mathematics

MDPI

Article

DQN-GNN-Based User Association Approach for Wireless Networks

Ibtihal Alablani and Mohammed J. F. Alenazi *

Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh P.O. Box 11451, Saudi Arabia; 438203904@student.ksu.edu.sa
* Correspondence: mjalenazi@ksu.edu.sa

Abstract: In the realm of advanced mobile networks, such as the fifth generation (5G) and beyond, the increasing complexity and proliferation of devices and unique applications present a substantial challenge for User Association (UA) in wireless systems. The problem of UA in wireless networks is multifaceted and requires comprehensive exploration. This paper presents a pioneering approach to the issue, integrating a Deep Q-Network (DQN) with a Graph Neural Network (GNN) to enhance user-base station association in wireless networks. This novel approach surpasses recent methodologies, including Q-learning and max average techniques, in terms of average rewards, returns, and success rate. This superiority is attributed to its capacity to encapsulate intricate relationships and spatial dependencies among users and base stations in wireless systems. The proposed methodology achieves a success rate of 95.2%, outperforming other methodologies by a margin of up to 5.9%.

Keywords: Graph Neural Networks; Deep Q-Network; User Association; 5G; Machine Learning; Reinforcement Learning

MSC: 90B18; 94C15; 68T05; 68T20

Citation: Alablani, I.; Alenazi, M.J.F. DQN-GNN-Based User Association Approach for Wireless Networks. *Mathematics* **2023**, *11*, 4286. https://doi.org/10.3390/math11204286

Academic Editor: Lingfei Mo

Received: 21 August 2023
Revised: 9 October 2023
Accepted: 11 October 2023
Published: 14 October 2023

1. Introduction

The Internet of Things (IoT) represents a vast network of interconnected devices, objects, or "things" that communicate and exchange data with each other. These devices can connect and interact with the external environment [1]. The increase in wireless connectivity in recent years has resulted in extensive research in Resource Allocation (RA) for wireless networks. With Next Generation Wireless (NGW) networks needing to support more connected devices and resource-intensive applications, there is a renewed focus on developing more efficient User Association policies [2,3].

Figure 1 displays a network with multiple Base Stations (BSs) and a collection of IoT devices and cell phones that represent users. These devices need to be associated with the most suitable BS to optimize the system's utility function. The User Association problem involves formulating strategies for this association process, which becomes complex and challenging in high-traffic or congested environments. The figure illustrates the complexity of these associations, highlighting the need for advanced methods to achieve efficient and effective User Association. The primary objective is to optimize the system's utility function, commonly associated with throughput. In scenarios of low traffic, basic heuristic methods can yield satisfactory results. However, these simplistic methods fail to deliver optimal solutions in high-traffic or congested environments. Therefore, identifying the ultimate association policy represents a complex and challenging problem. Allocating and managing resources in wireless networks is a difficult and complex task that requires significant effort to achieve the desired outcomes [4]. This issue can be tackled by treating the UA problem as a sequence of decisions and applying Deep Reinforcement Learning (DRL) strategies to create more effective policies [5].

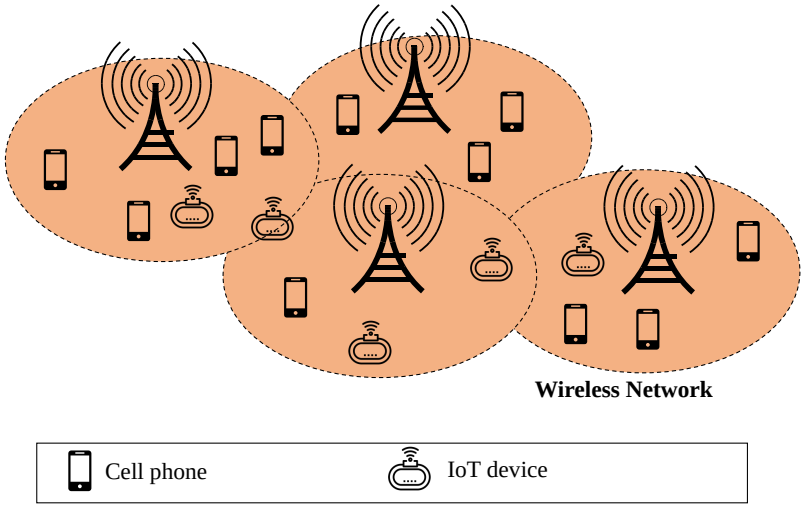

Figure 1. The cell selection issue in NGN.

Mathematics plays a fundamental role in machine learning by providing the theoretical foundations, algorithms, and tools necessary for understanding and analyzing complex data patterns [6]. Concepts from probability theory, optimization, and linear algebra are applied to formulate and solve the mathematical models underlying machine learning algorithms. Furthermore, mathematical analysis allows researchers to prove convergence properties, stability, and optimality guarantees for ML methods. Reinforcement Learning (RL) is a form of machine learning. It involves an agent that learns to make optimal decisions by interacting with its environment and getting feedback as rewards or penalties. The agent aims to master a policy, which is a set of rules directing states to actions for the maximal cumulative reward over time. This learning technique is iterative, with the agent continually refining its actions based on past experiences to better future outcomes. It is a versatile tool for solving complex issues [7]. In recent years, RL research has been focusing on applying its concepts to real-world problems. To emulate human learning, RL uses designs based on trial and error [8,9]. Deep Reinforcement Learning is an advanced variant of Reinforcement Learning that employs a Deep Q-Network to compute the Q value-action function [10]. By leveraging neural networks, it approximates the Q-function in Q-learning, which allows it to proficiently manage high-dimensional state spaces [11]. This approach has found applications in wireless networks, where it has been used to significantly boost their performance [12]. DRL has demonstrated its potential in dealing with complex problems by navigating large state spaces and improving decision-making processes.

A Graph Neural Network is a neural network variant that is specifically engineered to handle graph data structures, making it capable of understanding and inferring intricate relationships among various entities [13]. Within a GNN, each graph node is linked with a feature vector that encapsulates the node's characteristics. These vectors are processed through multiple layers of the neural network, each layer modifying the vectors according to the attributes of the neighboring nodes in the graph [14]. This mechanism empowers GNNs to gather and disseminate information across the graph, thereby facilitating reasoning about inter-entity relationships. The ultimate output of the GNN can serve various purposes, such as node categorization, link forecasting, and graph-oriented classification [15].

Combining DRL and GNN can enable the development of more advanced intelligent systems that can reason and act in complex environments with graph-structured data [16]. By using GNNs to encode and process graph-structured data, DRL agents can better understand the relationships between entities in their environment, allowing them to make more informed decisions and take actions that lead to better performance [17]. Furthermore,

the combination of DRL and GNN can enhance the ability of these intelligent systems to make generalizations, enabling them to perform well in unseen or partially observed environments with similar graph structures [18].

The main contribution of our work is to develop a deep reinforcement learning framework for the user association problem using graph representations. It is a DQN-GNN-based approach designed for wireless networks to effectively associate wireless users with the network.

In this paper, we present the following key contributions:

1. A novel DQN with a GNN-based approach is proposed for efficient user-to-base station association in wireless networks.
2. A comprehensive evaluation of our proposed method is conducted in terms of (a) average rewards, (b) average returns, and (c) success rate.
3. The DQN-GNN approach outperforms current recent work, such as Q-learning and max average approaches, achieving a success rate of 95.2%, which is higher than other methods by up to 5.9%.
4. The combination of DQN and GNN enhances performance by capturing complex relationships and spatial dependencies in wireless networks, leading to more accurate and efficient associations.

The remainder of this paper is structured as follows: Section 2 provides a mathematical perspective of the DQN-GNN approach. A comprehensive examination of related studies is offered in Section 3. A detailed explanation of the proposed method and the system model can be found in Section 4. The mathematical formulation and optimization of our proposed DQN-GNN approach are discussed in Section 5. The effectiveness of the proposed user association strategy is assessed in Section 6. The qualitative analysis and comparison are given in Section 7. Finally, the paper wraps up in Section 8, with conclusions and potential future work.

2. DQN and GNN: A Mathematical Perspective

In this section, we provide a mathematical perspective of the Deep Q-Network algorithm and explore its connection to Graph Neural Networks. DQN is a reinforcement learning algorithm that combines deep neural networks with the Q-learning algorithm to solve complex decision-making problems [19]. The core idea behind DQN is to approximate the Q-value function using a deep neural network and iteratively update the network parameters to improve the Q-value estimates [20].

