February 2022
Master’s Degree Thesis

Artificial Intelligence for UAV-assisted 5G Heterogeneous NOMA Systems with Priority-based Joint Resource Allocation

Graduate School of Chosun University

Department of Computer Engineering

Sifat Rezwan

Artificial Intelligence for UAV-assisted 5G Heterogeneous NOMA Systems with Priority-based Joint Resource Allocation

UAV 지원 5G 이기종 NOMA 시스템의 우선순위
기반 자원할당을 위한 인공지능 기술 연구

25th February, 2022

Graduate School of Chosun University

Department of Computer Engineering

Sifat Rezwan

Artificial Intelligence for UAV-assisted 5G Heterogeneous NOMA Systems with Priority-based Joint Resource Allocation

Advisor: Prof. Wooyeol Choi

A thesis submitted in partial fulfillment of the requirements for a Master’s degree

October 2021

Graduate School of Chosun University

Department of Computer Engineering

Sifat Rezwan

시팟 레즈완 석사학위논문을 인준함

위원장 조선대학교 교수 **신석주** (인)

위 원 조선대학교 교수 **강문수** (인)

위 원 조선대학교 교수 **최우열** (인)

2021년 12월

조선대학교 대학원

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Artificial Intelligence for UAV-assisted 5G Heterogeneous NOMA Systems with Priority-based Joint Resource Allocation

Sifat Rezwan
Advisor: Prof. Choi, Wooyeol, Ph.D.
Department of Computer Engineering
Graduate School of Chosun University

For heterogeneous demands in fifth-generation (5G) new radio (NR), massive machine type communication (mMTC), enhanced mobile broadband (eMBB), and ultra-reliable and low-latency communication (URLLC) services have been introduced. Non-orthogonal multiple access (NOMA) has been introduced to ensure these quality-of-services (QoS) requirements in which multiple devices can be served from the same frequency by manipulating the power domain and successive interference cancellation (SIC) technique. To maximize the efficiency of NOMA systems, optimal resource allocation, ensuring transmission link quality are the key issues that need to be solved. In this thesis, we propose a priority-based channel assignment with a deep *Q*-learning algorithm to maintain the 5G QoS requirements and increase the network performance. We formulate an optimal power allocation scheme under Karush–Kuhn–Tucker (KKT) optimality conditions incorporating different NOMA constraints. The main objectives are to maximize the channel sum-rate, system sum-rate, and system fairness. We also propose a novel FDRL-based multiple UAV-BS navigation scheme to serve the 5G devices suffering from NLOS, poor link quality, and multi-path fading with maximum coverage, link quality, and fairness. Finally, We conduct extensive simulations with respect to different system parameters and confirm that the proposed schemes perform better than other state-of-the-art schemes.

한글요약

UAV지원 5G 이기종 NOMA 시스템의 우선순위 기반 자원할당을 위한 인공지능 기술 연구

시팟 레즈완
지도교수: 최우열
컴퓨터공학과
조선대학교 대학원

5세대 무선 통신의 이기종 수요를 위해 massive machine type communication(mMTC), enhanced mobile broadband(eMBB), ultra reliable and low latency communication(URLLC) 기술이 발전하고 있다. 또한, 동일한 주파수에서 quality-of-service(QoS)를 보장하며 다수의 무선 신호를 동시 수신할 수 있도록 지원하기 위해, non-orthogonal multiple access(NOMA) 기술이 연구되고 있다. 이러한 NOMA 시스템에서 통신의 효율을 극대화하기 위해서는 최적의 자원 할당과 전송링크의 품질 보장이 해결되어야 한다.

본 논문에서는 5G 네트워크의 QoS 요구사항을 보장하고 네트워크 성능을 향상시키기 위해, 심층 Q학습 알고리즘을 사용하는 우선 순위 기반 채널할당 기술을 제안한다. NOMA 시스템의 다양한 제약조건을 통합하기 위해, Karush-Kuhn-Tucker(KKT) 조건을 활용하여 최적의 전력 할당 방식을 수학적으로 도출하고, 각 채널 및 전체 시스템의 sum-rate와 공정성을 최대화한다.

또한, UAV 지원 무선 네트워크를 고려하여, 연합학습을 활용한 강화학습 기반 다중 UAV-BS 탐색 방식을 제안한다. 이는 non-line-of-sight (NLOS), 열악한 무선 링크 품질 및 통신 범위, 다중 경로 페이딩에 의한 5G 네트워크 성능 저하를 극복하도록 설계한다.

다양한 시스템 매개변수에 대한 광범위한 시뮬레이션을 수행하여, 제안하는 기술이 다른 최신 기술에 비해 우수한 성능을 가짐을 확인한다.

I. INTRODUCTION

The fifth-generation (5G) wireless communication systems are rapidly growing owing to the high data rates, massive connectivity, and high quality of service (QoS) requirements [1]. The 3rd Generation Partnership Project (3GPP) divided these characteristics into three significant services. These major services include massive machine type communication (mMTC) that allows massive connectivity for IoT devices, enhanced mobile broadband (eMBB) that provides a high data rate for mobile platforms, and ultra-reliable and low-latency communication (URLLC) that ensures reliability and low latency for sensitive and crucial applications [2]–[4]. These services are categorized in terms of their quality-of-service (QoS), where URLLC has a strict QoS policy for high reliability and low latency, eMBB service has a moderate QoS policy, and mMTC has no specific QoS policy except for massive connectivity [5].

These types of QoS policies are tough to fulfill with the traditional orthogonal multiple access (OMA) due to limited spectrum resources, significant transmission losses, and long queuing delays [6], [7]. Many potential technologies have been introduced into the 5G communication network to maintain these diverse QoS requirements [8]. Among them, non-orthogonal multiple access (NOMA) is gaining popularity because it can support massive connectivity with limited resources, highly reliable transmissions, low transmission delays, and high spectral efficiency [9]–[11]. The key feature of NOMA is that multiple devices can be served from the same radio resource block (RRB), such as time, frequency, and codes, simultaneously utilizing the power domain [12], [13]. NOMA applies superposition coding to combine signals of multiple devices at the transmitter and successive interference cancellation (SIC) at the receiver to differentiate the signals of multiple devices manipulating the power domain [14], [15]. The NOMA system not only mitigates the multiple access interference but also increases the spectral efficiency and device fairness [16]. Thus, NOMA can easily maintain strict QoS policies for eMBB, mMTC, and URLLC services. By contrast, with conventional OMA, only one device can be served from each RRB at a time to avoid multiple access interference, which is insufficient to support high data rates and massive connectivity [17].

However, some significant challenges include power allocation, channel assignment, transmission link quality, multi-path fading, and non-line-of-sight (NLOS) in the 5G NOMA systems. One of the significant challenges is that joint power allocation and channel assignment involve a mixed-integer program which is a non-deterministic polynomial-time hard (NP-hard) problem [18]–[20]. For example, all possible combinations of channel assignment and power allocation are required to reach an optimal solution which makes the system complicated and requires extremely high computational power [21], [22]. When it comes to multi-carrier NOMA, the system becomes more complex. In multi-carrier NOMA, the channel sum-rate fairness is another problem as an increase in the system sum-rate does not necessarily increase the sum-rate of each channel. The Poor sum-rate of any channel can decrease the performance of the devices assigned to that channel [23].

