IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 19 August 2024, accepted 13 September 2024, date of publication 18 September 2024,
date of current version 30 September 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3462987

RESEARCH ARTICLE

Judgement-Based Deep Q-Learning Framework for Interference Management in Small Cell Networks

PILDO YOON, (Member, IEEE), YUNHEE CHO, JEEHYEON NA, (Member, IEEE), AND JEONGHO KWAK, (Member, IEEE)

Department of Electrical Engineering and Computer Science, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu 42988, Republic of Korea
Intelligent Small Cell Research Section, ETRI, Daejeon 34129, South Korea

Corresponding author: Jeongho Kwak (jeongho.kwak@dgist.ac.kr)

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2018-0-01659, 5G Open Intelligence-Defined RAN (ID-RAN) Technique based on 5G New Radio).

ABSTRACT Small cell technology for future 6G networks allows network operators to increase network capacity by reducing the distance between Base Stations (BSs) and users, thereby increasing wireless channel gains. However, it also leads to significant computational complexity to optimally mitigate inter-cell and/or inter-beam interference by dynamically managing beamforming, transmit power and user scheduling. In this paper, we formulate an optimization problem aiming to maximize the sum utility of users where decision variables are beam pattern selection, user scheduling and transmit power allocation in small cell networks. Next, we capture room for performance enhancement and low computational complexity that existing studies have overlooked by proposing *i)* a novel decision making process of DQN (Deep Q-Network) to jointly learn all decision variables in a single DRL (Deep Reinforcement Learning) model without a curse of dimensionality by adopting a user-specific state to each agent with distributed interference approximation meaning that interferences to all users in all neighbor BSs can be abstracted by a single user, and *ii)* a novel reward design so that the reward is judged by the result of a practical optimization-based solution. Finally, we show the superiority of the proposed DQL (Deep Q-Learning) algorithm compared to the existing interference management algorithms via simulations and provide insights for network providers who will leverage DQL in future small cell networks through in-depth performance analysis compared with conventional DQL algorithm and practical optimization algorithms.

INDEX TERMS Deep Q-learning, judgement-based learning, beam pattern selection, power allocation, user scheduling.

I. INTRODUCTION

Small cell network, deploying numerous cells within small area, is expected to offer enhanced network capacity. Particularly, when integrated with advanced communication technologies, e.g., mmWave Multi-Input and Multi-Output (MIMO), it is expected to offset limitations and alleviate high interference in the small cell network, i.e., vulnerability in long-range communication and interference reduction with directional beams. Despite these advantages, however, network operators face significant challenges as follows. The proliferation of cells leads to exponential increase in possible combinations of decision variables, substantially raising computational complexity for network resource management, including dynamic beamforming, user scheduling, and transmit power allocation [1]. As a result, several optimization-based techniques failed to achieve the optimality of the formulated problem due to the high computational complexity of the algorithms and hence, some researchers proposed heuristic-based sub-optimal algorithms.

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

Recently, Open RAN (Radio Access Network) architecture in 6G network stands at the forefront of network deployment, specifically within the small cell networks, by incorporating interoperability and standards across RAN components. This architecture significantly enhances the integration ability of the network with cutting-edge technologies such as mmWave MIMO, thereby meeting the growing demand for high-capacity and low-latency communications. Moreover, Open RAN facilitates the AI-driven functionalities in the internal components such as non real-time RIC (RAN Intelligent Controller) and near real-time RIC. By adopting an open and AI-based framework, it can drastically reduce the algorithmic complexity for dynamic beamforming, user scheduling and power allocation controls. This not only simplifies the advancement towards more intelligent and efficient network resource management but also enables network operators to rapidly adapt to technological advancements, ensuring a more scalable and robust cellular network ecosystem.

In this context, there have been a number of studies to efficiently unravel the computational complexity of network resource management with the aid of AI (Artificial Intelligence). For example, RL (Reinforcement Learning), that agent learns optimal action which maximizes the reward for a given state by interacting with the environment, has been actively studied from Q-learning to DRL (Deep RL) for efficient and enhanced resource management [2], [3], [4]. However, to fully leverage RL, there are not only needs for sophisticated reward design but also challenges in learning due to the exponential increase in the size of the state and action space as the dimension of decision variables expands. To this end, there exist some studies that proposed algorithms based on a blend of AI (e.g., DQL (Deep Q-Learning) and DNN (Deep Neural Network)) and optimization to control beamforming, user scheduling and transmit power allocation simultaneously [3], [4], [5]. For example, Ahmed et al. considered sum-rate maximization problem of which solution is to sequentially solve the user grouping, group-beam pairing and transmit power allocation [4]. In this work, DRL was employed for only group-beam pairing, yet transmit power and user grouping were determined by leveraging an optimization framework. However, it is noted that the agent was unable to learn the overall relationships among decision variables in the DRL framework since authors disentangled the intertwined decision variables. The reason authors adopted this approach is due to the high dimensionality of a set of state and action spaces, resulting in exceedingly complex interference relationships in resource management. To solve this issue, it requires tremendous size of DNN model and computing power, yet it is not realizable for each BS in small cell networks.

Practical transmit power control and user scheduling in multi-cell networks under dynamic channel conditions has been proposed with distributed manner [6]. In this work, they abstracted the interference received by all users in neighboring cells into the interference experienced by the most interfered users in those cells. Then, they exchanged this interference information, i.e., channel gains between BSs at long intervals and exchanged only the indices of the users scheduled by neighboring BSs at each time slot to operate in a distributed manner. We conjecture that this distributed nature can reduce the state and action spaces in the DRL framework by individually making Q-value for each user in each BS, and then selecting the best one for user scheduling. Note that conventional centralized optimization solution cannot achieve this since all states and actions are interdependent across all BSs.

Hence, in this paper, we propose a novel decision-making process of DQN (Deep Q-Network) that reduces the dimensionality of a set of the state and action spaces, enabling the agent to learn the relationships among all decision variables, i.e., beam pattern selection, user scheduling, and transmit power allocation. Moreover, we design a reward function so that the reward is judged by the result of a practical optimization-based solution, which are overlooked in prior studies. The DRL algorithm improves its performance by designing the reward function to encourage comparisons with the optimization-based solution [6], motivating the algorithm towards better outcomes.

Meanwhile, existing studies have shown performance superiority such as execution time or network performance and benefits of learning methods such as partial feedback compared with optimization-based algorithms. On top of them, we analyze the operational aspects of the existing optimization-based and learning-based solutions in small cell networks in perspectives of network operators. Our contributions are summarized as follows.

1) Novel decision-making process: we design decision-making process of agent by leveraging the nature of DQN with reasonable size of states and actions. Here, we integrate all decision variables into a single learning process so that each agent learns the relationship among entire decision variables, which were not addressed in the existing DQN solutions due to the complexity issue.
2) Judgement-based reward design: we design a reward so as to exploit both advantages of ‘‘DQL’’ and ‘‘approximated optimization solution’’ by accelerating or breaking the reward for action according to the judgement from a practical optimization-based solution.
3) Performance evaluation and analysis: we show the superiority of the proposed DQL, and insights to utilize DQL in small cell networks for network operators with operational analysis between optimization and learning.

In the rest of this paper, we begin with the related work in Section II. Then, we provide the system model in Section III. In Section IV, we formulate the stochastic optimization problem. Next, we describe the proposed DQN-based algorithm in Section V. We provide the simulation results in Section VI. Finally, we conclude this paper in Section VII.

II. RELATED WORK

Resource management in multi-cell networks has been widely studied aiming to maximize utility of users where the utility captures both time-averaged throughput of users and fairness of them. However, the joint optimization of various control variables in the existence of inter-cell and inter-beam interference makes the computational complexity of the algorithms significantly high. To this end, many researchers tried to make tractable algorithms with considerations of realistic aspects of cellular networks in terms of feedback information and computational complexity. The tractable algorithms can be categorized by two theoretical approaches.