Let us define the Q-value function for a given state-action pair as represented in Equation (1).

$$Q(s,a) = \mathbb{E}\left[r_t + \gamma \max_{a'} Q(s',a') \mid s,a\right], \tag{1}$$

where s represents the current state, a represents the action taken in that state, r_t denotes the immediate reward received after taking action a in state s at time t, s' represents the next state, and γ is the discount factor that determines the importance of future rewards.

The goal of DQN is to learn an optimal Q-value function $Q^*(s,a)$ that maximizes the expected cumulative reward. To achieve this, DQN utilizes a deep neural network parameterized by θ to approximate the Q-value function. Let $Q(s,a;\theta)$ represent the output of the neural network when the state-action pair (s,a) is passed through the network with parameters θ [21].

The DQN algorithm uses a loss function to measure the discrepancy between the predicted Q-values and the target Q-values. The target Q-value for a state-action pair (s,a) is given by Equation (2).

$$\hat{Q}(s,a) = r + \gamma \max_{a'} Q(s',a';\theta^-), \tag{2}$$

where r is the immediate reward obtained after taking action a in state s,s' is the next state, and θ^- represents the parameters of a separate target network that are updated less frequently than the online network. The loss function used in DQN is the mean squared error (MSE) between the predicted Q-values and the target Q-values [22], given in Equation (3).

$$L(\theta) = \mathbb{E}\Big[(\hat{Q}(s,a) - Q(s,a;\theta))^2\Big]. \tag{3}$$

To update the parameters of the neural network, DQN employs gradient descent to minimize the loss function. The weights θ are updated according to Equation (4), where α is the learning rate.

$$\theta \leftarrow \theta - \alpha \nabla_\theta L(\theta), \tag{4}$$

By iteratively applying the Q-learning updates and optimizing the neural network parameters, DQN learns an optimal policy that maximizes the expected cumulative reward. GNNs are a class of neural networks designed specifically to operate on graph-structured data. They have gained significant attention due to their ability to capture complex relationships and dependencies within graphs.

Let us consider a graph $G = (V, E)$, where V represents the set of nodes and E represents the set of edges connecting the nodes. Each node v_i in the graph is associated with a feature vector $x_i \in \mathbb{R}^d$, which represents the input features of the node. Additionally, each edge (v_i, v_j) can have an associated edge attribute $e_{ij} \in \mathbb{R}^p$, which represents the characteristics of the edge.

The goal of GNNs is to learn a node-level or graph-level representation that captures the structural information and the interactions between nodes and edges in the graph. GNNs achieve this by iteratively aggregating information from neighboring nodes and updating the node representations. The propagation rule of a GNN can be expressed using Equation (5).

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i)} \frac{1}{c_{ij}} W^{(l)} h_j^{(l)} + U^{(l)} h_i^{(l)}\right), \tag{5}$$

where $h_i^{(l)}$ represents the hidden representation of node v_i at layer l, $\mathcal{N}(i)$ represents the set of neighbors of node v_i, $W^{(l)}$ and $U^{(l)}$ are learnable weight matrices at layer l, and c_{ij} is a normalization factor that depends on the degree of node v_i and v_j. The function $\sigma(\cdot)$ represents a non-linear activation function, such as rectified linear unit (ReLU) or sigmoid.

By stacking multiple layers of GNNs, the model can capture increasingly complex graph patterns and dependencies. The final node representations can be used for various downstream tasks, such as node classification, link prediction, or graph classification.

3. Related Work

In this section, related studies that used reinforcement learning to solve the user association problem in wireless networks are discussed, along with their limitations.

In [23], Li, Z. et al. presented a distributed user association algorithm known as Online Reinforcement Learning Approach (ORLA). This innovative approach utilizes online RL to optimize load balancing within vehicular networks. ORLA leverages historical association experiences to adapt to dynamic vehicular environments and achieve superior association solutions. It also effectively handles network dynamics through real-time feedback and consistent traffic association patterns. When tested with the QiangSheng taxi movement dataset, comprising genuine taxi movement data from Beijing, ORLA outperformed other prevalent association methodologies in terms of load-balancing quality.

In their study, Li, Q. et al. built an intelligent user association algorithm, named reinforcement learning handoff (RLH), intended to minimize unnecessary handoffs in UAV

networks [24]. Furthermore, they devised two distinct UAV mobility control strategies to work in tandem with the RLH algorithm to improve system throughput. The RLH algorithm motivates user handoffs through rewards obtained from the reinforcement learning process. The two suggested UAV mobility control strategies are based on the estimation of the SNR and the K-Means method. As demonstrated by simulation results, the RLH algorithm can effectively lower handoffs by as much as 75%, thus proving its efficacy in diminishing unneeded handoffs.

In their research, Zhao et al. [2] introduced a reinforcement learning strategy to maximize network utility and maintain quality of service in diverse cellular networks. This method employs a dueling double deep Q-network (D3QN) strategy within a multi-agent reinforcement learning framework to tackle the issue of expansive action spaces. The distributed user equipment (UEs) gains access to the global state space via message passing, enabling the D3QN approach to quickly reach a subgame perfect Nash equilibrium. Simulations reveal that the D3QN surpasses other reinforcement learning methods in addressing large-scale learning challenges.

Ding et al. in [25] presented a multi-agent deep Q-learning network methodology to simultaneously enhance user association and power regulation in uplink heterogeneous networks (HetNets) utilizing orthogonal frequency division multiple access (OFDMA). They successfully tackled the non-convex and non-linear issue using this multi-agent DQN approach, which demands less environmental communication information compared to conventional methods such as game theory, fractional programming, and convex optimization. This proposed technique ensures maximum long-term overall network utility through a novel reward function while also maintaining the quality of service (QoS) for users. Simulations have shown the multi-agent DQN approach to surpass traditional Q-learning in terms of energy efficiency and convergence.

Chou et al. introduced an algorithm rooted in the Deep Deterministic Policy Gradient (DDPG) in [26] as a solution to the user association issue in wireless networks. They redefined the problem as a Markov Decision Process (MDP) and employed their proposed algorithm to take advantage of the supply-demand understanding of the Lagrange dual problem. The simulated outcomes reveal that their approach notably enhances the quality of experience (QoE), especially in situations with limited wireless resources and a high user count, in comparison to other baseline methods.

Guan et al. introduced a solution for dynamically optimizing user association and power allocation in each time slot in [27] to minimize the long-term average transmission power consumed by users. Such a joint problem can be formulated as a Markov decision process, which suffers from the curse of dimensionality when there are many users. The authors proposed a distributed relative value iteration (RVI) algorithm that reduces the dimensions of the MDP problem, enabling it to be broken down into multiple solvable small-scale MDP problems. Simulation results indicate that the proposed algorithm performs better than both the conventional RVI algorithm and a baseline algorithm with myopic policies in terms of long-term average transmission power consumption.

In [28], Zhang et al. developed two deep reinforcement learning algorithms for Internet of Things device association in wireless networks. The centralized DRL algorithm makes decisions for all devices simultaneously, using global information, while the distributed DRL algorithm makes decisions for one device at a time, using local information. Both algorithms use historical information to infer current information and achieve performance comparable to optimal user association policies that require real-time information. The distributed DRL algorithm is shown to have the advantage of scalability in simulations.

Sana et al. proposed a multi-agent reinforcement learning approach in [29] to address the issue of user association in wireless networks. Users act as independent agents and learn to coordinate their actions based on their local observations to optimize the network sum-rate. The proposed approach limits signaling overhead since there is no direct information exchange among the agents. Simulation results show that the algorithm is scalable, flexible,

and able to adapt to rapid changes in the radio environment, resulting in a large sum-rate gain compared to state-of-the-art solutions.

Dinh et al. in [30] investigated the problem of optimizing joint user-to-access points (AP) association and beamforming in an integrated sub-6GHz/mmWave system to maximize the system's long-term throughput while meeting various user quality-of-service requirements. The proposed method is based on Deep Q-Networks, where each user optimizes its AP association and interface requests and can be served by multiple APs simultaneously to support multiple applications. Each AP selects its associated users and applications served on each interface while optimizing its millimeter wave (mmWave) beamforming parameters. Simulation results demonstrate that the proposed method outperforms baseline DQN schemes, achieving high global throughput and reducing user outage probabilities.

In [31], Hsieh et al. propose a novel approach for user association in heterogeneous networks that directly operates in the hybrid space, using a parameterized deep Q-network (P-DQN) to maximize the average cumulative reward while considering constraints on wireless backhaul capacity and quality-of-service for each user device. The proposed P-DQN outperforms traditional approaches such as DQN and distance-based association in terms of energy efficiency while satisfying the QoS and backhaul capacity constraints. Simulation results show that in a HetNet with three small base stations (SBS) and five UEs, the proposed P-DQN improves energy efficiency by up to 77.6% and 140.6% compared to DQN and distance-based association, respectively.