In addition, overall spectral efficiency and sum-rate decrease owing to multi-path fading, link quality, and NLOS problems. Moreover, perfect signal decoding using SIC and fulfilling the QoS requirements of 5G services also depends on the resource allocation, and link quality [24]. An imperfect SIC can quickly decrease the overall performance of the system. Therefore, we divided this thesis into two parts. We investigate the power allocation and channel assignment jointly in the first part and the link quality, multi-path fading, and NLOS in the second part to overcome the challenges of the downlink NOMA system under various criteria.

A. Related Works

This section discusses the related works done by different researchers over the last few years to optimize the 5G NOMA systems. We divide this section into two parts. We review the literature addressing resource allocation challenges in the first part and link quality and coverage in the second part.

1. Resource allocation

Optimal resource allocation, such as power allocation and channel assignment, is the key to increase the overall system performance and fulfill the QoS requirements of the 5G network. Many researchers have proposed many approaches to obtain optimal solutions with different performance objectives [25], [26]. The most common objectives are to maximize the overall sum-rate of the system and fulfill the minimum data rate.

Ali *et al.* [25] proposed a power allocation technique with a user grouping scheme for a single-carrier NOMA system to maximize the sum-rate using Lagrange equations under Karush–Kuhn–Tucker (KKT) conditions. The authors have derived the Lagrange equations to obtain an optimal power allocation scheme while considering total power limitation, minimum data rate requirement, and SIC constraints under Karush–Kuhn–Tucker (KKT) conditions. Shao *et al.* [27] derived a dynamic device clustering technique and an optimal power allocation solution using the Nash bargaining solution (NBS) for the NOMA system based on the number of devices and channel gains. However, only a single-carrier NOMA system for IoT devices is considered. In [5], Shahini *et al.*

proposed priority-based URLLC and mMTC device grouping with fixed power allocation scheme. However, no authors considered the presence of URLLC, eMBB, and mMTC services in 5G networks. Parida *et al.* [28] solved only the non-convex power allocation problem using the difference of two convex functions (DC) programming to maximize the sum-rate of orthogonal frequency division multiple access (OFDMA)-based NOMA system. In another paper [29], Hojeij *et al.* used the water-filling algorithm for resource allocation to obtain the highest sum-rate possible. However, no optimality was provided for the obtained solution.

Nevertheless, the system sum-rate increases when it comes to multi-carrier NOMA. In [30], Zhu *et al.* derived a near-optimal power allocation solution considering two users per channel and iteratively assigned channels to the users. They also considered the minimum data rate constraints for each user while maximizing the sum-rate. However, the authors did not consider the different services of the 5G network. Choi *et al.* [26] used convex optimization to approximate the maximization problem for the minimum data rate requirement of users. Ning *et al.* [31] adopted a heuristic approach to solve the power allocation and channel assignment problem of the NOMA system for vehicular ad-hoc networks.

In addition to conventional convex optimization, many researchers explored the machine learning and artificial intelligence sectors to optimize the resource allocation problem of the NOMA system. In [32], Xiao *et al.* proposed fast and dynamic reinforcement learning (RL) based power allocation to maximize sum-rate and spectral efficiency of a multiple-input multiple-output (MIMO) NOMA system in the presence of smart jamming. The authors initially formulated the anti-jamming transmission game and derived the Stackelberg equilibrium of the game. *Q*-learning-based power allocation is then used to allocate power to users against jamming attacks. He *et al.* [33] proposed a joint power allocation and channel assignment for the NOMA system using deep reinforcement learning (DRL). They used the derived near-optimal power allocation from [30] considering two users per channel and performed channel assignment using a DRL algorithm consisting of an attention-based neural network. The authors then used a DRL algorithm consisting of an attention-based neural network to perform channel assignment while maximizing the overall sum-rate and minimum data rate for user fairness. An actor-critic (A2C) RL algorithm was used in [34] to obtain the optimal policy for resource allocation and user scheduling in HetNets with a hybrid energy supply. The actor parameterizes the policy using the Gaussian distribution to take stochastic actions, and the critic evaluates the value function and helps the actor learn the optimal policy.

In summary, many researchers found many optimal and near-optimal power allocation solutions for a single-carrier only. Most researchers focused on increasing the overall sum-rate while maintaining a minimum data rate for fairness. However, an increase in the overall sum-rate does not ensure an increase in the sum-rate of each channel. Furthermore, the sum-rate of a device is directly connected with the sum-rate of the channel.

2. Link quality and Coverage

In 5G communication networks, the geographical distribution of base stations (BSs) is planned to support a large amount of traffic for a long time with minimum latency, highest reliability, and massive connectivity [35]. Thus, the radio cell architecture is shifting toward small cells with low transmit power compared to previous generations of wireless communications [36]. However, establishing numerous BSs to cover unforeseeable traffic and poor connectivity areas is not economical and efficient [37]. Many researchers have proposed solutions utilizing unmanned aerial vehicles (UAVs) to provide a wide range of services and radio coverage [38].

Al-hourani *et al.* [39] proposed a mathematical model to obtain the optimum altitude of low-altitude aerial platforms (LAPs) for maximum coverage. The authors considered the percentage of built-up area to the total land area, the number of buildings per unit area, and the statistical distribution of buildings heights to estimate the line-of-sight (LOS) probability in a closed-form. Shi *et al.* [35] proposed a drone iterated particle swarm optimization (DI-PSO) algorithm to maximize the user coverage ratio by the drones while ensuring drone-BS channel quality in 3D space. The authors formulated the drone cell deployment problem as an NP-hard problem and solved it for each drone cell iteratively. Sharma *et al.* proposed a UAV-based solution to solve the coverage and capacity enhancement problem of 5G heterogeneous networks in [40]. The objectives of the proposed solution are the deployment of the UAV-BSs and cooperative network formation for addressing the traffic load. The authors adopted priority-wise dominance and the entropy method for optimality. Fotouhi *et al.* [41] developed two distributed algorithms to serve users with maximum spectral efficiency and minimum interference. The UAV-BS dynamically re-position itself using these two algorithms to increase the spectral efficiency and minimize the interference from other neighboring UAV-BSs. The same authors also proposed a service on-demand-based solution where UAVs serve multiple ground users balancing the traffic load in [42]. Similarly, Lyu *et al.* in [43] proposed a spiral placement algorithm where multiple UAV-BSs are placed sequentially in an inward spiral manner to cover all the ground users.

In contrast to the conventional techniques, many researchers explored the machine learning sectors to optimize the UAV trajectory for better coverage and line quality. In [44], Challita *et al.* proposed a DRL framework using echo state network (ESN) cells for optimizing trajectories of multiple UAV-BSs. The authors considered the whole network topology, QoS requirements, and location of other UAVs to learn optimal path, transmit power level, and user association vector. Abedin *et al.* proposed a DRL-based energy-efficient UAV-BS navigation solution for 5G wireless networks where a UAV-BS cruises over the ground users to maintain data freshness in [45]. The authors utilized a conventional DRL framework with experience replay memory to train the UAV-BSs. In contrast, Liu *et al.* proposed distributed multi-UAV navigation framework to establish energy-efficient long-term communication in disastrous scenarios utilizing DRL in [46]. The authors used an actor-critic framework to keep track of all the UAV-BSs serving the ground users.

In summary, many researchers have developed UAV-based solutions for better link quality and coverage. Some crucial parameters need to be considered while developing UAV-based solutions, which include UAV energy constraints, UAV charging, multi-UAV cooperation with minimum overhead, UAV navigation environment, obstacles, UAV to ground user channel quality, LOS, interference, and collision with other UAVs. However, no researchers have addressed all the problems in a single solution.

