

[16] A. Wong, T. Back, V. A. Kononova, and A. Plaat, "Deep multiagent reinforcement learning: Challenges and directions," *Artif. Intell. Rev.*, vol. 56, pp. 5023–5056, Oct. 2023.

[17] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, "The power of quantum neural networks," *Nat. Comput. Sci.*, vol. 1, no. 6, pp. 403–409, 2021.

[18] S. Park and J. Kim, "Trends in quantum reinforcement learning: State-of-the-arts and the road ahead," *ETRI J.*, vol. 46, no. 5, pp. 748–758, Oct. 2024.

[19] I. N. A. Ramatryana, G. B. Satrya, and S. Y. Shin, "Adaptive traffic load in IRSA-NOMA prioritizing emergency devices for 6G enabled massive IoT," *IEEE Wireless Commun. Lett.*, vol. 10, no. 12, pp. 2713–2717, Dec. 2021.

[20] X. Shao, Z. Sun, M. Yang, S. Gu, and Q. Guo, "NOMA-based irregular repetition slotted ALOHA for satellite networks," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 624–627, Apr. 2019.

[21] M. Ç. Moroğlu, H. M. Gürsu, F. Clazzer, and W. Kellerer, "Short frame length approximation for IRSA," *IEEE Wireless Commun. Lett.*, vol. 9, no. 11, pp. 1933–1936, Nov. 2020.

[22] E. Nisioti and N. Thomos, "Fast Q-learning for improved finite length performance of irregular repetition slotted ALOHA," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 2, pp. 844–857, Jun. 2020.

[23] Y. Feng et al., "Irregular repetition slotted ALOHA scheme with short packets under Rayleigh fading channels," *IEEE Trans. Veh. Technol.*, vol. 73, no. 5, pp. 6700–6713, May 2024.

[24] M. Fernández-Veiga, M. Sousa-Vieira, A. Fernández-Vilas, and R. P. Dáz-Redondo, "Irregular repetition slotted aloha with multiuser detection: A density evolution analysis," *Comput. Netw.*, vol. 234, Oct. 2023, Art. no. 109921.

[25] E. Casini, R. De Gaudenzi, and O. Del Rio Herrero, "Contention resolution diversity slotted ALOHA (CRDSA): An enhanced random access scheme for satellite access packet networks," *IEEE Trans. Wireless Commun.*, vol. 6, no. 4, pp. 1408–1419, Apr. 2007, doi: 10.1109/TWC.2007.348337.

[26] A. Mengali, R. De Gaudenzi, and P.-D. Arapoglou, "Enhancing the physical layer of contention resolution diversity slotted ALOHA," *IEEE Trans. Commun.*, vol. 65, no. 10, pp. 4295–4308, Oct. 2017.

[27] S. Alvi, S. Durrani, and X. Zhou, "Enhancing CRDSA with transmit power diversity for machine-type communication," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 7790–7794, Aug. 2018.

[28] H. Okada et al., "Adaptive transmission control considering available channels difference among terminals in satellite IoT systems using CRDSA/IRSA," *IEEE Access*, vol. 13, pp. 66420–66431, 2025.

[29] B. Zhao, G. Ren, X. Dong, and H. Zhang, "Optimal irregular repetition slotted ALOHA under total transmit power constraint in IoT-oriented satellite networks," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 10465–10474, Oct. 2020.

[30] S. Qin, G. Ren, Y. He, and D. Guan, "Spatial group-based NOMA-IRSA with adaptive degree distribution in IoT-enabled WSNs," *IEEE Sensors J.*, vol. 24, no. 21, pp. 35820–35831, Nov. 2024.

[31] J. Choi, "On throughput bounds of NOMA-ALOHA," *IEEE Wireless Commun. Letters*, vol. 11, no. 1, pp. 165–168, Jan. 2022.

[32] W. Fan, P. Fan, and Z. Ding, "On the throughput of NOMA-ALOHA in massive IoT with sparse active users," *IEEE Wireless Commun. Lett.*, vol. 13, no. 3, pp. 582–586, Mar. 2024.