In [32], Chen et al. proposed a decentralized method to adjust the flight paths of multiple Unmanned Aerial Vehicles (UAVs) over diverse Ground Users (GUs). Their aim was to maximize total data transfer and energy efficiency while maintaining fairness. They redefined the problem as a Decentralized Partially Observed Markov Decision Process (DEC-POMDP) and tackled it using a Coalition Formation Game (CFG) and Multi-Agent Deep Reinforcement Learning (MADRL). The CFG algorithm was employed to acquire a decentralized solution that converges to the Nash equilibrium. Subsequently, a MADRL-based procedure was utilized to perpetually optimize the UAVs' trajectories and energy usage in a centralized-training yet decentralized-execution manner. Simulations showed that their proposed method surpassed current methods in terms of fairness in data transfer and energy usage in a distributed way.

Joda et al. proposed strategies for placing network functions in cloud nodes and associating users with radio units (RUs) in [33] to minimize end-to-end delay and deployment cost in Open Radio Access Networks (O-RANs). The problem is formulated as a multi-objective optimization problem with a large number of constraints and variables. To solve the problem, a Markov Decision Problem was developed, and a DQN-based algorithm was proposed. The simulation results show that the proposed scheme reduces the average user delay by up to 40% and the deployment cost by up to 20% compared to the baselines.

Alizadeh and Vu in [34] developed a central load balancer designed to maintain equal distribution across all base stations at every stage of learning. Their proposed methodology introduces two different association vectors, allowing users to participate in background learning while simultaneously engaging in optimal data transmission. They also presented a measurement model designed to swiftly account for channel fluctuations and user mobility within dynamic networks. To minimize the rate of handover, they distinguish between the costs of transmission handover and learning handover, introducing a learning handover cost that decreases with the duration of stay. Simulation results reveal that the proposed algorithms not only converge quickly but also surpass the Third Generation Partnership Project (3GPP) handover, approaching near-optimal benchmarks for worst connection swapping.

In [35], Khoshkbari et al. proposed a novel deep Q-learning approach in which a satellite serves as an operative agent. This agent is responsible for scheduling each user to a terrestrial base station (TBS) or a high-altitude platform station (HAPS) within each time slot, utilizing channel state information (CSI) obtained from the preceding time slot.

This proposed approach yields results that almost mirror those achieved via the exhaustive search action selection method. Moreover, it outperforms a convex optimization-based user association scheme in scenarios where the CSI is noisy. The researchers further delve into the implications of imperfect CSI and highlight the superior performance of their proposed method under such circumstances.

Moon et al. proposed a decentralized user association technique based on a multi-agent actor-critic (AC) method in [36] to maximize the energy efficiency of an ultra-dense network (UDN). This technique aims to enhance the energy efficiency of ultra-dense networks. The actor network of the proposed technique decides the user association of the small base station, guided by local observations. Simultaneously, the critic network communicates the energy-efficient user association decision to the actor network. This mechanism allows each small base station's deep reinforcement learning agent to identify the user association decision, which optimizes network energy efficiency. According to the simulation results, the proposed method provides an average improvement in energy efficiency of more than 50% compared to traditional user association techniques.

In [37], Kim et al. proposed a curriculum learning technique to improve the accuracy of a reinforcement learning agent in solving the challenging problem of resource allocation in wireless networks, particularly in scenarios with high user mobility, such as the Internet of Vehicles (IoV). The proposed technique involves gradually increasing the mobility of each user during learning to enhance the model's accuracy. Simulation results demonstrate that the proposed method achieves faster convergence and better performance compared to traditional reinforcement learning techniques.

Table 1 displays a comparison among recent related works in terms of the used RL methods and main goals.

Table 1. Comparison among recent related works.

Cite	Authors	Year	RL Method	Main Goal
[23]	Li, Z. et al.	2017	Online historical-based RL	Balancing load in vehicular networks.
[24]	Li, Q. et al.	2018	Multi-agent RL	Reducing redundant handoffs in UAV networks.
[38]	Zhao et al.	2019	Multi-agent RL with D3QN	Optimizing network utility in heterogeneous cellular networks.
[25]	Ding et al.	2020	Multi-agent DQN	Jointly optimizing user association and power control in HetNets.
[26]	Chou et al.	2020	DDPG	Reformulating user association as MDP for QoE improvement.
[27]	Guan et al.	2020	Distributed RVI	Jointly optimizing user association and power allocation with reduced dimensionality.
[28]	Zhang et al.	2020	DRL	Associating IoT devices in wireless networks intelligently.
[29]	Sana et al.	2020	Multi-agent RL	Coordinating independent agents to optimize network sum-rate.
[30]	Dinh et al.	2021	DQN	Jointly optimizing user-AP association and beamforming in integrated sub-6GHz/mmWave system.
[31]	Hsieh et al.	2021	Parameterized DQN	Associating users in heterogeneous networks with wireless backhaul constraints.
[32]	Chen et al.	2022	MADRL	Decentralizing optimization of UAV trajectories over heterogeneous GUs.
[33]	Joda et al.	2022	DQN	Placing network functions and associating users in O-RAN networks.

Table 1. *Cont.*

Cite	Authors	Year	RL Method	Main Goal
[34]	Alizadeh and Vu	2022	MAB	Learning handover costs for reducing handover rate in dynamic networks networks.
[35]	Khoshkbari et al.	2023	DQN	Scheduling users in satellite networks with noisy CSI.
[36]	Moon et al.	2023	Multi-agent AC	Maximizing energy efficiency in user association for UDNs.
[37]	Kim et al.	2023	Curriculum learning	Allocating resources in wireless networks with high user mobility.

The limitations of the recent related reinforcement learning-based user association schemes are:

- Limited scalability: Some of the related works face scalability issues when applied to large-scale wireless networks due to the high computational complexity and communication overhead of the proposed algorithms.
- Limited generalization: Many of the related works are not generalized well to different wireless network scenarios, as the performance relies heavily on the quality and quantity of training data or may only address specific problems, such as load balancing or power allocation.
- Limited flexibility: Some of the related works lack flexibility in adapting to rapid changes in the radio environment or may require high signaling overhead due to the direct information exchange among agents or nodes.
- Limited efficiency: Some of the related works suffer from inefficient resource allocation or may not fully utilize available resources, leading to suboptimal performance in terms of energy efficiency or network throughput.

4. The Proposed Approach

4.1. Building the Proposed DQN-GNN Approach

In this section, we describe the details of building our proposed DQN-GNN approach for user association in wireless networks. The design of the GNN and DQN models, the training process, and the integration of the two models for user association are discussed.

The proposed model building phase goes through five phases, as shown in Figure 2.

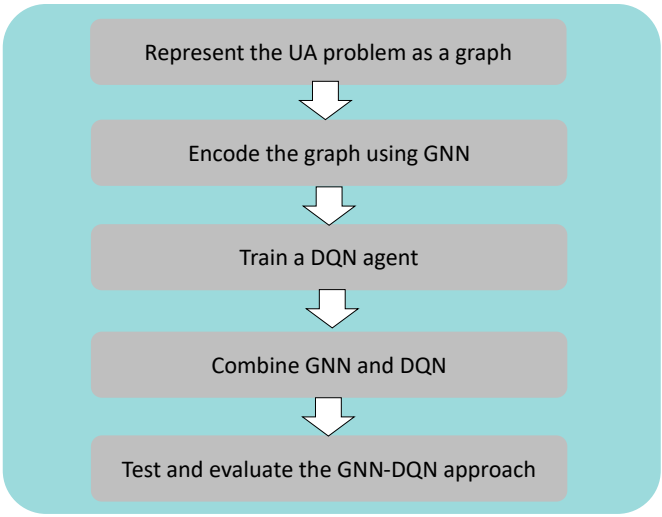

Figure 2. The building phases of the proposed approach.