The first approach is to make low-complex and heuristic algorithms on top of the optimization theory [6], [7], [8], [9], [10]. For example, Son et al. [7] proposed a practical joint user scheduling and power allocation algorithm by introducing a concept of reference user in single antenna multi-cell network environment. Here, the single reference user for each cell abstracts complex inference relationship from the corresponding BS to scheduled users in neighboring cells; this enables significant complexity reduction of the algorithm. Moreover, Hong et al. [8] reduced the complexity of the algorithm by decomposing the slot-by-slot problem into two subproblems with different time scales where the first subproblem is to find beam activation probability with a long time-scale and the second subproblem is to find user scheduling and power allocation with a short time-scale. Subsequently, they further reduced the computational complexity by sequentially making decisions of beam activation, user scheduling, and power allocation. In addition, Yoon et al. [6] proposed a low-complex and practical interference management algorithm by introducing critical user and power sharing virtual queue concepts which maximally exploit structural characteristics of hybrid centralized and distributed network architecture, namely EdgeSON to heuristically solve the optimization problem aiming to maximize the time-averaged utility of users constrained by the time average transmit power budget. Besides, Zhang et al. [10] focused on user association and power allocation control in ultra-dense networks with energy-harvesting BSs. Their problem is to balance load and manage cross-tier interference while meeting user QoS requirements. The problem is formulated as MIP (Mixed Integer Programming) and relaxed to convex, solved using Lagrangian dual decomposition with an iterative gradient-based algorithm. The proposed algorithm efficiently balances load and allocates power, meeting energy and QoS (Quality of Service) constraints.

The second approach is to apply a variant of learning methods to develop the algorithms [2], [3], [4], [5], [11], [12], [13]. For example, Amiri et al. [2] proposed a transmit power control algorithm using the reinforcement learning framework in the joint existence of a macro cell and femto cells. Here, they mainly focused on the impact of reward design for the reinforcement learning on the sum transmission rate in all cells. Moreover, Braga et al. [3] proposed a user scheduling solution with multi-agent DQL for solving beamforming problem which maximizes total rate in a multicell MISO (Multiple Input Single Output) system in the existence of wireless channel estimation errors. In addition, Ge et al. [12] considered homogeneous cellular networks with multi-antenna BSs using universal frequency reuse. Their problem is that downlink-beamforming coordination helps BSs manage interference, but it is impractical in dynamic environments due to computational complexity and significant overhead. To solve this issue, a DRL-based solution is proposed where each BS trains a Q-network to determine the appropriate beamformer using partially observable CSI (Channel State Information). Besides, Sun et al. [13] used DNN (Deep Neural Networks) to approximate the nonlinear mappings of signal processing algorithms, particularly WMMSE (Weighted Minimum Mean Squared Error) for power allocation in an interference management problem. Although these studies adopted DRL/DNN or optimization theory as the solution frameworks for power control, user scheduling and/or beamforming, there is no work to apply the DRL framework into entire beamforming, user scheduling and power control system due to the high complexity to design states and actions.

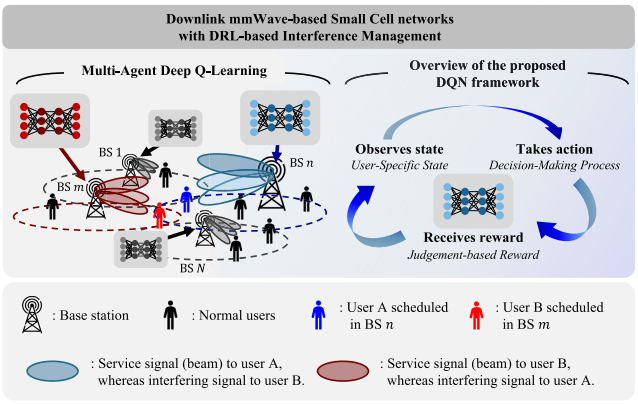


FIGURE 1. Judgement-based deep Q-learning framework.

III. SYSTEM MODEL

In this paper, we consider downlink mmWave-based small cell networks with DRL-based interference management as shown in Fig. 1. In wireless network, the service signal for specific user (e.g., User B in Fig. 1) can interfere the service signal for other users (e.g., User A in Fig. 1). Moreover, this interference forms a highly complex relationship depending on various factors such as decision variables and wireless channel conditions, and it is a key factor that degrades network performance (e.g., throughput). Therefore, in this paper, we aim to efficiently manage this complex interference by leveraging the advantages of learning. Specifically, we adopt deep Q-learning while proposing a decision-making process with user-specific state and judgement-based reward design.

We consider a time-slotted system indexed by t and OFDM (Orthogonal Frequency Division Multiplexing) system with a set of subchannels[1] $\mathcal{O} = \{1, \ldots, o, \ldots, O\}$. In this network, there are a set of Base Stations (BSs) $\mathcal{N} = \{1, \ldots, n, \ldots, N\}$ with L transmit antennas and a set of users associated with BS n, $\mathcal{K}_n = \{1, \ldots, k_n, \ldots, K_n\}$ with a single receive antenna. To specify whether user is scheduled for network service, we define user scheduling indicator as $I^o_{n,k_n}(t) \in \{0, 1\}$. Here, $I^o_{n,k_n}(t) = 1$ means user k_n is scheduled on subchannel o of BS n at time slot t, and vice versa. To denote beamformer of BS n for subchannel o at time slot t, we define $\mathbf{w}^o_n(t) = \sqrt{p^o_n(t)}\bar{\mathbf{w}}^o_n(t)$ where $p^o_n(t)$ and $\bar{\mathbf{w}}^o_n(t) \in \mathbb{C}^{L\times 1}$ mean the allocated transmit power and beam pattern which is a set of directed beams, for subchannel o of BS n at time slot t, respectively. Here, we define a set of patterns $\mathcal{X} = \{X_1, \ldots, X_B\}$ with codebook design [14] where each pattern consists of following elements:

$$X_b = \left\{ \frac{1}{\sqrt{L}} \exp\left(j\frac{2\pi}{S} \left\lfloor \frac{\frac{l\cdot \mathrm{mod}(b+\frac{B}{2}, B)}{2}}{\frac{B}{A}} \right\rfloor \right) \middle| \forall l \in \{1, \cdots, L\} \right\}, \tag{1}$$

where $\mathrm{mod}(\cdot)$ and $\lfloor\cdot\rfloor$ are modular and floor operation, respectively. Moreover, A is a parameter used to adjust the beamwidth. In other words, to generate a narrow beam, it can be set to a small value and vice versa. For the pattern set, we define pattern selection indicator as $B^o_{n,b}(t) \in \{0, 1\}$ where $b \in \mathcal{B} = \{1, \cdots, B\}$ to imply that pattern b is selected by BS n for subchannel o at time slot t when $B^o_{n,b}(t) = 1$, i.e., $\bar{\mathbf{w}}^o_n(t) = X_b$, and vice versa.

From the defined system model above, we define SINR (Signal-to-Interference-plus-Noise Ratio) of user k_n which includes inter-beam interference as follows:

$$\mu^o_{k_n}(t) = \frac{\sum_b B^o_{n,b}(t) I^o_{n,k_n}(t) \left| {h^o_{n,k_n}}^\dagger(t)\sqrt{p^o_n(t)}X_b \right|^2}{\eta^o_{k_n}(t) + \sigma^2}, \tag{2}$$

where $h^o_{n,k_n}(t) \in \mathbb{C}^{L\times 1}$ is a direct channel between BS n and user k_n at time slot t and subscript $\dagger$ is Hermitian transpose operation. Moreover, σ and $\eta^o_{k_n}(t)$ denote noise power and interference that user k_n receives at time slot t as follows[2]:

$$\eta^o_{k_n}(t) = \sum_{m\setminus n}\sum_b B^o_{m,b}(t) \sum_{k_m} I^o_{m,k_m}(t) \left| {h^o_{m,k_n}}^\dagger(t)\sqrt{p^o_m(t)}X_b \right|^2. \tag{3}$$

From the defined SINR, the normalized data rate of user k_n on subchannel o at time slot t is calculated as follows:

$$c^o_{k_n}(t) = \log_2(1 + \mu^o_{k_n}(t)). \tag{4}$$

[1]Here, a subchannel is a set of subcarriers and we assume that inter-subchannel interference can be ignored [7].