B. Contributions

In this thesis, we investigate resource allocation schemes to maximize the performance of multi-carrier NOMA systems under multiple performance metrics. We also investigate multi-UAV navigation schemes to provide better link quality and coverage to the ground devices considering multiple crucial parameters. We propose a priority-based joint resource allocation scheme with DQL and a federated DRL (FDRL)-based multi-UAV navigation scheme for heterogeneous NOMA systems considering the UAV performance constraints and the key constraints and services of 5G networks.

The contributions of proposed resource allocation scheme are described as follows:

- We formulate an optimal power allocation scheme that maximizes the overall system efficiency for any given channel assignment using Lagrange multipliers under KKT optimality conditions and incorporates different constraints of NOMA.

- We propose a priority-based channel assignment scheme using deep *Q*-learning (DQL) to maximize the performance and fairness of multi-carrier NOMA. We prioritize the devices present in the 5G network based on the QoS requirement and categorize them based on URLLC, eMBB, and mMTC services. The agent of the DQL explores the 5G network environment and learns the prioritization and channel assignment to achieve an optimal policy. We use an autoencoder architecture for the policy network, followed by a long short-term memory (LSTM) network.

- We consider different constraints of the NOMA system, including the total power budget of the base station (BS), the minimum data rate requirement of each device, the QoS policies of different services of the 5G network, and the sum-rate maximization with channel fairness constraints.

- We consider maximizing sum-rate (MSR), maximizing channel sum-rate (MCSR), and maintaining the 5G QoS policies as our main objectives.
- Finally, we analyze and compare the proposed schemes in different scenarios with the conventional OMA system.

The contributions of proposed multi-UAV navigation scheme are described as follows:

- We propose a novel multi-UAV navigation scheme using FDRL to serve URLLC, mMTC, and eMBB devices suffering from NLOS, poor link quality, and multi-path fading in 5G heterogeneous networks.
- We deploy multiple UAV-BSs to serve the ground devices in the suffered area. The UAV-BSs cruise over the ground devices (GDs) in the 3D considering the LOS, UAV-to-device channel gain and serve them using the 5G NOMA system. The prime objectives of the UAV-BSs are to cover as many GDs as possible with a minimum amount of energy for the maximum amount of time.
- We utilize the federated learning (FL) algorithm to train the UAV-BSs in a distributed manner. Thus, each UAV-BS does not have to learn the actions of other UAV-BSs and can easily replace another UAV-BS in case of emergency. We consider the BS as the central server for model aggregation and the UAV-BSs as DRL-based learning agents. Moreover, we incorporate

autonomous visits for charging where a UAV-BS can return to the BS while another fully charged UAV-BS replaces it.

- We have utilized the proportional–integral–derivative (PID) controller to drive the UAV-BSs. We also have clustered ground devices in the suffered area utilizing the K-means algorithm to avoid collision among UAVs.
- Finally, we analyze and compare the proposed schemes in different scenarios with other baseline schemes.

C. Thesis Layout

The thesis is organized as follows. In Chapter II, we present fundamentals of the reinforcement learning. Then in chapter III, we describe the problem statement, proposed solution and simulation analysis of priority-based joint resource allocation with DQL. Next in Chapter IV, we describe the problem statement, proposed solution and simulation analysis of FDRL-Based multi-UAV navigation. And finally, we conclude the thesis in Chapter V.

II. Fundamentals of Reinforcement Learning

To understand the proposed solution, we have to understand the fundamentals of reinforcement learning (RL), deep reinforcement learning (DRL), and federated deep reinforcement learning (FDRL). Thus, we briefly discuss the internal structure, decision-making process, and convergence process of RL, DRL, and FDRL in this chapter.

A. Reinforcement learning

Reinforcement learning is an effective and extensively used tool of AI which learns about the environment by taking different actions and achieves an optimal policy for operation. The RL consists of two main components: an agent and an environment. The agent explores the environment and decides which action to take using the Markov decision process (MDP) [47].

MDP is a framework for modeling decision-making problems and helping the agent to control the process stochastically [47]. MDP is an useful tool for dynamic programming and RL techniques. Generally, MDP has four parameters represented by the tuple (S, A, p, r), where S is a finite state space, A is a finite action space, p is the transition probability from the present state s to the next state s' after taking action a, and r is the immediate reward given by the environment for action a [48]. As shown in Fig. 1, at each time step t, the agent observes its present state s_t in the environment and takes action a_t. Then, the agent receives a reward r_t and the next state s_{t+1} from the environment. The main goal of the agent is to determine a policy π to accumulate the maximum possible reward from the environment. In long term, the agent also tries to maximize the expected discounted total reward defined by $\max[\sum_{t=0}^{T} \delta r_t(s_t, \pi(s_t))]$, where $\delta \in [0,1]$ is the discount factor. Using the discounted reward, a Bellman equation named the

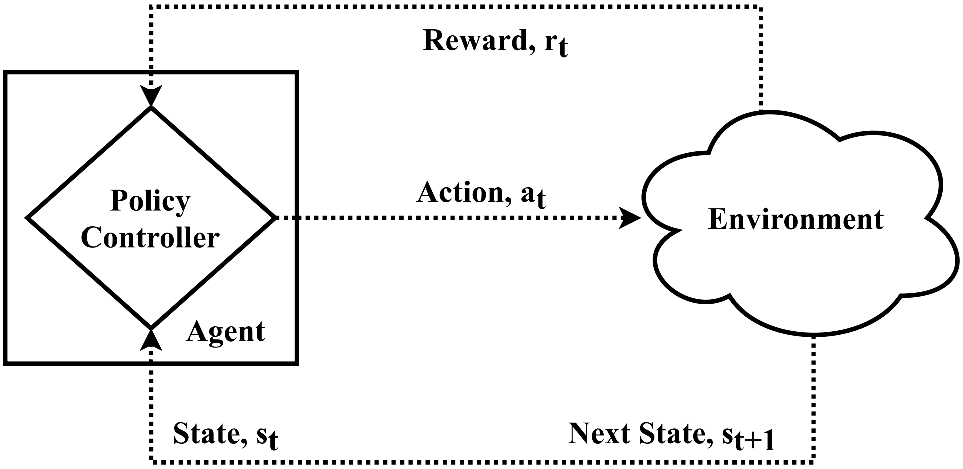


Figure 1: The agent-environment in Markov decision process.

Q-function (2) is formed to take the next action a_t using MDP when the state transition probabilities are known in advance. The Q-function can be expressed as

$$Q(s_t, a_t) = (1-\alpha) \times Q(s_t, a_t) + \alpha[r + \delta(\max Q(s_{t+1}, a_t))], \tag{1}$$

where α is the learning rate.