1. Represent the UA problem as a graph: The user association problem is represented as a graph, where each node corresponds to a user or a base station, and the edges represent the wireless connections between them.
2. Encode the graph using GNN: A GNN is used to encode the graph structure of the network and learn a representation for each node that captures its importance and connectivity within the network. This allows the system to reason about the relationships between users and BSs and make more informed decisions about user association. In this work, we use a GNN model called LocalGNN for both the policy network and the target network, which was proposed by Shaddad et al. in [39]. A notable feature of the LocalGNN is its ability to locally aggregate information for each node, and this aggregation is also extended to include neighboring nodes within K hops. This means that the feature extraction and computation for each node can be performed locally, which greatly supports the scalability of the proposed algorithm.
3. Train a DQN agent: A DQN agent is trained to learn a policy that selects the best BS for each user to connect to based on the current state of the network. The state can be defined as the current set of user-BS associations, as well as additional network parameters such as signal strength, traffic load, and interference levels. The DQN agent can learn to optimize the network performance by selecting the best user-BS associations.
4. Combine GNN and DQN: The GNN and DQN models are combined by using the GNN to encode the graph structure of the network and provide input to the DQN agent. The DQN agent can then use the learned representation of each node to make more informed decisions about user association. This combination of models can lead to more efficient and effective user-BS associations and better overall network performance.
5. Test and evaluate the GNN-DQN approach: Once the models are trained, the system can be tested and evaluated using wireless network data. To assess the effectiveness of our proposed GNN-DQN approach for user association, we compared its results to those obtained using other user association methods. This comparison allows us to evaluate the performance of our approach relative to existing methods and determine its efficacy in solving the user association problem in wireless networks.

Figure 3 shows the diagram of the proposed DQN-GNN-based approach. There are two entities: intelligent DQN-GNN agent and 5G network environment. The DQN and GNN work together to provide an adaptive and efficient approach to User Association in 5G networks, with the DQN making real-time allocation decisions based on the current network state and the GNN providing predictions of future network states to improve the DQN's decision-making ability.

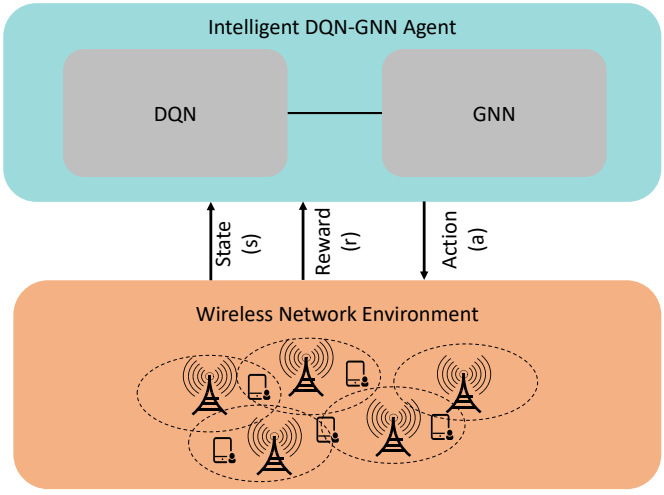

Figure 3. The diagram of the proposed DQN-GNN-based approach.

4.2. The System Model

Suppose we have N base stations in a 5G network environment, each of which possesses a finite number of frequency resources. According to the 5G framework, these resources are called Resource Blocks (RB). During each time interval t, there is a chance that a user may arrive, with the arrival probability following a certain distribution. If a user does arrive, one of the k base stations with the highest signal-to-noise-and-interference ratio (SNIR) and available resources is required to form an association with the new user. In our proposed reinforcement learning model, a tuple is formed by (s, a, T, r, γ).

- The current state in the agent-environment system, denoted as s, is a fusion of the condition of the base station and the user situation, according to [40]. The state of the base station encapsulates the number of linked users and the average utility achieved thus far, along with the existing state of the system and the features of the new user. The user's state, on the other hand, includes the Received Signal Strength Indicator (RSSI) from the associated base stations and a specific demand that must be met.
- The action that the agent takes in a particular state, symbolized as a, involves choosing a base station from the options available, according to [41]. However, an action is not necessary at every time interval. To ensure a well-defined Markov process, decision-making is incorporated into the state. If no user is present, the demand drops to zero, and the only task required is to update the system state without any action needed.
- T denotes the succeeding state in which the environment transitions after the agent executes a specific action, as described in [42]. The descriptors of the base station are updated to reflect the effects of the action, which could be an increase in the number of connected users and a new mean utility, as well as the impact of time, which could be a decrease in the number of connected users if a user's demand has been satisfied. The characteristics of the new user are revised every time a new user joins. It is important to note that while the transitions over the base station's features are deterministic, given the action a and the state s, they are stochastic for the new user's features.
- The reward that the agent earns for executing a particular action in a specific state is represented as r, as per [43]. In the context of our research, the reward is the logarithm of the sum of the throughput between users, which supports equity in resource allocation and is commonly used in related literature.
- The parameter γ signifies the discount factor utilized to prioritize future rewards in the agent's decision-making process, as per [44]. The agent aims to optimize the expected discounted cumulative reward by updating a policy (π) using one of Bellman's equations. The action-value function for policy π is utilized, and the state-action value function is refreshed using the optimality equation.

Table 2 displays the parameters of the proposed DQN-GNN model.

Algorithm 1 provides the complete pseudocode for the proposed DQN-GNN user association approach for wireless networks, where θ and ϕ are the parameters of the Q-network and GNN, respectively. The replay memory buffer is represented by D, which contains the experiences of the agent. $pi_\epsilon(a|s)$ is the epsilon-greedy policy used by the agent to select actions in the current state. s' is the next state, and *done* indicates whether the episode has terminated. The learning rate used to update the Q-network and GNN parameters is represented by α. The target Q-value (Q_{target}) is used to update the Q-network parameters. L is the loss function used to compute the difference between the predicted and target Q-values. $\nabla_\theta L$ is the gradient of the loss function with respect to the Q-network parameters. $\nabla_\phi Q_{\text{GNN}}(s_i, a_i; \phi, \theta)$ is the gradient of the GNN output with respect to the GNN parameters, while $\nabla_\theta Q(s_i, a_i; \theta)$ is the gradient of the Q-network output with respect to the Q-network parameters. The algorithm iteratively updates the Q-network and GNN parameters based on the observed transitions and their corresponding target Q-values, with the goal of maximizing the cumulative reward over a sequence of time steps. By learning the optimal user association policy, this approach can improve user experience and network efficiency in wireless networks.

Table 2. The parameters of the proposed DQN-GNN model.

Parameters	Values
LEARNING_RATE	1×10^{-3}
BATCH_SIZE	30
GAMMA	0.5
EPS_START	0.95
EPS_END	0.05
EPS_DECAY	1×10^{4}
TARGET_UPDATE	5000
UPDATE_FREQUENCY	10
DIMENSION_OF_NODE_SIGNALS	[4,1]
NUMBER_OF_FILTER_TAPS	2
ACTIVATION_FUNCTION	ReLU
OPTIMIZATION_ALGORITHM	Adam

Algorithm 1: DQN-GNN User Association Approach

Initialize θ, ϕ, D
for *each episode* **do**
  Initialize s;
  for *each* t **do**
    Select a based on $pi_{\epsilon}(a|s)$ policy;
    Execute a and observe s', r, and *done*;
    Store transition $(s, a, r, s', done)$ in D;
    Sample mini-batch of transitions $(s_i, a_i, r_i, s'_i, done_i)$ from D;
    Compute $Q_{target} = r_i + \gamma \max_{a'} Q_{GNN}(s'_i, a'; \phi; \theta)$;
    Compute $L = (Q_{target} - Q(s_i, a_i; \theta))^2$;
    Update $\theta \leftarrow \theta$ - $\alpha \nabla_{\theta} L$;
    Update $\phi \leftarrow \phi$ - $\alpha \nabla_{\phi} Q_{\mathrm{GNN}}(s_i, a_i; \phi, \theta) \cdot \nabla_{\theta} Q(s_i, a_i; \theta)$;
    Update $s \leftarrow s'$;
  end
end

The data in the environment are generated dynamically based on the specified parameters and actions. To store the agent's experiences, a replay memory buffer called D is utilized. This buffer contains transitions in the form of $(s, a, r, s', done)$, representing the state, action, reward, next state, and termination information at each time step during an episode. During the training phase, the algorithm selects a mini-batch of transitions $(s_i, a_i, r_i, s'_i, done_i)$ from the replay memory buffer D. These transitions are then used to compute the target Q-values and update the Q-network and GNN parameters. In the testing phase, the algorithm does not rely on a specific dataset or replay memory buffer. Instead, it assesses the performance of the trained policy by executing it within the environment and evaluating the achieved outcomes based on predefined metrics.

5. Mathematical Formulation and Optimization

In this section, we delve into the mathematical aspects of our proposed DQN-GNN approach for user association in wireless networks. The formulation provides the mathematical groundwork for the DQN-GNN model, setting up the problem in a way that allows

for the application of reinforcement learning techniques. Then, we discuss the optimization process of the DQN-GNN model.