[2]Hereinafter, the symbol $\setminus$ is used to represent the elements of a set excluding the right-hand element. From the defined system model above, we define SINR (Signal-to-Interference-plus-Noise Ratio) of user k_n which includes inter-beam interference as follows.

IV. PROBLEM FORMULATION

Now, we formulate an optimization problem where the objective is to maximize sum utility of time-averaged data rates of users, $C^o_{k_n} = \lim_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} c^o_{k_n}(t)$, constrained by the maximum power budget and (beam, user) scheduling of each subchannel as follows.

$$\begin{aligned}(P1): \max_{(\boldsymbol{B},\boldsymbol{I},\boldsymbol{p})} & \sum_{n=1}^{N}\sum_{o=1}^{O}\sum_{k_n=1}^{K_n} U(C^o_{k_n}), \\ s.t. & \sum_{b=1}^{B} B^o_{n,b}(t) = 1, \forall n \in \mathcal{N}, \forall o \in \mathcal{O}, \\ & \sum_{k_n=1}^{K} I^o_{n,k_n}(t) = 1, \forall n \in \mathcal{N}, \forall o \in \mathcal{O}, \\ & 0 \le p^o_n(t) \le P^o_{\max}, \forall n \in \mathcal{N}, \forall o \in \mathcal{O},\end{aligned}$$

where the utility function is defined as $U(f(x)) = \log(1 + f(x))$ so as to capture both throughput and fairness among all users [15]. Using this function as the objective function ensures that users with lower throughput receive more weight in the scheduling process. This helps to balance the network by giving more priority to users with poorer connections, thereby improving overall fairness. The logarithmic function grows more slowly at higher values, which means that increases in throughput for users already experiencing high rates contribute less to the objective function than similar increases for users with lower throughput. Besides, $P^o_{\max}$ denotes the maximum transmit power for subchannel o and decision variables are joint pattern selection $\boldsymbol{B}$, user scheduling $\boldsymbol{I}$ and power allocation $\boldsymbol{p}$. The original optimal solution to solve this problem should be based on the optimization theory since this is an optimization problem. However, this problem is known as NP-hard, i.e., there is no known algorithm that can be solved within polynomial time due to the complex inter-beam interference relationships and complex decision variables, i.e., beamforming, user scheduling and power control every time slot, which is MINLP (Mixed Integer and Non-Linear Program) [7], [16]. Hence, we have two options to obtain tractable solution of this problem: *(i)* a heuristic solution with approximation on top of optimization theory, and *(ii)* a DRL solution with a help of optimization-based solution. In this paper, we adopt multi-agent DQN with reasonable state and action space complexity and judgement-based reward design to achieve high performance in the next section.

V. MUTI-AGENT DQN APPROACH WITH NOVEL DECISION-MAKING PROCESS AND REWARD DESIGN

A. DECISION-MAKING PROCESS FOR DQN

In this subsection, we briefly review Q-learning and DQL, and introduce our decision-making process. The Q-learning is a well-known model-free learning that agent learns the optimal action by updating values of state-action (Q-values).

Typically, Q-value is updated as follows [2]:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left(r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right), \tag{5}$$

where s, a and r mean state, action and reward emitted from environment by acting a for state s, respectively, and s' corresponds to next state given from environment after action a. Moreover, α and γ refer learning rate in regard to the degree of update and discount factor that indicates the level of expectations about the future, respectively. In update formula (5), each agent learns the optimal action for a given state by exploiting known rewarding actions and exploring unfamiliar actions that may lead to promising rewards in the future. However, it requires each agent to maintain a Q-table for all combinations of states and actions. Hence, as the number of states and actions increases, the size of Q-table exponentially increases. To tackle this, by replacing Q-table with DNN, the agent can approximate the complex relationship between state and action without maintaining huge Q-table [4]. This manner is typically called DQL and each agent learns Q-values by minimizing loss invoked by target network parameterized with θ^- and train network parameterized with θ [17]. This loss is illustrated with MSE (Mean Squared Error) as follows:

$$\mathcal{L} = \left(r + \gamma \max_{a'} Q(s',a';\theta^-) - Q(s,a;\theta) \right)^2. \tag{6}$$

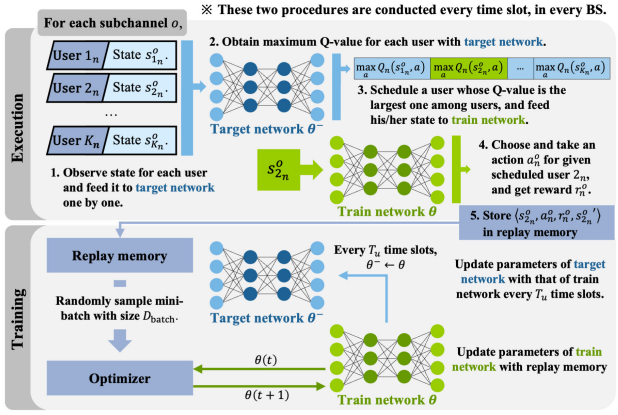


FIGURE 2. The proposed decision-making process of agent.

Meanwhile, existing studies to solve the IM (Interference Management) problem have adopated this DRL framework. However, as mentioned earlier, they have utilized DRL for a subset of decision variables, and adopted other frameworks, e.g., optimization, for the rest of decision variables due to the inherent complexity of IM. For example, let us consider that there are two agents (BSs) learning actions for four beam patterns, two users and $\{0, P_{\max}\}$ transmit power levels. Here, the number of possible (BS, beam pattern, user, transmit power level) combinations for interference is $4^2 \times 2^2 \times 2^2$, which drastically increases to $4^N \times 2^N \times 2^N$ as the number of BSs increases to N. Hence, even under a simple assumption of action space, the possible combinations of states might be extremely high. Accordingly, to learn this complex relationship for interference, a tremendous size of neural network is required with longer training time and rich computing resources.

Hence, to resolve this curse of dimensionality issue, we focus on the fact that the largest Q-value for the current state exhibits the value of state. As a result, it leads to the insight that if we design states standing for each user, the agent can produce the value of state, i.e., the value of user, as Q-value with learned action. This user-specific state design cannot be realizable due to the high number of state combinations for the original intertwined inter-beam interference relationships for all users. However, if we adopt interference abstraction technique of reference user concept in [6], then states can be modeled for each user.In other words, in the previous work [6], they abstracted the interference received by all users in neighboring cells into the interference experienced by the most interfered users in those cells, thereby operating in a distributed manner, i.e., each BS can manage interference with small feedback exchange among neighboring BSs. Therefore, we follow this model to design our DQN framework so that states and actions can be modeled for each user. Then, by scheduling user having the largest Q-value at each iteration,[3] the remained beam pattern selection and transmit power allocation can be chosen as actions by feeding his/her state to the agent. This approach can reduce the size of action space as well as the number of possible combinations of states for interference; hence we can train the agent with small-sized DNN model.

Finally, this decision-making process can be illustrated as shown in Fig. 2. Here, target network and train network have the same size where the input size corresponds to states of each user and the output size corresponds to the number of available beams multiplied by the number of transmit power levels. Hence, if there are K_n users in BS n, each BS yields K_n Q-values for K_n target networks.

In the execution phase, each agent, i.e., BS first observes states for all associated users and feeds them to target network one by one. Then, the agent selects the largest Q-value as output of the target network for each user and schedules a user whose Q-value is the largest. Next, the agent determines action, i.e., beam pattern selection and transmit power allocation, for the scheduled user with a train network, and receives next state from environment and then stores the tuple of state, action, reward and next state to replay memory. Here, tuples generated from the same BS for all subchannels are stored in the same replay memory to let the agent learn for the general state.