Algorithm 1 The Q-learning Algorithm

```
$Q(S,A) = 0$.
Initialize $\alpha, \delta, \varepsilon$.
for $t = 1, 2, \ldots, T$ do  Choose an action $a_t$ for present state $s_t$ based on the value of $\varepsilon$.
Obtain an immediate reward $r_t$ and next state $s_{t+1}$.
Update $Q(S,A)$ via Markov decision process (2).
$s_t \leftarrow s_{t+1}$  Optimal policy, $\pi(s) = \arg\max Q(S,A)$
```

RL with a Q-function is also known as Q-learning. Initially, the agent explores

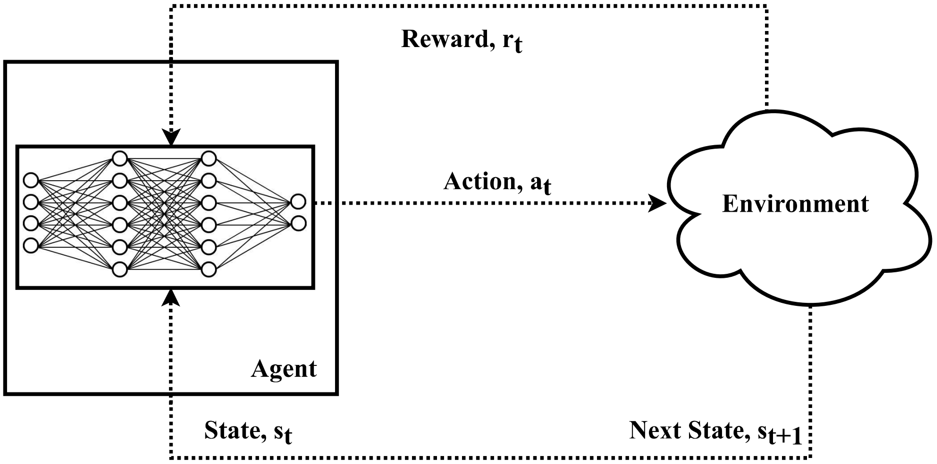


Figure 2: Simple Deep *Q*-learning.

every state of the environment taking different actions and forms a *Q*-table using the *Q*-function for each state-action pair. Then, the agent starts exploiting the environment by taking actions with the maximum *Q*-value from the *Q*-table. This policy is known as the ε-greedy policy, where the agent starts exploring or exploiting the environment depending on the value of the probability ε. An illustration of *Q*-learning is presented in Algorithm 1.

B. Deep reinforcement learning

The *Q*-learning algorithm is efficient in terms of its comparatively small action and state space. However, the system becomes more complicated for large action and state space. In this situation, the *Q*-learning algorithm may not be able to achieve an optimal policy owing to the complex and large *Q*-table. To overcome this problem, researchers replaced the *Q*-table with a deep neural network (DNN)

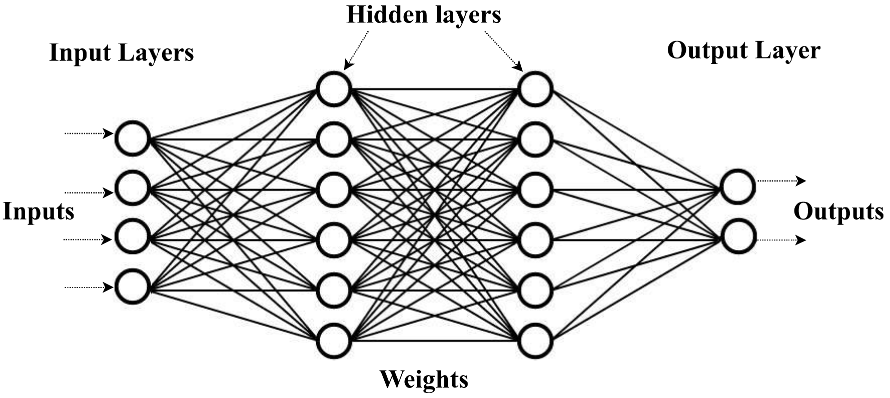


Figure 3: Deep Neural Network.

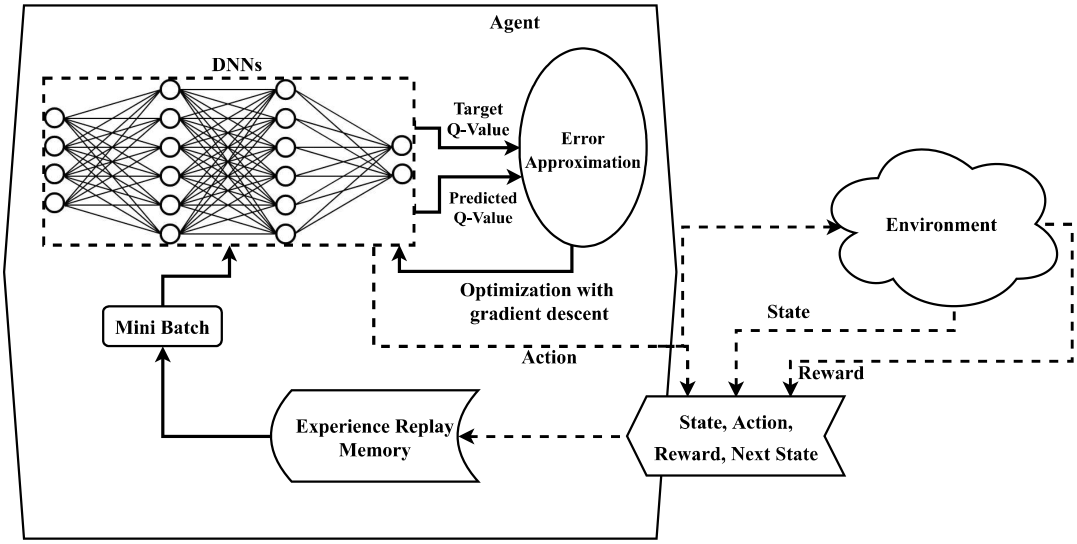


Figure 4: DQL Framework.

and named it deep Q-learning (DQL) [49]. DQL is a DRL that works with Q-values similar to Q-learning, except for the Q-table part as shown in Fig. 2.

The main goal of the DNN is to skip manual calculations each time by learning from the data. A DNN is a computational non-linear model like the structure of the human brain, which can learn and perform tasks such as decision-making, prediction, classification, and visualization [50]. It is composed of neurons arranged in multiple layers. It typically has one input layer, two hidden layers, and on output layer, interconnected as depicted in Fig. 3 [51]. The input layer accepts the inputs with the input neurons and sends them to the hidden layers. The hidden layer then sends the data to the output layer. Every neuron has a weighted input, an activation function, and an output. The activation function determines the output depending on the input of the neuron [52]. It acts as a trigger that depends on the weighted input.

Algorithm 2 The Deep *Q*-learning Algorithm

```
Initialize policy and target DQL network with random $w$ and $w'$, respectively.
Initialize experience replay memory (ERM).
Initialize $\varepsilon$.
for $t = 1, 2, \ldots, T$ do  Select an action $a_t$ for present state $s_t$ based on probability $\varepsilon$.
Observe the immediate reward $r_t$ and next state $s_{t+1}$.
Insert $(s_t, a_t, r_t, s_{t+1})$ in ERM.
Create a mini-batch with random sample of $(s_t, a_t, r_t, s_{t+1})$ from ERM.
Optimize the weights $w$ of the policy DNN with gradient descent via MDP.
$w' \leftarrow w$ after certain number of time steps.
```

During the training phase, the weighted values of the inputs of the neurons are updated based on the outputs of the output layer using backpropagation by the agent. The agent takes the output of the policy DNN and compares it with a target DNN model and calculates error [53]. Then the agent updates the policy DNN using backpropagation. This process is generally referred as optimization with gradient descent. After a certain time, the agent updates the target DNN using policy DNN. For a more stable convergence of the optimal policy, experience replay memory (ERM) is introduced into the DQL framework [54], [55]. The agent takes different actions and saves the present states, obtained rewards, next states, and actions taken in ERM [54], [55]. Then, the agent takes a mini-batch of data from the ERM and trains the policy DNN. Fig. 4 and Algorithm 2 illustrate the framework and flow of the DQN better [56]. Thus, the agent can make decisions efficiently and in a timely manner using the learned DNN.