5.1. Problem Formulation

The User Association problem can be formulated as a Markov Decision Process with the state space S, action space A, transition probability P, and reward function R. In our problem context:

- The state $s_t \in S$ at time t is defined by the current user-BS associations and network conditions, such as the number of connected users and the average utility achieved up to time t.
- The action $a_t \in A$ at time t is the decision made by the DQN-GNN agent to associate a user with a specific BS.
- The transition probability $P(s_{t+1}|s_t, a_t)$ is determined by the dynamics of the wireless network, such as the arrival and departure of users and changes in network conditions.
- The reward $R(s_t, a_t)$ is the utility of the system after taking action a_t in state s_t, which is defined as the logarithm of the sum of the throughput between users.

The goal of the DQN-GNN agent is to learn a policy π that maximizes the expected cumulative discounted reward $E[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)]$, where $\gamma \in [0, 1]$ is the discount factor.

Our DQN-GNN approach is particularly relevant in the context of wireless networks and the Internet of Things. In wireless IoT networks, a large number of interconnected devices are continuously interacting, leading to dynamic and complex network conditions. The state space (S) in our approach, which represents the current user-BS associations and network conditions, can effectively capture the complex dynamics of such networks. The action space (A), which represents the decision to associate a user with a specific BS, allows for efficient resource allocation in these networks, where resources are often scarce and need to be judiciously allocated. The transition probability (P) and the reward function (R) can model the impact of these decisions on the network's performance, providing a way to navigate the complex and dynamic IoT environment.

Moreover, the policy learned by the DQN-GNN agent, which aims to maximize the expected cumulative discounted reward, can guide the decision-making process in these networks. This policy essentially provides a strategy for efficient user association in the face of dynamic network conditions and limited resources, which are typical characteristics of wireless IoT networks. Therefore, our DQN-GNN approach provides a mathematical framework for optimizing user association in wireless IoT networks and can significantly enhance network performance in these environments.

5.2. Optimization Process

The optimization process is a crucial component of our proposed DQN-GNN approach. It refers to the iterative method of adjusting the parameters of the DQN and GNN models to minimize the difference between the predicted Q-values and the actual rewards. This process is key to improving the model's performance in the user association task over time.

This process can be visualized in the context of the Markov Decision Process, as shown in Figure 4. The diagram illustrates the interaction between the DQN-GNN agent and the wireless network environment over multiple time steps. Starting from an initial state (s), the agent takes an action (a), which leads to a new state (s′) and a reward (r). This cycle repeats as the agent continues to interact with the environment, with the agent taking another action (a′) in state (s′), leading to another new state (s″) and another reward (r′).

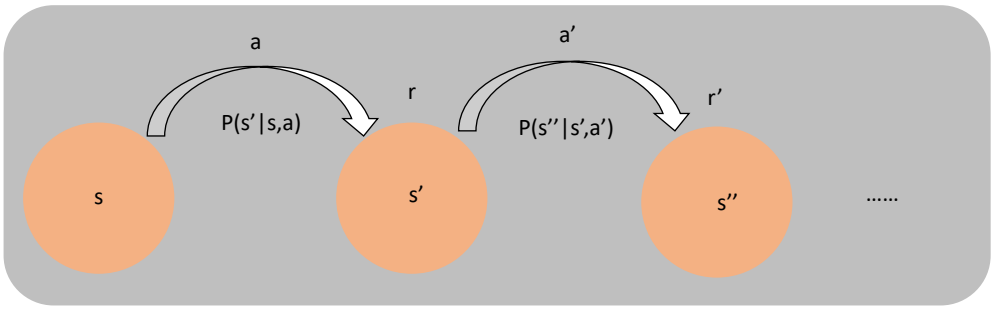

Figure 4. Illustration of the Markov Decision Process in the DQN-GNN approach.

Formally, the optimization process of our DQN-GNN model can be mathematically formulated as follows:

$$\theta^*, \varphi^* = \arg\min_{\theta,\varphi} \mathbb{E}_{(s,a,r,s')\sim D}[(Q_\theta(s,a) - (r + \gamma \max_{a'} Q_\varphi(s',a')))^2] \quad (6)$$

In this equation, θ, and φ are the parameters of the policy network (DQN) and the target network (GNN), respectively. The policy network is responsible for making the decisions (i.e., choosing the actions), while the target network is used to generate the target Q-values for the update of the policy network. The replay memory buffer (D) stores the agent's experiences in the form of state-action-reward-next state tuples. These experiences are sampled during the training process to update the model parameters. $Q_\theta(s,a)$ and $Q_\varphi(s',a')$ are the estimated Q-values of the current and next state-action pairs, respectively. These are the outputs of the DQN, which estimates the maximum expected future rewards for taking action a in state s.

The objective of the optimization process is to find the optimal parameters θ^* and φ^* that minimize the expectation of the squared difference between the predicted Q-values and the actual rewards (plus the discounted maximum Q-value of the next state). This difference represents the temporal difference error, which measures the discrepancy between the current Q-value estimate and the more accurate estimate obtained after observing the reward and the next state. Optimization is performed using an optimizer, which iteratively adjusts the parameters in the direction that reduces the error. Through this optimization process, the DQN-GNN model learns to make more accurate predictions and better decisions, leading to improved performance in the user association task.

6. Performance Evaluation

In this section, we present the results of our experiments using the proposed DQN-GNN model. We utilized Google Colab, a cloud-based Jupyter Notebook environment, to implement and evaluate our proposed DQN-GNN-based user association approach for wireless networks.

6.1. Performance Metrics

To evaluate the effectiveness of our proposed DQN-GNN model, we have used several key performance indicators (KPIs); namely, average rewards, average returns, and the success rate of agents.

- Average Rewards: The average rewards metric measures the average amount of reward that agents receive over a specific period of time. It is calculated as follows:

$$Average\ Rewards = \frac{1}{m}\sum_{i=1}^{m} r_i, \quad (7)$$

where m is the total number of episodes, and r_i is the reward obtained by the agent in the i^{th} episode.

- Average Returns: The average returns metric measures the average sum of discounted rewards that agents receive over a specific period of time. It is calculated as follows:

$$Average\ Returns = \frac{1}{m}\sum_{i=1}^{m}\sum_{t=0}^{S}\gamma^{t}r_{i,t}, \tag{8}$$

where S is the total number of time steps in each episode, $r_{i,t}$ is the reward obtained by the agent at time step t in the i^{th} episode and γ is the discount factor.

- Success Rate of Agents: The success rate of agents metric measures the percentage of successful episodes in which the agents achieve the desired goal. Specifically, this metric measures the percentage of episodes in which the agent achieved an average return that meets or exceeds a success threshold. The success threshold is set to 3.5; this value was chosen based on the difficulty of the task and the performance of the baseline models. The success rate of agents is calculated using Equation (9).

$$A_{SR} = \frac{Num_{EpAR} \geq ST}{Total_{Ep}} \times 100 \tag{9}$$

where A_{SR} represents the success rate of agents. In this context, it is the ratio of the number of successful episodes to the total number of episodes, expressed as a percentage. Num_{EpAR} represents the number of episodes in which the average return is greater than or equal to a success threshold (ST). The success threshold is a predetermined value that the average return of episodes should meet or exceed for the episode to be considered successful. The total count of episodes is represented by $Total_{Ep}$.

6.2. Performance Results

Figure 5 displays a line graph that compares the performance of three algorithms: (a) Baseline Max Arg, (b) Q-learning, and (c) the proposed DQN-GNN approach. The x-axis represents the averaged episodes, while the y-axis represents the average rewards. The term 'averaged episodes' refers to a method of computing the average rewards obtained by an agent in the simulation. This method involves computing a rolling average of the rewards earned by the agent over a fixed number of iterations, where each iteration corresponds to a predetermined number of episodes. The blue line represents the average rewards achieved by the DQN-GNN algorithm for each episode on the x-axis, while the red and green lines represent the average rewards achieved by the Q-learning algorithm and the average maximum, respectively. As the figure shows, the mean of the average rewards achieved by the proposed model is 2.357, which is approximately 48.34% higher than the minimum reward of 2.104. In contrast, the average max approach attains a mean of average rewards of 2.228, reflecting a percentage difference of 41.95% compared to its minimum reward of 2.031. Similarly, the Q-learning approach obtains a mean of average rewards of 2.223, representing a percentage difference of 41.84% compared to its minimum reward of 2.051. The DQN-GNN algorithm achieved the highest average rewards, followed by the average max and then the Q-learning algorithm. The proposed algorithm leverages the power of DQN and GNN to capture complex patterns and dependencies, leading to improved decision-making and higher average rewards. The average max approach, although not as effective as the DQN-GNN algorithm, still outperforms Q-learning. This is because the average max approach considers the maximum reward obtained in each episode, providing a more optimistic estimate of the agent's performance. It benefits from occasional high rewards, which can outweigh the impact of low rewards and result in a higher average. In contrast, Q-learning relies on a tabular representation of the action-value function and suffers from slow convergence in complex environments. It may struggle to accurately estimate the optimal action-value function and make suboptimal decisions, leading to lower average rewards compared to the average max approach. This suggests

that the DQN-GNN algorithm is the most effective of the three algorithms in terms of user association efficiency, while the Q-learning algorithm is the least effective.