In the training phase, the agent randomly samples mini-batches with size D_{btc} from replay memory of which the buffer size is R_{buf}. Next, the agent updates the train network

[3]This is because that the objective of DQN is to maximize Q-value every time slot.

by minimizing MSE loss (6). For every T_u time slots, the agent updates the parameters of its target network with that of the train network. In this process, target network is used to obtain the value of user to be scheduled, as its parameters do not change during T_u time slots. This ensures stable learning procedure for stable user scheduling. Moreover, this process can be conducted with afforded complexity thanks to DNN with fixed parameters [11].

B. STATE, ACTION, AND REWARD DESIGN

Next, we define state, action and reward function for our decision-making process design. Here, to simplify notations, we omit indices of subchannel o and time slot t since state, action and reward are defined for a single subchannel and a single time slot. To feed a user-specific state to agent, we follow interference approximation concept in a practical optimization-based algorithm [7].[4] Hence, the state of user k_n is defined with following elements: (i) time-averaged data rate, $w_{k_n} = \frac{1}{t}\sum_{\tau=0}^{t-1} c_{k_n}(\tau)$, (ii) normalized channel gain from BS n for each beam pattern, $\{|h_{n,k_n}^{\dagger} X_b|^2 / g_{\max} | \forall b \in \mathcal{B}\}$, where $g_{\max} = \max_{b\in\mathcal{B}} |h_{n,k_n}^{\dagger} X_b|^2$, (iii) distance from BS n divided by the BS radius and (iv) normalized maximum receiving interference from other BSs, $\{\max_b |h_{m,k_n}^{\dagger} X_b|^2 / g_{\max} | \forall m \in \mathcal{N} \setminus n\}$. Moreover, to give information about harm to others compared to own gain, we define a victim for each beam pattern X_b as $v_{n,b} = \arg\max_{k\in\mathcal{K}\setminus\mathcal{K}_n} |h_{n,k}^{\dagger} X_b|^2$. Accordingly, we define (v) time-averaged data rate of victims, $\{w_{v_{n,b}} | \forall b \in \mathcal{B}\}$, (vi) normalized maximum interference towards other users for each beam pattern, $\{|h_{n,v_{n,b}}^{\dagger} X_b|^2 / |h_{n,k}^{\dagger} X_b|^2 | \forall b \in \mathcal{B}\}$, and (vii) distances between victims and BS n divided by BS radius. Although we assume perfect CSI (Channel State Information) such as [4], each agent distributedly requires only CSI of users in a corresponding BS and edge users in the neighboring BSs, hence we can obtain them with reasonable overheads like [7].

Second, we define action set $\mathcal{A}$ as discrete transmit power set with quantization level P_{qn} as follows:

$$\mathcal{A} = \left\{ (p, b) \middle| \forall p \in \left\{ 0, \frac{1}{p_{\text{qn}} - 1} p_{\max}, \ldots, p_{\max} \right\}, \forall b \in \mathcal{B} \right\}. \tag{7}$$

Note that action space should be $P_{\text{qn}} B K_n$ to control all combinations of variables in a single DQN model. However, in our decision-making process, action space is limited on $P_{\text{qn}} B$ while using a single model since our action space is defined per each user.

Now, we propose a novel reward function which consists of judgement term and objective term. For judgement term, we leverage the result from a practical optimization-based algorithm in [6] as follows (see equation (21) of [6]):

$$p_n = \left[\frac{1/w_{k_n}}{\ln 2 + \text{tax}_n} - \frac{\sum_{m\neq n} \sum_b B_{m,b} |h_{m,k_n}^{\dagger} X_b|^2}{\sum_b B_{n,b} |h_{n,k_n}^{\dagger} X_b|^2} \right]_0^{P_{\max}}, \tag{8}$$

where tax_n is calculated every time slot to quantify the amount of interference towards a user whose received inteference from selected action of BS n is the largest among scheduled users in other BSs. This optimization-based algorithm showed high throughput performance with practical computational complexity. Indeed, the complexity of the above closed form power allocation per cell is $O(N)$ where N is the number of users, which is quite low, and this equation allows distributed computation at each BS. Additionally, the required information per time slot includes the user indicators scheduled by each BS and the long-term average interference, minimizing the feedback needed. Hence, we devise a way to provide a judgement for the action of agent where the criterion is given by the result of this optimization-based algorithm, namely ULTIMA in [6]. In this regard, we introduce a judgement term as follows:

$$r_{n,\text{jud}} = \sum_m \sum_{k_m} \frac{c_{k_m}}{w_{k_m}} - \sum_m \sum_{k_m} \frac{c_{k_m}(\boldsymbol{a}_n^{\text{REF}})}{w_{k_m}}, \tag{9}$$

where $\boldsymbol{a}_n^{\text{REF}}$ is an action tuple of all BSs where transmit power of BS n is replaced by p_n calculated from (8). This reward, being designed as the difference between sum data rate divided by time-averaged data rate with current selected action and that from ULTIMA in [6], can be either negative value or positive value. A negative value happens when the result of ULTIMA is higher than that from currently learned action, and vice versa. As a result, this reward forces the agent to learn in two ways. In one way, the agent learns to adjust its action to minimize negative reward, thereby mimicking the solution of ULTIMA. In the other way, the agent seeks better action than ULTIMA to increase positive reward. Finally, we define reward function with objective term and judgement term as follows:

$$r_n = \frac{c_{k_n}}{w_{k_n}} + r_{n,\text{jud}}. \tag{10}$$

where objective term c_{k_n}/w_{k_n} is introduced to capture our objective function in a long-term perspective. In other words, the maximization of c_{k_n}/w_{k_n} results in the maximization of our objective function in **(P1)** [18].

C. ALGORITHM DESCRIPTION

The description of the proposed decision-making algorithm is shown in Algorithm 1. Every time slot and subchannel, an agent, i.e., BS observes the user-specific states as described in Section V-B. Next, in the order of observed states for users, the agent obtains Q-values for each state using the target network and stores the highest value among them. Through this step, the Q-values of all users is obtained, hence the agent can schedule a user whose Q-value is the largest. For remaining action, transmit power and beam pattern, the

[4]This approximation means that each BS decides transmit power by considering only a user who receives the highest interference from BS [7]. This distributed feature makes original RL to transition from single agent (before approximation) to multi-agent with user-specific states (after approximation).

Algorithm 1 Decision-Making Process Description.

This algorithm is independently operated in each BS n.
Output: A tuple of $(\boldsymbol{B}_n, \boldsymbol{I}_n, \boldsymbol{p}_n)$
Initialization:
Zero-initialization for decision variables:
$\boldsymbol{B}_n = \{0\}, \boldsymbol{I}_n = \{0\}, \boldsymbol{p}_n = \{0\}$.
Action set for transmit power and beam pattern, (7).
$\mathcal{A} = \{a_1, \ldots, a_{P_{\text{qn}}B}\}$.
Randomly initialize the parameters of train network and target network, $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^-$.