C. Federated deep reinforcement learning

The key idea of FL is to train a machine learning model in a distributed manner across multiple devices using local data-sets without sharing them [57], [58]. Google introduced FL recently to overcome their mobile users' statistical and data security challenges [59], [60]. Many researchers are currently working on FL to make it personalized. The main focus involves optimizing the distributed mobile device interactions, communication costs, data distribution, and device reliability [60].

In general, FL involves N devices $(G_1, G_2, \ldots, G_N)$ with their local data $(D_1, D_2, \ldots, D_N)$ to train themselves utilizing a machine-learning model such as deep learning and deep reinforcement learning. The conventional way is to collect all the data and train a single model MS. However, in the FL system, the individual devices will train their model, and a central server will aggregate all local models $(M_{L1}, M_{L2}, \ldots, M_{LN})$ to create a global model M_{FD} as shown in Fig. 5. The accuracy of the M_{FD} should be close to the M_S. The accuracy

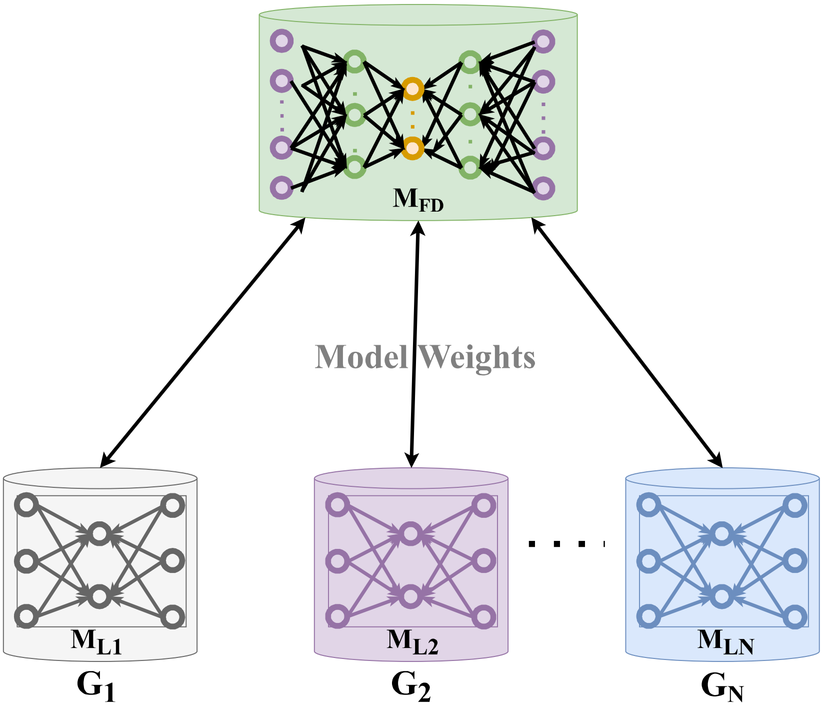


Figure 5: Simple Federated Learning.

difference can be denoted as δ_f.

This sub-section discusses the FL utilizing DRL, also known as FDRL. Generally, FDRL can be categorized into Horizontal FDRL (HFDRL) and Vertical FDRL (VFDRL), depending on the architecture of the FL algorithm.

1. HFDRL

Many researchers have been studying parallel RL for an extended period, in which multiple agents are interacting with the different environments to perform the same type of tasks. The agents are also learning the actions taken by other agents to cooperate. However, things get more complex and energy-inefficient when the number increases. Moreover, there is no privacy preservation among

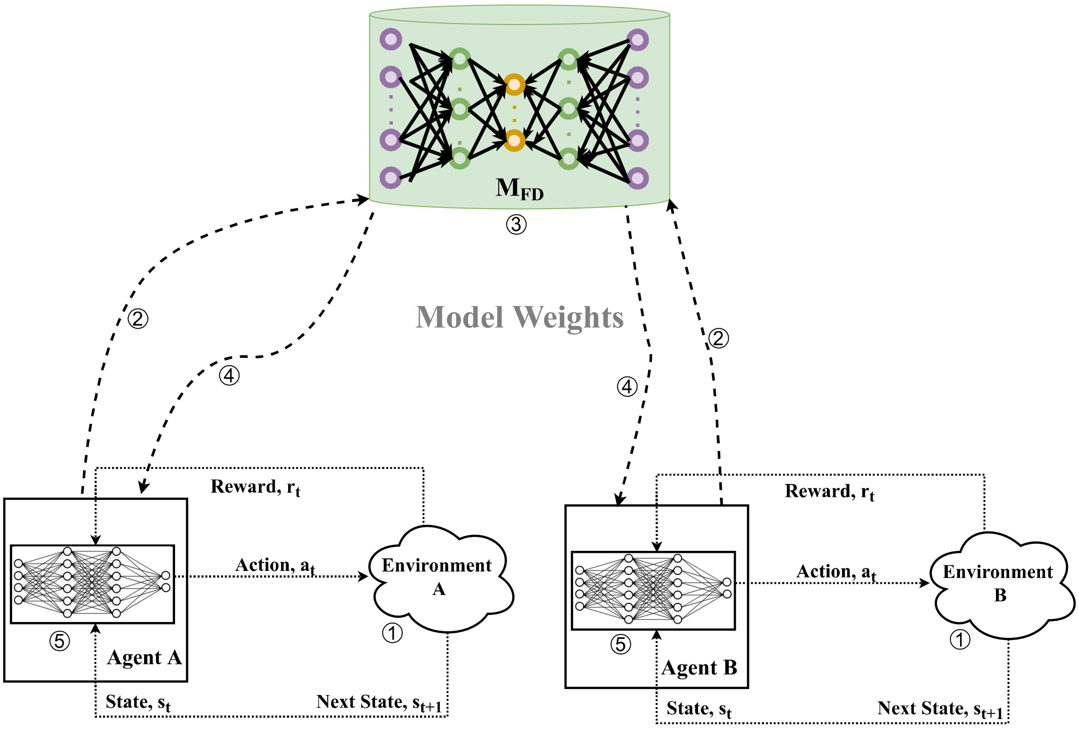


Figure 6: HFDRL Framework.

the agents. Therefore, it is necessary to adopt HFDRL to lower the complexity, increase the energy-efficiency, and provide privacy. A basic HFDRL framework is shown in Fig. 6.

In Fig. 6, multiple DRL agents interact with different environments to perform the same tasks. Their main objective is to achieve an optimal policy in their environments. A federated server aggregates the different models from different agents to obtain a general optimal policy. The basic steps of HFDRL can be described as following [61]:

- **Step 1:** All agents train their own DRL model, locally and independently, interacting with their environments.

- **Step 2:** The DRL agents send their masked model parameters to the federated server.
- **Step 3:** The federated server encrypts and aggregates the model parameters to obtain the global model M_{FD}.
- **Step 4:** The federated server sends the global model parameters to all DRL agents.
- **Step 5:** The DRL agents update their local model with the global model parameters and continue to perform tasks.