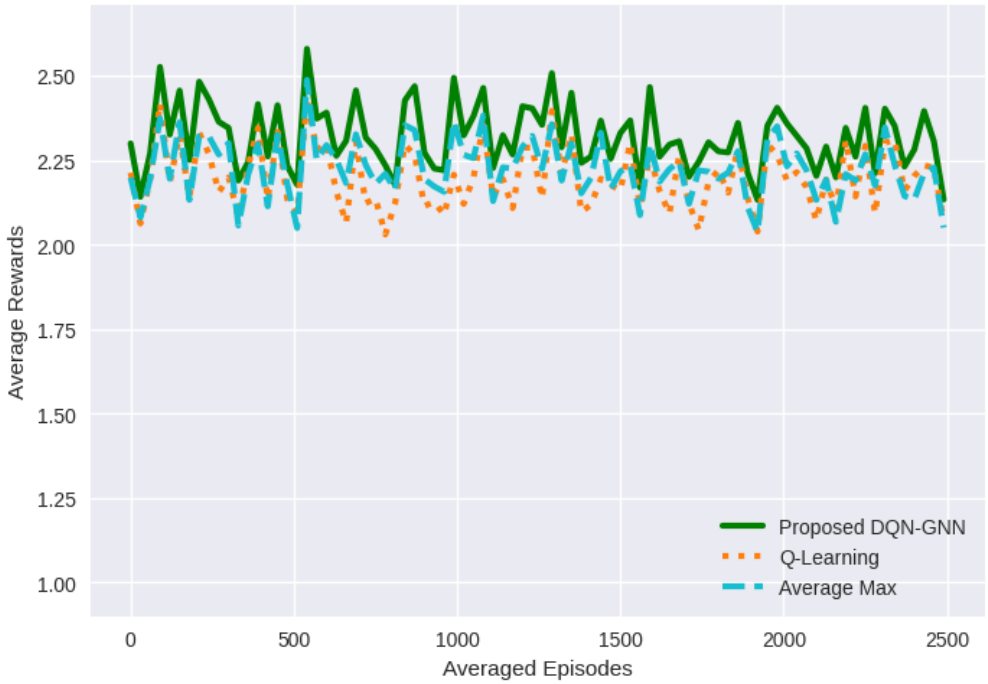

Figure 5. Average rewards.

In Figure 6, the average return versus averaged episodes is plotted for the three agents. The average return for the proposed DQN-GNN model is 4.637, which is approximately 45.96% higher than the minimum return of 3.192. In contrast, the average max approach achieves an average return of 4.445, reflecting a percentage difference of 41.65% compared to its minimum return of 3.138. Similarly, the Q-learning approach achieves an average return of 4.420, representing a percentage difference of 46.20% compared to its minimum return of 3.023. The returns for the proposed model are generally higher than those for Q-learning and Average Max. The proposed DQN-GNN model achieves the highest return values in most cases, indicating that it is more effective in solving the problem of resource allocation in wireless networks. Furthermore, it exhibits a faster convergence rate compared to the other two agents. The plot shows that the returns for the proposed DQN-GNN model converge more quickly to a high steady-state value, while the returns for Q-learning and Average Max take longer to converge and do not reach the same high steady-state values as the proposed DQN-GNN model.

Table 3 shows a comparison of average rewards and returns for the three approaches. From the table, we can infer that the Proposed Model consistently outperforms both Average Max and Q-learning in terms of average rewards and returns, as indicated by its higher minimum, maximum, and mean reward values.

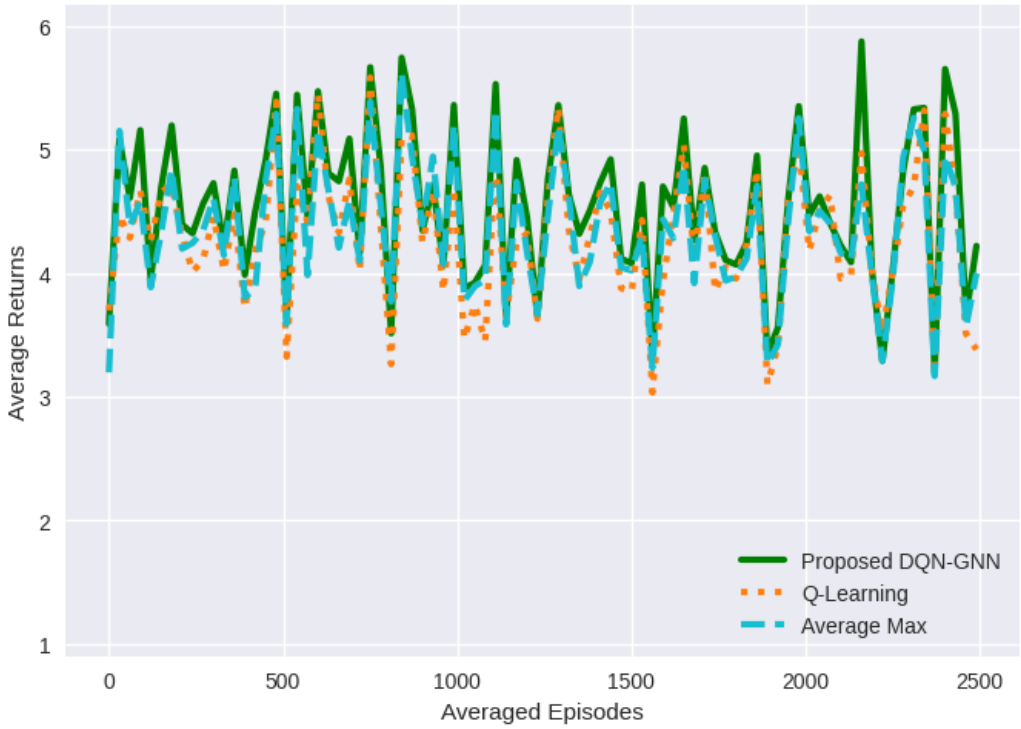

Figure 6. Average returns.

Table 3. Comparison of average rewards and returns for the three approaches.

	Proposed Model	**Average Max**	**Q-Learning**
			Average Rewards
Minimum	2.104	2.031	2.051
Maximum	2.623	2.518	2.487
Mean	2.357	2.228	2.223
			Average Returns
Minimum	3.192	3.138	3.023
Maximum	5.773	5.474	5.725
Mean	4.637	4.445	4.420

Figure 7 represents the success rate of three agents: the proposed DQN-GNN model, Q Learning, and Average Max. The success rate of the proposed DQN-GNN model was 0.952, which is higher than that of the Q Learning model (0.893) and the Average Max model (0.928). The proposed DQN-GNN model outperformed the other two models for several reasons. One reason is that the DQN-GNN model combines the strengths of both the DQN and GNN models, which enables it to learn more complex tasks and achieve higher success rates. The DQN model uses deep reinforcement learning algorithms to learn and optimize the policy of the agent, while the GNN model helps to capture the spatial relationships between the different states in the environment. This combination of techniques allows the DQN-GNN model to learn more efficiently and effectively. Additionally, the DQN-GNN model uses a replay buffer to store and replay previous experiences, helping to reduce the correlation between experiences and improve the stability of the training process. This technique helps the DQN-GNN model to learn more effectively and achieve higher success rates.

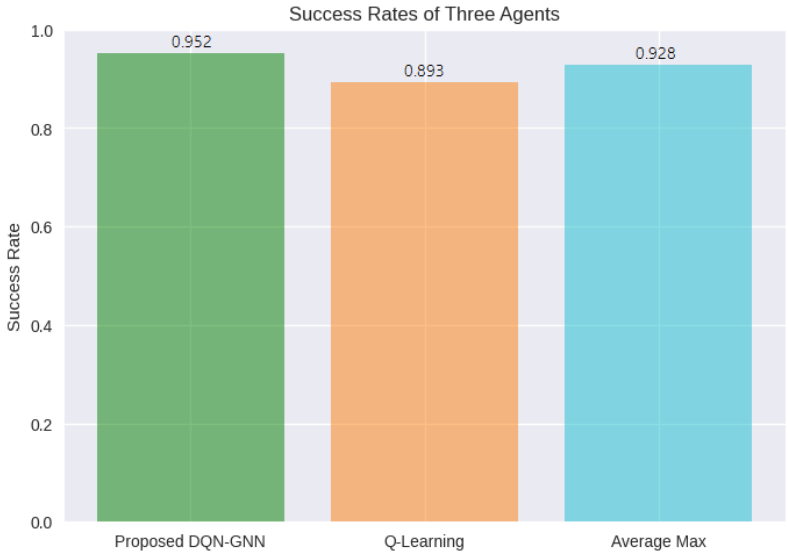

Figure 7. Success rate.