while $t \leq T$ **do**
  for $o \in \mathcal{O}$ **do**
    Observe the user-specific states for all subchannels:
    $\boldsymbol{s}^o = \{s_1^o, \ldots, s_{|\mathcal{K}_n|}^o\}$.
    Initialize Q-values set for user scheduling:
    $V \leftarrow \{\}$.
    for $s \in \boldsymbol{s}^o$ **do**
      Obtain Q-values for state s with target network:
      $V_s = \{Q(s, a_1; \boldsymbol{\theta}^-), \ldots, Q(s, a_{|\mathcal{A}|}; \boldsymbol{\theta}^-)\}$.
      Store the largest Q-value among V_s:
      $V \leftarrow \{V, \max V_s\}$.
    end
    Schedule a user whose Q-value is the largest one among users:
    $I^o_{n,k_n}(t) = 1$, where $k_n = \arg\max V$.
    Obtain Q-values for state of user k_n with train network:
    $V_{k_n} = \{Q(s_{k_n}, a_1; \boldsymbol{\theta}), \ldots, Q(s_{k_n}, a_{|\mathcal{A}|}; \boldsymbol{\theta})\}$.
    Select beam pattern and transmit power based on ϵ-greedy policy with exploration rate ϵ:
    with probability ϵ:
      Select action $a = (p, b)$ uniformly randomly from action set, $\mathcal{A}$.
    with probability $1 - \epsilon$:
      Select action $a_{\arg\max V_{k_n}} = (p, b)$ of which Q-value is the largest one.
    $p_n^o(t) = p$ and $B^o_{n,b}(t) = 1$.
  end
  Update ϵ with decreasing rate of ϵ_{dc}:
  $\epsilon = \max\{\epsilon_{\min}, (1 - \epsilon_{\text{dc}})\epsilon\}$.

agent selects action with train network based on ϵ-greedy policy. Here, the ϵ-greedy policy allows the agent to take action randomly with a probability of ϵ (exploration); on the other hand, it chooses the action with the highest Q-value among the actions (exploitation). In the end of time slot, the agent updates ϵ with decreasing rate of ϵ_{dc} so that sufficient exploration of various cases is ensured at the beginning stage and the agent exploits the learned actions consistently at the later stage.

VI. PERFORMANCE EVALUATION

A. SIMULATION SETUP

In this section, we first provide our simulation settings summarized in Table 2. First, we model the mmWave wireless channel with Rayleigh fading channel model based on the first-order Gauss-Markov process [19] with its correlation coefficient 0.64 and UMi Street Canyon path loss model [20]. For network topology, we consider a homogeneous network where the distance between any two BSs is uniform with BS coverage diameter. In this setup, we deploy 7 BSs (agents) and 12 dummy BSs that cause interference with randomly selected patterns and maximum transmit power. Moreover, we construct fully connected DNN for DQN with two hidden layers consisting of 128 and 64 nodes and use epsilon-greedy method which decreases its exploration probability at rate of 10^{-4}. In all simulations in this paper, we set total 70,000 time slots and calculate the average performance from 50,000 time slot to 70,000 time slot after convergence of the learning. Note that control parameters in this simulation are a tuple of transmit power control, beamforming and user scheduling.

TABLE 1. System parameters.

System Parameters	Value
Center frequency	28GHz
The number of subchannels, O	5
The maximum transmit power per subchannel, $P^o_{\max}$	0.2W
Subchannel bandwidth, BW	100MHz
The number of transmit antennas, L	3
The number of patterns, B	4
Power quantization level, P_{qn}	5

TABLE 2. Hyper parameters.

Hyper Parameters	Value
Replay memory buffer size, R_{buf}	2500
Size of mini-batch, D_{btc}	64
Target network update period, T_u	100
Learning rate, α	0.005
Optimizer for train network	RMSprop

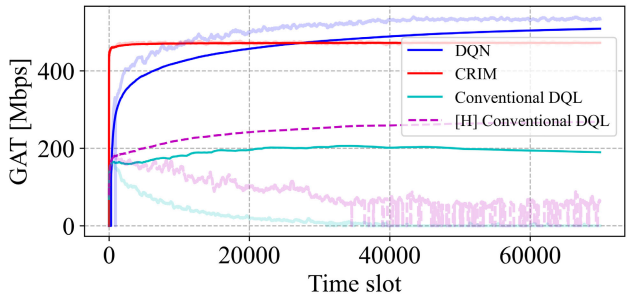
To evaluate the performance of our DQN framework, we consider four benchmark algorithms as follows: *i*) *CRIM* [8] is Lyapunov optimization-based algorithm with probabilistic pattern selection and virtual queue-based[5] user scheduling and transmit power allocation algorithm, *ii*) *PF-DQN* is DQL algorithm that adopts PF (Proportional Fairness) user scheduling and same DQN for the remained decision variables. In other words, PF-DQN is following the conventional approaches [3], [4], [5] where the links among decision variables are untangled. *iii*) *MATCH* [9] is a Gale-Shapely matching algorithm that each BS matches user and subchannel based on channel gain as preferences for possible user scheduling scenarios. Then, each BS selects a match result that provides the highest sum data rate divided by time-averaged data rate, and *iv*) *ONOFF* [21] is an algorithm that turns on BSs with maximum transmit power of which the giving interference is lower than a predetermined threshold and schedules users with PF scheduling. Moreover, we use performance metrics as GAT (Geometric Average of Throughput for all users) which captures our objective

[5]It denotes a typical data structure with First-In-First-Out manner. In [8], authors create virtual queues for each user to express how much service they have received by designing departure of queue as achieved data rate.

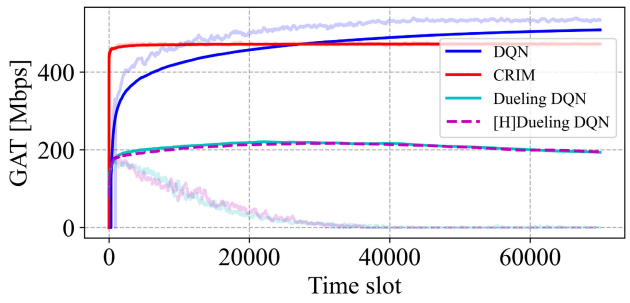
TABLE 3. Neural network configurations.

	Configuration 1	Configuration 2	Configuration 3	Configuration 4
Layer 1	128	512	512	512
Layer 2	64	1024	1024	1024
Layer 3	-	512	512	512
Layer 4	-	256	256	512
Layer 5	-	-	256	-
Batch size	64	256	256	-
Replay buffer	2500	10000	10000	-
FLOPs	16.4K	2.36M	2.42M	2.62 M

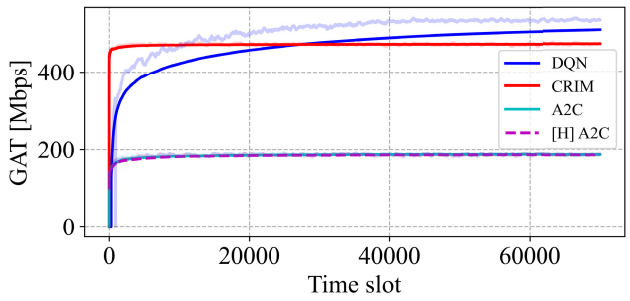
function in **(P1)** thanks to the shape of log function and Avg. Throughput (Average Throughput for all users).



(a) Conventional DQN.

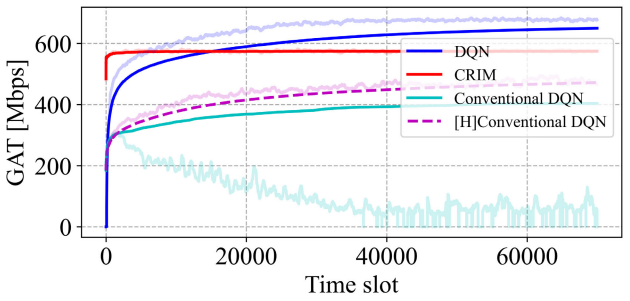


(b) Dueling DQN.

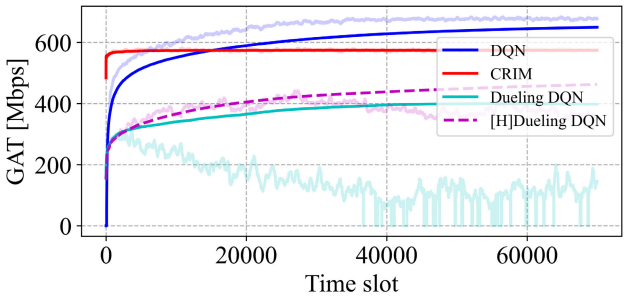


(c) A2C.