2. VFDRL

VFDRL comes into action when multiple agents are interacting with the same environment to perform different tasks. To cooperate with each other, the agents have to learn the actions taken by other agents. The main goals of the VFDRL are to train the DRL agents more effectively from the same environment and make the agents more robust. In VFDRL, the agents can share their masked model parameters but can not share their raw data obtained from the same environment. A basic VFDRL framework is shown in Fig. 7.

In Fig. 7, multiple DRL agents are interacting with the same environments to perform different tasks. Their main objective is to achieve an optimal policy for their assigned task in the same environment. A federated server aggregates the different models from different agents to obtain a general optimal policy and relays one agent's masked model parameters to others. The basic steps of VFDRL can be described as following [61]:

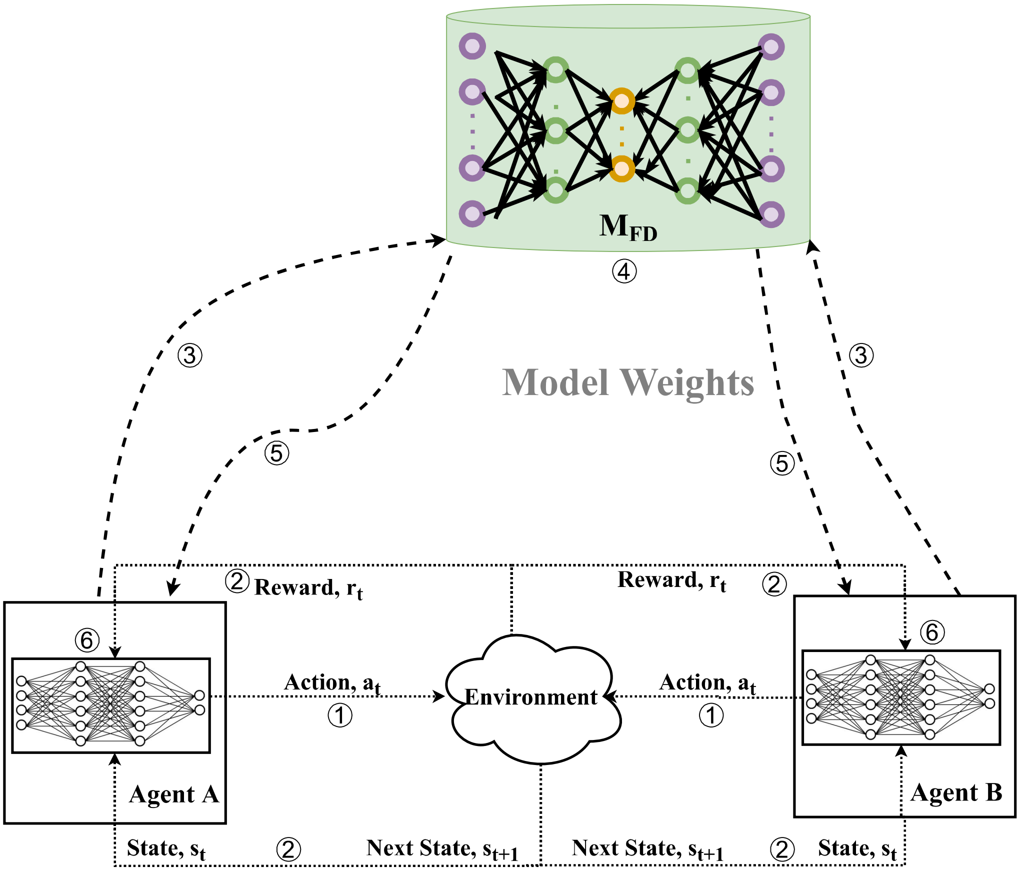


Figure 7: VFDRL Framework.

- **Step 1:** All agents train their own DRL model, locally and independently, interacting with the same environments.

- **Step 2:** The DRL agents get their feedback from the environment, such as states and rewards.

- **Step 3:** The DRL agents compute the mid-products and send the masked mid-product model parameters to the federated server.

- **Step 4:** The federated server encrypts and aggregates the mid-product model parameters to obtain the global model M_{FD}.
- **Step 5:** The federated server sends the global model parameters to all DRL agents.
- **Step 6:** Each DRL agent updates their local model with the global model parameters and continues to perform tasks.

III. Priority-based Joint Resource Allocation with DQL

In this chapter, we describe the proposed resource allocation scheme to maximize the performance of multi-carrier NOMA systems under multiple performance metrics. To describe the proposed solution, we first derive the basic architecture of multi-carrier NOMA and the system model and then formulate the constraints of NOMA into a single RL problem.

A. Problem Statement

In this section, we discuss the fundamentals of multi-carrier NOMA. We also briefly describe the system model and derive different equations based on the constraints of NOMA system and the objectives of our proposed solution.

1. Multi-Carrier NOMA

With NOMA, multiple devices can be served using the same RRB utilizing the power domain for both uplink and downlink transmissions. We consider a simple downlink multi-carrier NOMA system where the BS serves different types of devices at the same time over the wireless channels. Fig. 8 shows, a scenario of 5G network consisting of three different devices. The BS assigns one channel to every three devices, where the signals of the three devices are multiplexed at different power levels. Therefore, the devices receive their desire signals along with the signals of other two devices of that channel as noise or interference. The unwanted signals will act as noise if the power level of the desired signal is high; otherwise the unwanted signals will act as interference. To decode the desired signal, each device uses SIC technology. SIC decodes the signal with the

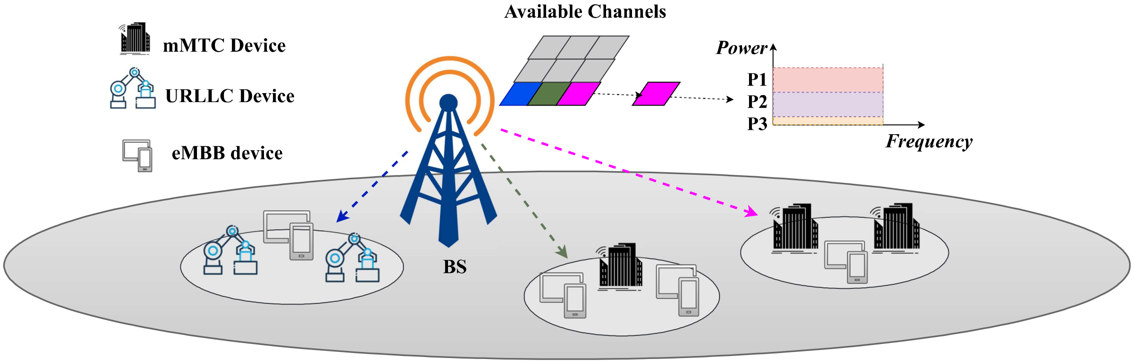


Figure 8: Simple multi-carrier NOMA system.

highest power and subtracts that signal from the main signal until it decodes the desire signal. The perfect SIC depends on the channel state information (CSI) such as signal-to-noise and interference ratio (SINR) [62], and the SINR depends on the channel assignment and power allocation. In this case, the data rate for each device for its channel can be calculated using (2).

$$R_i^k = \log_2\left(1+\frac{P_i^k\Gamma_i^k}{\sum_{j=1}^{i-1}P_j^k\Gamma_i^k+1}\right), \quad k,i=1,2,3, \tag{2}$$

where Γ is the channel to noise ratio (CNR) for the assigned channel k and P is the assigned power. The details of (2) are given in Sub-section 2.   .