7. Qualitative Analysis and Comparison

The DQN-GNN approach offers a robust solution to user association in wireless networks due to its inherent ability to adapt to dynamic network conditions. In comparison, the Online Reinforcement Learning Approach presented by Li, Z. et al. [23] focuses on load balancing in vehicular networks. Although ORLA is effective in its specific context, it may not fully capture the complex network topologies and spatial dependencies that exist between users and base stations. The incorporation of GNNs in the DQN-GNN approach allows it to efficiently model these relationships, leading to improved user association and overall network performance.

In comparison to the Reinforcement Learning Handoff (RLH) approach proposed by Li, Q. et al. [24], the DQN-GNN approach provides a more comprehensive solution to the user association problem. Although RLH is effective in reducing redundant handoffs in UAV networks, it does not optimize the overall network utility, particularly in diverse cellular networks. The DQN-GNN approach, on the other hand, aims to optimize network performance through better user association, demonstrating its versatility in addressing different aspects of the user association problem.

On the other hand, the Dueling Double Deep Q-Network approach by Zhao et al. [38] applies multi-agent RL to optimize network utility in heterogeneous cellular networks. Although D3QN is a novel approach, it may face scalability issues with increasing network size due to its high computational complexity. Owing to the scalable nature of deep learning models, the DQN-GNN approach can be extended more efficiently to large wireless networks, thereby making it a more feasible solution for real-world applications.

Table 4 shows a qualitative analysis and comparison of the proposed DQN-GNN approach with three other approaches: ORLA, RLH, and D3QN. The comparison is conducted based on several key factors, namely spatial dependencies, scalability, communication overhead, and optimization objective.

Table 4. Comparative Analysis of DQN-GNN Approach with Similar Approaches

Approach	DQN-GNN	ORLA	RLH	D3QN
Spatial Dependencies	High	Moderate	Low	Moderate
Scalability	High	Moderate	Moderate	Low
Communication Overhead	Low	High	High	High
Optimization Objective	Network Performance	Load Balancing	Handoff Reduction	Network Utility

8. Conclusions and Future Work

In this paper, we propose a deep reinforcement learning approach that uses a Graph Neural Network to estimate the q-value function for the user association problem in wireless networks. Our approach is able to model the network topology and capture spatial dependencies between users and base stations, resulting in more accurate and efficient associations. It outperforms existing techniques, such as Q-learning and max-average approaches, in terms of average rewards and returns. Furthermore, it achieves a success rate of 95.2%, which is higher than other techniques by up to 5.9%. There are several reasons for this superior performance. First, the incorporation of GNNs into our approach allows for the accurate representation of the complex network topology and facilitates the learning of dependencies between users and BSs. By considering these factors, our approach can make more informed and context-aware decisions, which ultimately lead to better rewards and returns. Second, the use of deep reinforcement learning allows our approach to learn from past experiences and optimize the user association process over time. The DQN framework, combined with GNNs, enables our approach to estimate the q-value function more accurately, leading to more optimal and efficient associations. Additionally, the ability of GNNs to propagate information across the network graph enhances the understanding of the underlying structure and relationships, enabling our approach to exploit this information for improved decision-making. Overall, the integration of GNNs into the DQN framework empowers our approach to outperform traditional techniques by leveraging the network topology, capturing spatial dependencies, and effectively learning from past experiences, resulting in superior average rewards and returns.

In future work, we plan to extend our approach to address more complex scenarios, such as larger network topologies and dynamic environments. This expansion will allow us to explore the scalability and adaptability of our method in a broader context. We are also interested in investigating the application of transfer learning, which has shown promise in enabling knowledge transfer between tasks or domains. By leveraging pre-existing knowledge and fine-tuning GNN models, we can potentially enhance the efficiency and generalization capabilities of our approach across different network topologies. Furthermore, we recognize the importance of evaluating the robustness of our approach to uncertainties and noise sources inherent in wireless networks.

Author Contributions: Conceptualization, I.A.; Methodology, M.J.F.A.; Formal analysis, M.J.F.A.; Investigation, I.A.; Writing—original draft, I.A.; Writing—review & editing, M.J.F.A. All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project no. (IFKSUOR3–525-3).

Data Availability Statement: The code is available upon request.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

3GPP	Third Generation Partnership Project
AC	Actor-Critic
AP	Access Points
BS	Base Station
CFG	Coalition Formation Game
CSI	Channel State Information
D3QN	Dueling Double Deep Q-Network
DDPG	Deep Deterministic Policy Gradient
DEC-POMDP	Decentralized Partially Observed Markov Decision Process
DQN	Deep Q-Network
DRL	Deep Reinforcement Learning
GNN	Graph Neural Network
GUs	Ground Users
HAPS	High-Altitude Platform Station
HetNets	Heterogeneous Networks
IoT	Internet of Things
IoV	Internet of Vehicles
KPIs	Key Performance Indicators
MADRL	Multi-Agent Deep Reinforcement Learning
MDP	Markov Decision Process
mmWave	Millimeter Wave
NGW	Next Generation Wireless
OFDMA	Orthogonal Frequency Division Multiple Access
O-RANs	Open Radio Access Networks
ORLA	Online Reinforcement Learning Approach
P-DQN	Parameterized Deep Q-Network
QoE	Quality of Experience
QoS	Quality of Service
RA	Resource Allocation
RBs	Resource Blocks
RL	Reinforcement Learning
RLH	Reinforcement Learning Handoff
RSSI	Received Signal Strength Indicator
RUs	Radio Units
RVI	Relative Value Iteration
SBS	Small Base Stations
SNIR	Signal-to-Noise-and-Interference Ratio
TBS	Terrestrial Base Station
UA	User Association
UAVs	Unmanned Aerial Vehicles
UDN	Ultra-Dense Network
UE	User Equipment

References

1. Lombardi, M.; Pascale, F.; Santaniello, D. Internet of things: A general overview between architectures, protocols and applications. *Information* **2021**, *12*, 87. [CrossRef]
2. Ramazanali, H.; Mesodiakaki, A.; Vinel, A.; Verikoukis, C. Survey of user association in 5G HetNets. In Proceedings of the 2016 8th IEEE Latin-American Conference on Communications (LATINCOM), Medellin, Colombia, 15–17 November 2016; pp. 1–6.
3. Ge, X.; Cheng, H.; Guizani, M.; Han, T. 5G wireless backhaul networks: Challenges and research advances. *IEEE Netw.* **2014**, *28*, 6–11. [CrossRef]
4. Elfatih, N.M.; Hasan, M.K.; Kamal, Z.; Gupta, D.; Saeed, R.A.; Ali, E.S.; Hosain, M.S. Internet of vehicle's resource management in 5G networks using AI technologies: Current status and trends. *IET Commun.* **2022**, *16*, 400–420. [CrossRef]
5. Randall, M.; Belzarena, P.; Larroca, F.; Casas, P. GROWS: Improving decentralized resource allocation in wireless networks through graph neural networks. In Proceedings of the 1st International Workshop on Graph Neural Networking, Rome, Italy, 9 December 2022; pp. 24–29.