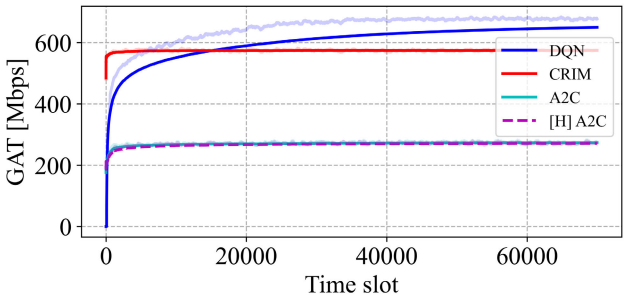
FIGURE 3. Convergence performance for the number of users per BS as 8.



(a) Conventional DQN.



(b) Dueling DQN.



(c) A2C.

FIGURE 4. Convergence performance for the number of users per BS as 5.

B. SIMULATION RESULTS

1) CONVERGENCE AND COMPUTATIONAL COMPLEXITY

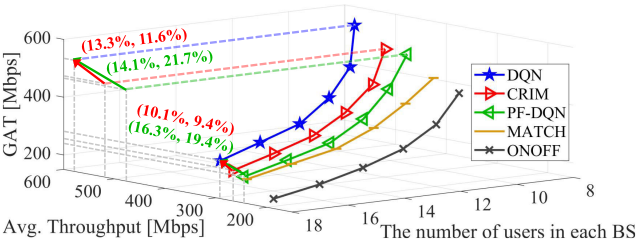
First, we compare the decision-making process of our proposed method with conventional DRL (Deep Reinforcement Learning) methods to assess the convergence performance and computational complexity. Conventional DRL methods refer to a scenario where each agent receives the state for the entire users associated in each BS and then makes decisions on action. As benchmark algorithms for conventional DRL methods, we use Conventional DQN, Dueling DQN, and

A2C (Actor-Critic). Moreover, these approaches utilize two neural network configurations: i) the first one represents the small-sized neural network configuration used in our proposed method, and ii) the second one denotes a wider and deeper neural network. Specifically, we represent each configuration in Table 3 where configuration 1 is small-sized neural network used for our proposed method, and configuration 2, 3 and 4 are heavy neural network used for conventional DQN, Dueling DQN and A2C, respectively.

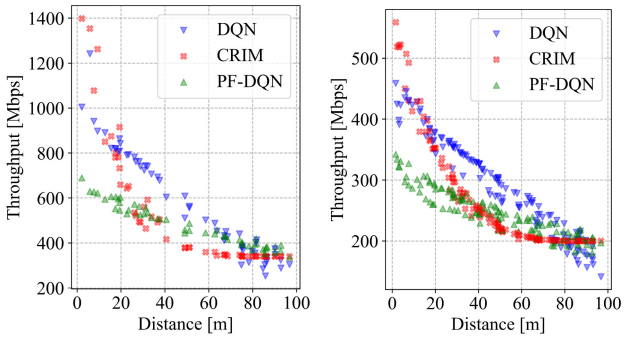
We conduct simulations on the network where the number of users in each BS is 8 and the radius of BS is 100m. Finally, we illustrate the simulation results as shown in Fig. 3, and we mark [H] (Heavy) indicator for methods using configuration 2, 3, and 4 in the figure legend. In each subfigure, the translucent lines indicate the GAT performance calculated using a sliding window manner with a size of 500 time slots, while the opaque lines represent the cumulative GAT performance measured at each time slot. As illustrated in each subfigure, the conventional DRL methods with small-sized neural network configuration fail to achieve high performance and stable learning. Moreover, even when the neural network size is expanded, these methods do not reach successful and stable learning procedure. In contrast, our proposed approach demonstrates high performance and stable learning with even small-sized neural network. Moreover, this impact can be highlighted by comparing the FLOPs (FLoating Point Operations), where FLOPs represent the number of multiplication and addition operations required during the forward of neural network. As shown in Table 3, the proposed DRL outperforms the benchmark algorithms with lower FLOPs, i.e., computational complexity.

To further validate these observations, we conduct simulations with a reduced number of users per BS and reduced cell radius, say five users and 60m, respectively. The results are shown in Fig. 4. As the number of users decreases, it is observed that the stability of learning in the benchmark algorithms improves compared to previous results. This suggests that utilizing a wider and deeper neural network could potentially lead to more stable and higher performance. However, the benchmark algorithms still fail to achieve high performance and stable learning, even with higher computational complexity than ours. Despite this, our proposed approach, thanks to its decision-making process and distributed nature of interference abstraction, continues to achieve stable learning and high performance even with a small-sized neural network.

In summary, it can be seen that the existing DRL methods are difficult to learn the complex and spatio-temporally varying interference relationships in small cell networks and to control a number of decision variables for managing them, even at the expense of high computational complexity. However, it can be also seen that the proposed DQN enables to efficiently learn the interference relationships thanks to the decision-making process and distributed nature of interference abstraction.

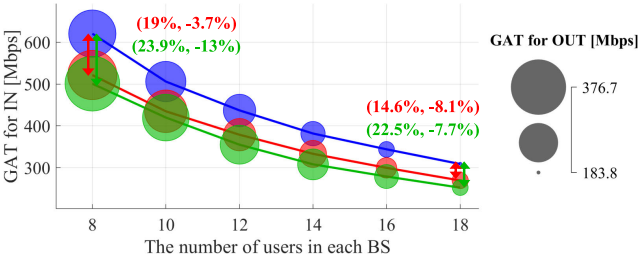


(a) Overall performance for the different number of users. The tuples in the figure represent performance increment rate in terms of GAT and Avg. Throughput, in order.



(b) Avg. Throughput performance according to distance from BS (the number of users: 8).

(c) Avg. Throughput performance according to distance from BS (the number of users: 18).



(d) GAT performance for IN and OUT environment. The tuples in the figure represent performance increment rate in terms of GAT for IN and OUT, in order

FIGURE 5. **Performance analysis for the different number of users.**

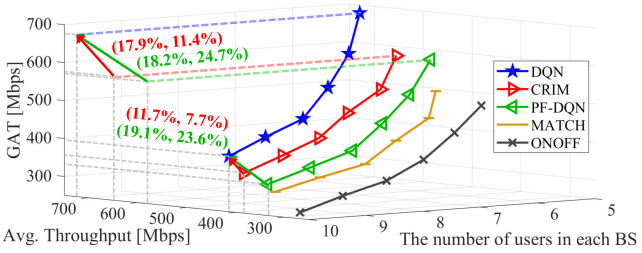
2) THROUGHPUT PERFORMANCE

To examine the overall performance according to the different number of users,[6] we conduct simulations with 100m cell radius where users are randomly distributed in the coverage of their associated BSs. As shown in Fig. 5(a), it can be confirmed that our proposed DQN framework outperforms benchmark algorithms in both GAT and Avg. Throughput. Moreover, the importance of joint learning for decision variables can be confirmed through performance enhancement of our algorithm compared to PF-DQN. Next, we focus on the operational aspects of DQN, PF-DQN and optimization-based algorithm, i.e., CRIM.

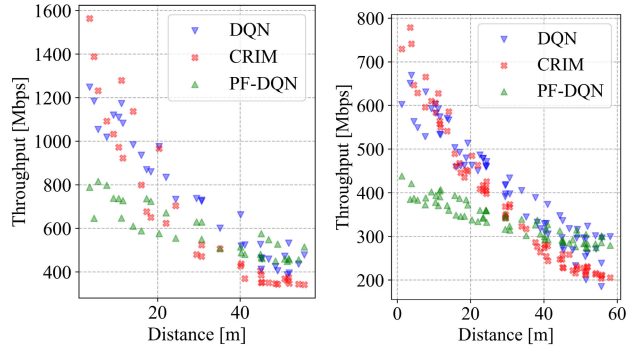
To this end, we first present the Avg. Throughput performance of users according to their distance from

[6]We assume that all BSs associate with identical number of users and associated user sets are disjoint from each other. However, our decision-making process can operate with varying distributions of the number of users.

the associated BS, as shown in Fig. 5(b) and Fig. 5(c). Interestingly, we observe that there are consistent trends in the operational aspects of all three algorithms across simulation settings. Specifically, the proposed DQN method provides higher Avg. Throughput for users closer to their associated BS compared to the other algorithms, while delivering lower Avg. Throughput performance for users who are farther away.