2. System Model

We consider a micro-cell of a 5G network consisting of 5G enabled devices with a base station (BS). We also consider the downlink of single-input and single-output (SISO) NOMA system as shown in Fig. 9, where the total number of devices is N and the number of channels is K. There are three types of devices that require three different services of 5G network: eMBB devices $UE_1, UE_2, \dots, UE_e$; URLLC devices $UL_1, UL_2, \dots, UL_l$; and mMTC devices

$MC_1, MC_2, \ldots, MC_m$. We also consider that the total available bandwidth (BW_t) is divided into all channels having channel bandwidth (BW_{ch}) of 180 kHz. The maximum number of devices per channel is n, which ranges from $2 \leq n \leq N$, and the total number of channels is $K = ceil(N/n)$.

We consider perfect CSI to develop the proposed scheme. However, for a practical wireless environment, we also consider an imperfect CSI to evaluate the proposed scheme. Let us assume that the k^{th} channel is assigned to n devices, where the power allocated to the n^{th} device is P_n and the desired signal of the n^{th} device is x_n. After combining the signals of the n devices, the BS transmits them over the k^{th} channel which can be represented as follows:

$$X^k = \sum_{i=1}^{n} \sqrt{P_i^k} x_i, \quad i = 1, 2, \ldots, n \tag{3}$$

At the device end, the transmitted signal reaches with path loss component and additive white Gaussian noise (AWGN), which can be represented as

$$y^k = \sum_{i=1}^{n} \sqrt{P_i^k} h_i^k x_i + w^k, \quad i = 1, 2, \ldots, n, \tag{4}$$

where h_i^k is the channel gain of the i^{th} device and w^k denotes the AWGN with thermal noise power variance, σ_k. After receiving the signal, the receiver uses the SIC technique to decode its signal. Perfect SIC depends on the SINR of the device on the channel that it has been using for communication. Let us consider the CNR of the n^{th} device for k^{th} channel is

$$\Gamma_n^k = \frac{\mid h_i \mid^2}{\sigma_k}. \tag{5}$$

We know from the earlier discussion that different power levels are allocated to the devices of a channel. As per NOMA, the highest power is allocated

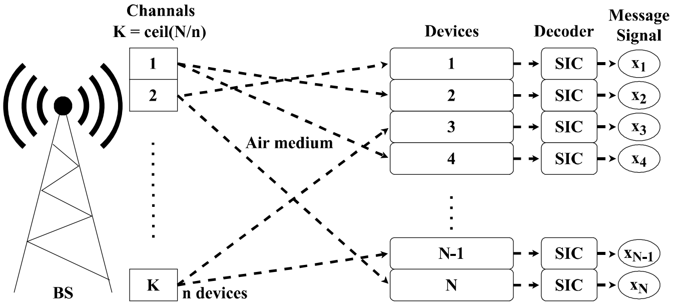


Figure 9: System architecture of multi-carrier SISO-NOMA system.

to the device with the lowest CNR and vice versa. For example, for devices having $\Gamma_1^k > \Gamma_2^k > \ldots > \Gamma_n^k$ CNR are assigned with power $P_1^k < P_2^k < \ldots < P_n^k$, respectively. Therefore, the SINR and the data rate for each device of a specific channel can represented as (6) and (2), respectively.

$$\gamma_i^k = \frac{P_i^k \Gamma_i^k}{\sum_{j=1}^{i-1} P_j^k \Gamma_i^k + 1}, \quad i = 1, 2, \ldots, n. \tag{6}$$

To perform perfect SIC, the BS allocate power to each device above certain threshold level P_{th} as shown in (7). For example, the device with low CNR must have higher power than the sum of other high CNR devices' power for perfect completion of the SIC technique.

$$\begin{aligned} \left(P_i^k - \left(\sum_{j=1}^{i-1} P_j^k \right) \right) \Gamma_d^k \geq P_{th}, \\ i = 1, 2, \ldots, (n-1), \\ d = n, \ldots, 2, 1, \\ k = 1, 2, \ldots, K. \end{aligned} \tag{7}$$

3. Problem Formulation

We consider each device has a set of channels $\Gamma_N = \{\Gamma_N^1, \Gamma_N^2, \ldots, \Gamma_N^k\}$ for channel assignment and range of power from $P_N \in [0.01, 0.99] \times P_T$ where P_T is the total power budget per channel for power allocation. In this study, we focus on the sum-rate as the key performance indicator for the optimization of channel assignment and power allocation in the NOMA system which can be represented as

$$\begin{aligned} R_{\text{sum}} = \sum_{1}^{K} \sum_{i=1}^{n} \log_2 \left(1 + \frac{P_i^k \Gamma_i^k}{\sum_{j=1}^{i-1} P_j^k \Gamma_i^k + 1} \right), \\ i = 1, 2, \ldots, n, \\ k = 1, 2, \ldots, K. \end{aligned} \tag{8}$$

We also consider the minimum data rate requirement of all devices which can be expressed as

$$\begin{aligned} \log_2 \left(1 + \frac{P_i^k \Gamma_i^k}{\sum_{j=1}^{i-1} P_j^k \Gamma_i^k + 1} \right) \geq R_i^k, \\ i = 1, 2, \ldots, n, \\ k = 1, 2, \ldots, K. \end{aligned} \tag{9}$$

The sum of the power per device in a channel must less or equal than P_T, and can be written as

$$\sum_{i=1}^{n} P_i^k \leq P_{\mathrm{T}}, k = 1, 2, \ldots, K. \tag{10}$$

In this study, we derive an optimal power allocation scheme and propose a priority-based channel assignment with a deep Q-learning algorithm for maintaining the QoS policies of the 5G services, MSR, and MCSR to ensure fairness among the devices and the increase in system performance. As DQL requires power allocation to evaluate the channel assignment and train the DNN, we first derive a power allocation solution for any given channels, and then we build the DQL framework for priority-based channel assignment to obtain an optimal solution for the NOMA system.

B. Power Allocation

In this section, we derive the optimal power allocation for any given channel while considering different constraints of NOMA to increase the maximum sum-rates and system efficiency. The power allocation solution is derived based on the power allocation solution in [25]. We consider sorting the devices in descending order based on their distances from BS. As our main target is to maximize the sum-rates, we can represent (8) as a maximizing convex function for a given channel k considering (7), (9), and (10), which can be formulated as follows:

$$\begin{aligned}
\underset{P_i^k}{\text{maximize}} \quad & \sum_{i=1}^{n} \log_2 \left(1 + \frac{P_i^k \Gamma_i^k}{\sum_{j=1}^{i-1} P_j^k \Gamma_i^k + 1} \right) \\
\text{subject to} \quad & \log_2 \left(1 + \frac{P_i^k \Gamma_i^k}{\sum_{j=1}^{i-1} P_j^k \Gamma_i^k + 1} \right) \geq R_i^k, \\
& \sum_{i=1}^{n} P_i^k \leq P_{\mathrm{T}}, \\
& \left(P_i^k - \left(\sum_{j=1}^{i-1} P_j^k \right) \right) \Gamma_d^k \geq P_{th}, \\
& \forall i = 1, 2, \ldots, n; d = n, \ldots, 2, 1.
\end{aligned} \tag{11}$$