6. Panesar, A.; Panesar, A. Machine learning algorithms. In *Machine Learning and AI for Healthcare: Big Data for Improved Health Outcomes*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 85–144.
7. Fayaz, S.A.; Jahangeer Sidiq, S.; Zaman, M.; Butt, M.A. Machine learning: An introduction to reinforcement learning. In *Machine Learning and Data Science: Fundamentals and Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2022; pp. 1–22.
8. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 2018.
9. Moos, J.; Hansel, K.; Abdulsamad, H.; Stark, S.; Clever, D.; Peters, J. Robust reinforcement learning: A review of foundations and recent advances. *Mach. Learn. Knowl. Extr.* **2022**, *4*, 276–315. [CrossRef]
10. Yu, F.R.; He, Y. *Deep Reinforcement Learning for Wireless Networks*; Springer: Berlin/Heidelberg, Germany, 2019.
11. Kurek, M.; Jaśkowski, W. Heterogeneous team deep Q-learning in low-dimensional multi-agent environments. In Proceedings of the 2016 IEEE Conference on Computational Intelligence and Games (CIG), Santorini, Greece, 20–23 September 2016; pp. 1–8.
12. He, Y.; Liang, C.; Yu, F.R.; Zhao, N.; Yin, H. Optimization of cache-enabled opportunistic interference alignment wireless networks: A big data deep reinforcement learning approach. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–6.
13. Xiong, S.; Li, B.; Zhu, S. DCGNN: A single-stage 3D object detection network based on density clustering and graph neural network. In *Complex & Intelligent Systems*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 1–10.
14. Sathana, V.; Mathumathi, M.; Makanyadevi, K. Prediction of material property using optimized augmented graph-attention layer in GNN. *Mater. Today Proc.* **2022**, *69*, 1419–1424. [CrossRef]
15. Bhadra, J.; Khanna, A.S.; Beuno, A. A Graph Neural Network Approach for Identification of Influencers and Micro-Influencers in a Social Network:* Classifying influencers from non-influencers using GNN and GCN. In Proceedings of the IEEE 2023 International Conference on Advances in Electronics, Communication, Computing and Intelligent Information Systems (ICAECIS), Bangalore, India, 19–21 April 2023; pp. 66–71.
16. Zheng, X.; Huang, W.; Li, H.; Li, G. Research on Generalized Intelligent Routing Technology Based on Graph Neural Network. *Electronics* **2022**, *11*, 2952. [CrossRef]
17. Munikoti, S.; Agarwal, D.; Das, L.; Halappanavar, M.; Natarajan, B. Challenges and opportunities in deep reinforcement learning with graph neural networks: A comprehensive review of algorithms and applications. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**. [CrossRef]
18. Hara, T.; Sasabe, M. Deep Reinforcement Learning with Graph Neural Networks for Capacitated Shortest Path Tour based Service Chaining. In Proceedings of the 2022 IEEE 18th International Conference on Network and Service Management (CNSM), Thessaloniki, Greece, 31 October–4 November 2022; pp. 19–27.
19. Long, Y.; He, H. Robot path planning based on deep reinforcement learning. In Proceedings of the 2020 IEEE Conference on Telecommunications, Optics and Computer Science (TOCS), Shenyang, China, 11–13 December 2020; pp. 151–154.
20. Wang, X.; Gu, Y.; Cheng, Y.; Liu, A.; Chen, C.P. Approximate policy-based accelerated deep reinforcement learning. *IEEE Trans. Neural Networks Learn. Syst.* **2019**, *31*, 1820–1830. [CrossRef]
21. Pérez-Gil, Ó.; Barea, R.; López-Guillén, E.; Bergasa, L.M.; Gomez-Huelamo, C.; Gutiérrez, R.; Diaz-Diaz, A. Deep reinforcement learning based control for Autonomous Vehicles in CARLA. *Multimed. Tools Appl.* **2022**, *81*, 3553–3576. [CrossRef]
22. Zhu, X.; Dong, H. Shear Wave Velocity Estimation Based on Deep-Q Network. *Appl. Sci.* **2022**, *12*, 8919. [CrossRef]
23. Li, Z.; Wang, C.; Jiang, C.J. User association for load balancing in vehicular networks: An online reinforcement learning approach. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2217–2228. [CrossRef]
24. Li, Q.; Ding, M.; Ma, C.; Liu, C.; Lin, Z.; Liang, Y.C. A reinforcement learning based user association algorithm for UAV networks. In Proceedings of the 2018 IEEE 28th International Telecommunication Networks and Applications Conference (ITNAC), Sydney, Australia, 21–23 November 2018; pp. 1–6.
25. Ding, H.; Zhao, F.; Tian, J.; Li, D.; Zhang, H. A deep reinforcement learning for user association and power control in heterogeneous networks. *Ad Hoc Netw.* **2020**, *102*, 102069. [CrossRef]
26. Chou, P.Y.; Chen, W.Y.; Wang, C.Y.; Hwang, R.H.; Chen, W.T. Deep reinforcement learning for MEC streaming with joint user association and resource management. In Proceedings of the ICC 2020 IEEE International Conference on Communications (ICC), Virtually, 7–11 June 2020; pp. 1–7.
27. Guan, X.; Huang, Y.; Dong, C.; Wu, Q. User association and power allocation for uav-assisted networks: A distributed reinforcement learning approach. *China Commun.* **2020**, *17*, 110–122. [CrossRef]
28. Zhang, Q.; Liang, Y.C.; Poor, H.V. Intelligent user association for symbiotic radio networks using deep reinforcement learning. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 4535–4548. [CrossRef]
29. Sana, M.; De Domenico, A.; Yu, W.; Lostanlen, Y.; Strinati, E.C. Multi-agent reinforcement learning for adaptive user association in dynamic mmWave networks. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 6520–6534. [CrossRef]
30. Dinh, T.H.L.; Kaneko, M.; Wakao, K.; Kawamura, K.; Moriyama, T.; Abeysekera, H.; Takatori, Y. Deep reinforcement learning-based user association in sub6GHz/mmWave integrated networks. In Proceedings of the 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 9–12 January 2021; pp. 1–7.
31. Hsieh, C.K.; Chan, K.L.; Chien, F.T. Energy-efficient power allocation and user association in heterogeneous networks with deep reinforcement learning. *Appl. Sci.* **2021**, *11*, 4135. [CrossRef]
32. Chen, G.; Zhai, X.B.; Li, C. Joint optimization of trajectory and user association via reinforcement learning for UAV-aided data collection in wireless networks. *IEEE Trans. Wirel. Commun.* 2022, *22*, 3128–3143. [CrossRef]

33. Joda, R.; Pamuklu, T.; Iturria-Rivera, P.E.; Erol-Kantarci, M. Deep Reinforcement Learning-Based Joint User Association and CU–DU Placement in O-RAN. *IEEE Trans. Netw. Serv. Manag.* **2022**, *19*, 4097–4110. [CrossRef]
34. Alizadeh, A.; Vu, M. Reinforcement learning for user association and handover in mmwave-enabled networks. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 9712–9728. [CrossRef]
35. Khoshkbari, H.; Sharifi, S.; Kaddoum, G. User Association in a VHetNet with Delayed CSI: A Deep Reinforcement Learning Approach. *IEEE Commun. Lett.* **2023**, *27*, 2257–2261. [CrossRef]
36. Moon, J.; Kim, S.; Ju, H.; Shim, B. Energy-Efficient User Association in mmWave/THz Ultra-Dense Network via Multi-Agent Deep Reinforcement Learning. *IEEE Trans. Green Commun. Netw.* **2023**, *7*, 692–706. [CrossRef]
37. Kim, D.U.; Park, S.B.; Hong, C.S.; Huh, E.N. Resource Allocation and User Association Using Reinforcement Learning via Curriculum in a Wireless Network with High User Mobility. In Proceedings of the 2023 International Conference on Information Networking (ICOIN), Bangkok, Thailand, 11–14 January 2023; pp. 382–386.
38. Zhao, N.; Liang, Y.C.; Niyato, D.; Pei, Y.; Wu, M.; Jiang, Y. Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 5141–5152. [CrossRef]
39. Shaddad, R.Q.; Neda'a, A.A.; Alzylai, M.O.; Shami, T.M. Biased user association in 5G heterogeneous networks. In Proceedings of the IEEE 2021 International Conference of Technology, Science and Administration (ICTSA), Taiz, Yemen, 22–24 March 2021; pp. 1–4.
40. Ji, Z.; Hu, Z.; Wang, Y.; Shao, Z.; Pang, Y. Reinforced pedestrian attribute recognition with group optimization reward. *Image Vis. Comput.* **2022**, *128*, 104585. [CrossRef]
41. Lee, H.; Eom, C.; Lee, C. QoS-Aware UAV-BS Deployment Optimization Based on Reinforcement Learning. In Proceedings of the 2023 International Conference on Electronics, Information, and Communication (ICEIC), Beijing, China, 14–16 July 2023; pp. 1–4.
42. Badakhshan, S.; Jacob, R.A.; Li, B.; Zhang, J. Reinforcement Learning for Intentional Islanding in Resilient Power Transmission Systems. In Proceedings of the 2023 IEEE Texas Power and Energy Conference (TPEC), College Station, TX, USA, 13–14 February 2023; pp. 1–6.
43. Kim, S.; Jang, M.G.; Kim, J.K. Process design and optimization of single mixed-refrigerant processes with the application of deep reinforcement learning. *Appl. Therm. Eng.* **2023**, *223*, 120038. [CrossRef]
44. Ballard, T.; Luckman, A.; Konstantinidis, E. A systematic investigation into the reliability of inter-temporal choice model parameters. In *Psychonomic Bulletin & Review*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 1–29.