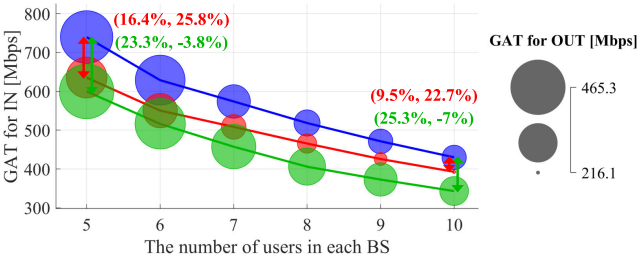


(a) Overall performance for the different number of users. The tuples in the figure represent performance increment rate in terms of GAT and Avg. Throughput, in order.



(b) Avg. Throughput performance according to distance from BS (the number of users: 5).

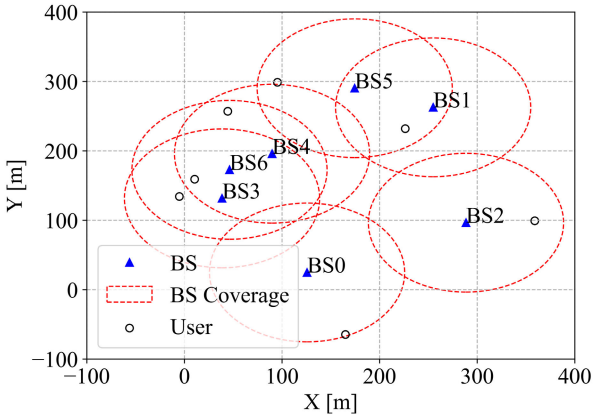
(c) Avg. Throughput performance according to distance from BS (the number of users: 10).



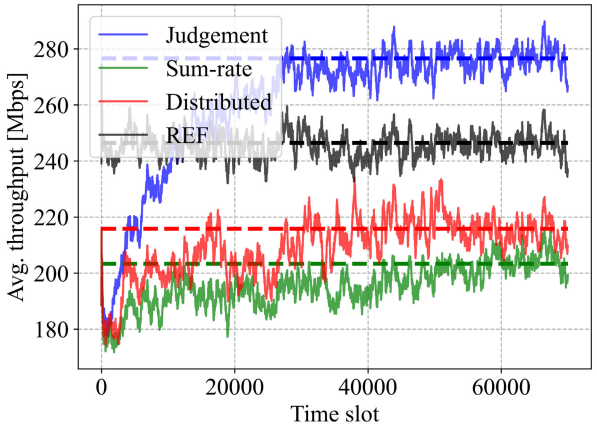
(d) GAT performance for IN and OUT environment. The tuples in the figure represent performance increment rate in terms of GAT for IN and OUT, in order

FIGURE 6. Performance analysis for the different number of users.

To provide a clear view of these results, we define *IN* and *OUT* regions where users are located closer than 80m and users are located farther than 80m from their associated BS, respectively. Next, we represent the GAT performance of users in each region as shown in Fig. 5(d). In this figure, the line plots and symbol sizes represent the GAT performance of the IN and OUT users, respectively. As observed in the Fig. 5(b) and Fig. 5(c), the proposed DQN improves performance for IN users by up to 23.9%, but it results in up to an 8.1% decrease in performance for OUT users. This is because CRIM determines the user scheduling based on the virtual queue, i.e., users with lower data rate (users in the OUT) has large virtual queue resulting in more service chances and GAT increment. In contrast, DQN just aims to maximize its reward, and the strategy for this results in more focus on users of the IN.



(a) Heterogeneous network configuration.



(b) Convergence performance.

FIGURE 7. Convergence performance for transmit power allocation.

To confirm this phenomenon in details, we conduct additional simulations on a smaller cell radius (i.e., 60m) environment. Here, the number of users are set to scale down proportionally to the decrement of BS radius for comparison. As illustrated in Fig. 6(a), we figure out that our DQN not only outperforms benchmark algorithms but also increases the performance gap between ours and CRIM in the smaller cell environment. Additionally, different from the previous observations, our DQN outperforms the other algorithms for the majority of users as shown in 6(b) and 6(c) regardless of distance. Furthermore, these results can also be confirmed in 6(d), where IN and OUT are separated by 50 m. This is due to the fact that in a smaller cell environment, inter-cell interference tends to be stronger for most users, leading to a more pronounced impact of efficiently designed algorithm, i.e., our DQN, on interference. Meanwhile, the PF-DQL demonstrates superiority in OUT but exhibits inferiority in IN for both simulations due to its strong emphasis on fairness.

3) IMPACT OF DIFFERENT REWARD FUNCTIONS

Next, we conduct additional simulations to verify the validity of the proposed judgement-based reward design. We consider a single-user and single-antenna scenario where the agent learns to maximize the sum data rate of all users through the transmit power allocation. Additionally, to reflect a realistic aspects of wireless network environment, we employ a heterogeneous network using PPP (Poisson Point Process) with seven BSs, as shown in Fig. 7(a). For comparison, we use an optimization-based algorithm, ULTIMA, and two DQN methods employing different reward designs, namely *Sum-rate* and *Distributed*. The reward design of Sum-rate is calculated for each BS n as the sum of data rate provided by the entire BSs, and the reward design of Distributed consists of two terms as expressed in Eq. (12). Here, REF and REF_n denotes the reference user and its associated BS, respectively.

$$c_{\text{REF}_n \setminus n}(t) = \log_2\left(1 + \frac{h_{\text{REF}_n,\text{REF}}(t) p_{\text{REF}_n}(t)}{\sum_{m \setminus n} h_{m,\text{REF}}(t) p_m(t) + \sigma^2}\right), \quad (11)$$

$$r_n(t) = c_n(t) - (c_{\text{REF}_n \setminus n}(t) - c_{\text{REF}_n}(t)), \quad (12)$$

where the first term represents the data rate provided by BS n, and the second term represents the loss in data rates of reference user caused by BS n, calculated as Eq. (11). The result is shown in Fig. 7(b). As depicted in Fig. 7(b), the proposed judgment-based reward design enables learning to achieve higher performance compared to existing reward designs.

An important point to note is the efficiency of the proposed method in heterogeneous networks and multi-agent learning environments. In heterogeneous networks, BSs generate highly diverse and intensive interference, leading to performance degradation for users. Therefore, each agent must learn according to its own interference environment, necessitating a sophisticated reward design to facilitate the benefits of learning. In this context, the reward design of Sum-rate achieves the lowest performance, since each agent receives rewards based on the actions of all agents. In contrast, both Distributed and proposed reward designs allow distributed learning tailored to each agent's interference environment thanks to the concept of reference user, leading to higher performance than the Sum-rate. Nevertheless, the proposed design outperforms the Distributed, because the judgement-based reward enables the agent to be judged by the correctness of its actions and by imposing a strict and stable penalty than Distributed. Moreover, due to the judgement term in the proposed design, the agent is compelled to determine better solutions than ULTIMA, i.e., to maximize positive reward and to minimize negative reward, thereby achieving higher performance compared to ULTIMA.

4) DISCUSSION

Now, we discuss the necessary feedback information and training duration for the proposed decision-making process.

First, as we apply an approximation of interference by introducing the reference user concept, each BS requires two types of feedback information: (i) intra-cell feedback from users within the same cell, and (ii) inter-cell feedback from neighboring BSs. For intra-cell feedback, CSI from all users within the cell must be relayed to the BS. However, for inter-cell feedback, each BS needs CSI information pertaining to the reference user in neighboring cells. To acquire this information, the BS must be aware of the scheduled users in neighboring cells and their CSIs at each time slot. To minimize inter-cell feedback, it is efficient to only correct the CSIs of edge users in neighboring cells, as these users, often experiencing high interference, are typically chosen as reference users.