[33] I. N. A. Ramatryana, G. B. Satrya, G. Budiman, and A. B. Mnaouer, "NOMA-ALOHA with prioritized burst repetition for machine-type communications," *IEEE Trans. Veh. Technol.*, vol. 73, no. 11, pp. 17864–17868, Nov. 2024.
[34] J. Su, G. Ren, and B. Zhao, "NOMA-based coded slotted ALOHA for machine-type communications," *IEEE Commun. Lett.*, vol. 25, no. 7, pp. 2435–2439, Jul. 2021.
[35] Y. Ko and J. Choi, "Reinforcement learning for NOMA-ALOHA under fading," *IEEE Trans. Commun.*, vol. 70, no. 10, pp. 6861–6873, Oct. 2022.
[36] X. Wu, Y. Ko, and A. M. Tyrrell, "Distributed multi-agent reinforcement learning for heterogeneous NOMA-ALOHA systems," *IEEE Trans. Cogn. Commun. Netw.*, early access, Oct. 7, 2024, doi: 10.1109/TCCN.2024.3474709.
[37] B. Narottama and S. Y. Shin, "Quantum neural networks for resource allocation in wireless communications," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 1103–1116, Feb. 2022.
[38] B. Narottama, T. Jamaluddin, and S. Y. Shin, "Quantum neural network with parallel training for wireless resource optimization," *IEEE Trans. Mobile Comput.*, vol. 23, no. 5, pp. 5835–5847, May 2024.
[39] B. Narottama and S. Y. Shin, "Federated quantum neural network with quantum teleportation for resource optimization in future wireless communication," *IEEE Trans. Veh. Technol.*, vol. 72, no. 11, pp. 14717–14733, Nov. 2023.
[40] D. Wierichs, J. Izaac, C. Wang, and C. Y.-Y. Lin, "General parameter-shift rules for quantum gradients," *Quantum*, vol. 6, p. 677, Mar. 2022.
[41] *NR Physical Channels and Modulation*, 3GPP Standard TS 38.211, 2020.
[42] P. Popovski et al., "Wireless access for ultra-reliable low-latency communication: Principles and building blocks," *IEEE Netw.*, vol. 32, no. 2, pp. 16–23, Mar./Apr. 2018.
[43] *NR and NG-RAN Overall Description; Stage-2*, 3GPP Standard TS 38.300, 2021.
[44] C. J. Watkins and P. Dayan, "*Q*-learning," *Mach. Learn.*, vol. 8, pp. 279–292, May 1992.
[45] Silvirianti, B. Narottama, and S. Y. Shin, "UAV coverage path planning with quantum-based recurrent deep deterministic policy gradient," *IEEE Trans. Veh. Technol.*, vol. 73, no. 5, pp. 7424–7429, May 2024.
[46] Silvirianti, B. Narottama, and S. Y. Shin, "Layerwise quantum deep reinforcement learning for joint optimization of UAV trajectory and resource allocation," *IEEE Internet Things J.*, vol. 11, no. 1, pp. 430–443, Jan. 2024.
[47] M. Han, *Reinforcement Learning Approaches in Dynamic Environments*. Paris, France: Télécom ParisTech, 2018.
[48] S. Zhang et al., "On the convergence and sample complexity analysis of deep Q-networks with epsilon-greedy exploration," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, 2023, pp. 1–3.
[49] H. V. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 2094–2100.
[50] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. 7th Int. Conf. Learn. Represent.*, 2019, pp. 1–19.
[51] T. Kobayashi and W. E. L. Ilboudo, "t-Soft update of target network for deep reinforcement learning," *Neural Netw.*, vol. 136, pp. 63–71, Apr. 2021.
[52] S. Park, G. S. Kim, Z. Han, and J. Kim, "Quantum multi-agent reinforcement learning is all you need: Coordinated global access in integrated tn/ntn cube-satellite networks," *IEEE Commun. Mag.*, vol. 62, no. 10, pp. 86–92, Oct. 2024.

WON JAE RYU (Member, IEEE) received the B.S. degree in electronic engineering, and the M.S. and Ph.D. degrees in IT convergence engineering from the Kumoh National Institute of Technology (KIT), South Korea, in 2012, 2014, and 2020, respectively. From 2014 to 2016, he worked as a Senior Researcher with Bexel Corporation. Since August 2020, he has been serving as a Research Professor with the ICT Convergence Research Center, KIT. His current research interests include, but are not limited to, scheduling and resource allocation in wireless communication for ultra-reliable low-latency communication.

JAE-MIN LEE (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Seoul National University, Seoul, South Korea, in 2005. From 2005 to 2014, he was a Senior Engineer with Samsung Electronics, Suwon, South Korea, where he was a Principle Engineer from 2015 to 2016. Since 2017, he has been an Associate Professor with the School of Electronic Engineering and the Department of IT-Convergence Engineering, Kumoh National Institute of Technology, Gumi, Gyeongsangbuk, South Korea. His current research interests include the smart IoT convergence applications, industrial wireless control networks, UAVs, metaverse, and blockchain.

DONG-SEONG KIM (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Seoul National University, Seoul, South Korea, in 2003, where he was a Full-Time Researcher with the ERC-ACI from 1994 to 2003. From March 2003 to February 2005, he was a Postdoctoral Researcher with the Wireless Network Laboratory, School of Electrical and Computer Engineering, Cornell University, Ithaca, NY, USA. From 2007 to 2009, he was a Visiting Professor with the Department of Computer Science, University of California at Davis, Davis, CA, USA. He is currently the Director of the KIT Convergence Research Institute and the ICT Convergence Research Center (ITRC and NRF Advanced Research Center Program) supported by Korean Government and the Kumoh National Institute of Technology. His research interests include the real-time IoT and smart platforms, industrial wireless control networks, and networked embedded systems. He is a Senior Member of ACM.