Second, concerning the training duration of the proposed decision-making process, our simulation results indicate that performance saturation after 10,000 time slots, i.e., 10 seconds. Given that the proposed Deep Q-Network (DQN) solution consistently outperforms the optimization-based approach after this period for all simulation environment such as the number of users, the training duration is deemed reasonably adequate.

VII. CONCLUSION

In this paper, we leveraged DQL to efficiently solve the utility maximization problem with decision variables of beam pattern selection, user scheduling and transmit power allocation. In order to resolve the curse of dimensionality due to the high dimension of state-action combinations and enhance learning efficiency, we proposed a novel decision-making process which designs per-user state and action spaces and a judgement-based reward design. Finally, we not only showed the superiority of the proposed DQN but also figured out interesting points that give network operators the following insights: *i*) our proposed DQN can be an attractive framework for overall performance enhancement, especially in a smaller cell environment, and *ii*) the network operator can utilize our DQN and PF-DQN in complementary way for high throughput and high fairness among users in relatively large cell environment.

REFERENCES

[1] Y. Teng, M. Liu, F. R. Yu, V. C. M. Leung, M. Song, and Y. Zhang, "Resource allocation for ultra-dense networks: A survey, some research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2134–2168, 3rd Quart., 2019.

[2] R. Amiri, M. A. Almasi, J. G. Andrews, and H. Mehrpouyan, "Reinforcement learning for self organization and power control of two-tier heterogeneous networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 8, pp. 3933–3947, Aug. 2019.

[3] I. M. Braga, E. D. O. Cavalcante, G. Fodor, Y. C. B. Silva, C. F. M. E. Silva, and W. C. Freitas, "User scheduling based on multi-agent deep Q-learning for robust beamforming in multicell MISO systems," *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2809–2813, Dec. 2020.

[4] I. Ahmed, M. K. Shahid, and T. Faisal, "Deep reinforcement learning based beam selection for hybrid beamforming and user grouping in massive MIMO-NOMA system," *IEEE Access*, vol. 10, pp. 89519–89533, 2022.

[5] S. He, J. Yuan, Z. An, W. Huang, Y. Huang, and Y. Zhang, "Joint user scheduling and beamforming design for multiuser MISO downlink systems," *IEEE Trans. Wireless Commun.*, vol. 22, no. 5, pp. 2975–2988, May 2023.

[6] P. Yoon, J. Hong, S. Ahn, Y. Cho, J. Na, and J. Kwak, "ULTIMA: Ultimate balance of centralized and distributed benefits for interference management in 5G cellular networks," *IEEE Access*, vol. 11, pp. 85694–85710, 2023.
[7] K. Son, S. Lee, Y. Yi, and S. Chong, "REFIM: A practical interference management in heterogeneous wireless access networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 6, pp. 1260–1272, Jun. 2011.
[8] J. Hong, P. Yoon, S. Ahn, Y. Cho, J. Na, and J. Kwak, "Three steps toward low-complexity: Practical interference management in NOMA-based mmWave networks," *IEEE Access*, vol. 10, pp. 128366–128379, 2022.
[9] Z. Cheng, Z. Wei, and H. Yang, "Low-complexity joint user and beam selection for beamspace mmWave MIMO systems," *IEEE Commun. Lett.*, vol. 24, no. 9, pp. 2065–2069, Sep. 2020.
[10] H. Zhang, S. Huang, C. Jiang, K. Long, V. C. M. Leung, and H. V. Poor, "Energy efficient user association and power allocation in millimeter-wave-based ultra dense networks with energy harvesting base stations," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 9, pp. 1936–1947, Sep. 2017.
[11] W. Xia, G. Zheng, Y. Zhu, J. Zhang, J. Wang, and A. P. Petropulu, "A deep learning framework for optimization of MISO downlink beamforming," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1866–1880, Mar. 2020.
[12] J. Ge, Y.-C. Liang, J. Joung, and S. Sun, "Deep reinforcement learning for distributed dynamic MISO downlink-beamforming coordination," *IEEE Trans. Commun.*, vol. 68, no. 10, pp. 6070–6085, Oct. 2020.
[13] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for interference management," *IEEE Trans. Signal Process.*, vol. 66, no. 20, pp. 5438–5453, Oct. 2018.
[14] W. Zou, Z. Cui, B. Li, Z. Zhou, and Y. Hu, "Beamforming codebook design and performance evaluation for 60GHz wireless communication," in *Proc. ISCIT*, Hangzhou, China, Oct. 2011, pp. 30–35.
[15] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, Oct. 2000.
[16] L. Venturino, N. Prasad, and X. Wang, "Coordinated scheduling and power allocation in downlink multicell OFDMA networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 6, pp. 2835–2848, Jul. 2009.
[17] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with deep reinforcement learning," 2013, *arXiv:1312.5602*.
[18] A. L. Stolyar and H. Viswanathan, "Self-organizing dynamic fractional frequency reuse for best-effort traffic through distributed inter-cell coordination," in *Proc. IEEE INFOCOM*, Rio de Janeiro, Brazil, Apr. 2009, pp. 1287–1295.
[19] M. Dong, L. Tong, and B. M. Sadler, "Optimal insertion of pilot symbols for transmissions over time-varying flat fading channels," *IEEE Trans. Signal Process.*, vol. 52, no. 5, pp. 1403–1418, May 2004.
[20] *Study on Channel Model for Frequency Spectrum Above 6 GHz; (Release 14)*, document 3GPP TR 36.3900.
[21] *Study on Small Cell Enhancements for E-UTRA and E-UTRAN—Physical-Layer Aspects (Release 12)*, document 3GPP TR 36.872.

PILDO YOON (Member, IEEE) received the B.S. degree in computer engineering from Hanbat National University, Daejeon, South Korea, in 2022, and the master's degree from DGIST, Daegu, South Korea, in 2024. His research interests include practical interference management in heterogeneous cellular networks and resource optimization on cellular network via reinforcement learning.

YUNHEE CHO received the M.S. and Ph.D. degrees from the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2001 and 2014, respectively. Since 2001, she has been with the Electronics and Telecommunications Research Institute (ETRI), Daejeon, where she is currently a Principal Researcher. Her research interests include 4G, 5G small cells, self-organizing networks (SON), compact MIMO, beamforming, interference management of multi-cell OFDMA networks, quality of experience (QoE), and universal access of HetNet.

JEEHYEON NA (Member, IEEE) received the B.S. degree in computer science from Chonnam National University, and the M.S. and Ph.D. degrees in computer science from Chungnam National University, in 2002 and 2008, respectively. She has been with the Electronics and Telecommunications Research Institute (ETRI), since 1989, where she is currently the Director of the Intelligent Ultra Dense Small Cell Research Section. Her research interests include 4G, 5G small cells, self-organizing networks (SON), and location management and paging for mobile communication networks. She is a member of IEICE Communication Part.

JEONGHO KWAK (Member, IEEE) received the B.S. degree (summa cum laude) in electrical and computer engineering from Ajou University, Suwon, South Korea, in 2008, and the M.S. and Ph.D. degrees in electrical engineering from KAIST, Daejeon, South Korea, in 2011 and 2015, respectively. He is currently an Associate Professor with the Department of Electrical Engineering and Computer Science, DGIST, Daegu, South Korea. Prior to joining DGIST, he was with INRS-EMT, Montreal, Canada, and the Trinity College Dublin, Dublin, Ireland, as a Postdoctoral Researcher and a Marie Skłodowska-Curie Fellow, respectively. His current research interests include learning model and resource allocation in hybrid cloud/edge network architecture, energy optimization in heterogeneous networks, and radio resource management for 5G wireless cellular networks.