The convex problem (11) can also be expressed in Lagrangian form as

$$
\begin{aligned}
& L(P, \tau, \nu, \psi) \\
& = \sum_{i=1}^{n} \log_2 \left(1 + \frac{P_i^k \Gamma_i^k}{\sum_{j=1}^{i-1} P_j^k \Gamma_i^k + 1}\right) \\
& = \tau^k \left(P_T - \sum_{i=1}^{n} P_i^k\right) \\
& + \sum_{i=1}^{n} \nu_i^k \left\{P_i^k \Gamma_i^k - \left(\sum_{j=1}^{i-1} P_j^k \Gamma_i^k - 1\right) \times \left(\phi_i^k - 1\right)\right\} \\
& + \sum_{i=2}^{n} \psi_i^k \left(P_i^k \Gamma_d^k - \sum_{l=1}^{i} P_l^k \Gamma_d^k - P_{Th}\right),
\end{aligned}
\tag{12}
$$

where τ, ν, and ψ are the Lagrange multipliers, $\forall i = 1, 2, \ldots, n$, and $\phi_i^k = 2^{\frac{R_i^k}{KBW_{ch}}}$. Taking the derivatives of (12) with respect to P_i, τ, ν, and ψ, multiple KKT conditions can be found. For n-device NOMA, there are $2n$ Lagrange multipliers resulting in 2^{2n} combinations. For example, for $n = 2, 3, 4, \ldots, 8$, the number of combinations are $16, 64, 256, \ldots, 65536$, respectively. However, checking all types of combinations is not computationally feasible. After solving only n equations according to [63] for $2, 3, 4$-device NOMA, $2, 4, 8$ combinations are found that satisfy the KKT conditions, respectively. Therefore, the closed-form solution of the power allocation for n-device NOMA for a given channel k is near-optimal and can be written as

$$
\begin{aligned}
P_x &= \frac{P_T}{2^{(n-1)}} + \frac{(x-1)P_{th}}{2^{(x-1)}\Gamma_{(x-1)}} - \left(\sum_{i=x}^{n-1} \frac{P_{th}}{2^i \Gamma_i}\right), \\
P_j &= \frac{P_T}{2^{(n-q-2)}} + \frac{P_{th}}{2\Gamma_{(j-1)}} - \left(\sum_{i=j}^{n-1} \frac{P_{th}}{2\Gamma_i}\right),
\end{aligned}
\tag{13}
$$

where $x = 1, 2$, $j = 3, 4, \ldots, n$, $q = 0, 1, \ldots, (n-3)$, and devices have $\Gamma_1^k > \Gamma_2^k > \ldots > \Gamma_n^k$ CNR with power $P_1^k < P_2^k < \ldots < P_n^k$, respectively.

C. Priority-based Channel Assignment

In this section, we propose a priority-based channel assignment scheme using deep Q-learning. First, we formulate the channel assignment problem based on the priority, MSR, and MCSR, and then model the channel assignment problem as a reinforcement task and introduce an autoencoder followed by an LSTM network to create the DQL framework. Finally, we use the near-optimal power allocation solution and train the DNN for validation.

1. Priority-based Channel Assignment

The 5G wireless network provides three different services with different QoS requirements, such as URLLC service has highest QoS requirements, eMBB service has average QoS requirements, and mMTC service has least QoS requirements. We prioritize the devices in the network based on the services they are using and their QoS requirements where the URLLC devices have the highest priority, the eMBB devices have the second-highest priority and the mMTC devices are the least priority devices. The BS sorts the URLLC, eMBB, and mMTC devices in descending order based on their distances from BS. Subsequently, the BS assigns URLLC devices to the channels with highest gain first, then assigns the eMBB devices and mMTC devices accordingly to the channels available as shown in Fig. 10. This figure shows an illustration of priority-based channel assignment for 3-device NOMA where 4 URLLC, 5 eMBB, and 3 mMTC devices are present. However, assigning channels is subject to the CNR of each device with the BS.

Another main requirement of the optimization of the channel assignment is to maximize the channel and overall sum-rates. The BS have $\binom{N}{n}$ combinations

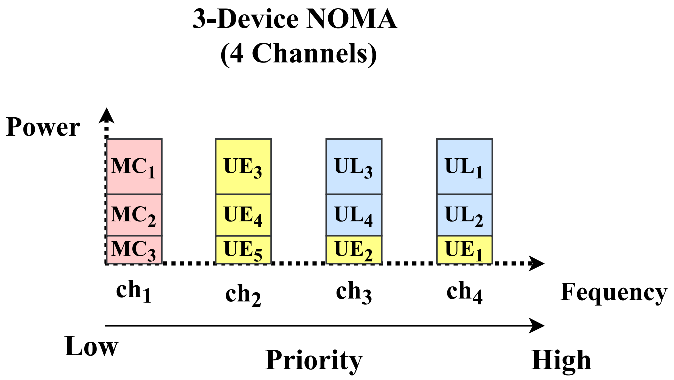


Figure 10: Proposed priority-based sample channel assignment for 3-device NOMA system for 12 active devices.

for each channel k to check for maximize the sum-rate. Therefore, the total combination in general is $\sum_{i=1}^{K} \binom{N-(n\times i)}{n}$ for MCSR. When it comes to priority, the low priority devices can not replace the high priority devices in a channel. However, high or equal priority devices can replace the equal or low priority devices in any given channel. The maximization process incorporating with the priority scheme is computationally complex since the BS has to check all the possible combinations of the device. To reduce the computational complexity, we propose a DQL framework to assign channels to the devices while maintaining the priority and maximizing the sum-rates.

2. Deep Q-Learning Framework

In this section, we propose a DQL framework and train it to optimize the priority-based channel assignment problem. The deep Q-learning algorithm generally consists of an agent with a deep neural network (DNN) and an environment. The agent interacts with the environment and decides which action to take. The BS acts as an agent and interacts with the environment consisting of URLLC,

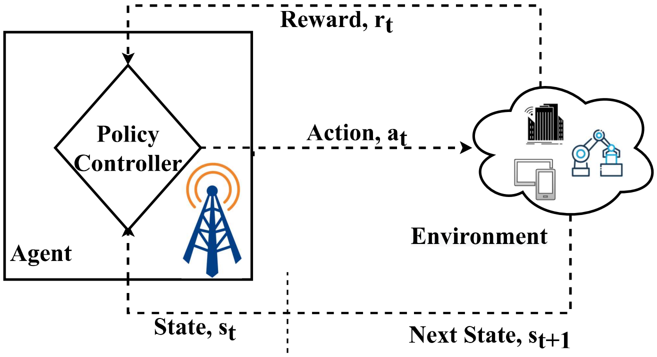


Figure 11: Simple Q-learning.

eMBB, and mMTC devices' information. Initially, the agent starts exploring the environment to collect the channel information of every device. At each time step t, based on the present state s_t of the agent in the environment, the agent predicts an action a_t using the DNN to assign a channel. In return, the agent receives an immediate reward r_t and the next state s_{t+1} from the environment as shown in Fig. 11. The agent receives a good reward r_t if it performs a good channel assignment. By predicting actions, the agent learns about the environment and achieves an optimal channel assignment policy π_c. This optimal policy is learned at each time step t by the DNN. The agent updates and improves the policy π_c by repeating the channel assignment process for multiple episodes. One episode terminates when there are no channels left for assignment. We define the state, action, and reward for use in the proposed DNN as follows:

- **State**: We consider the channel information for each device as the states of the environment. There are N devices having K channel preferences. Therefore, the state space has $N \times K$ elements and can be represented as $S = \{\Gamma_1^1, \Gamma_1^2, \Gamma_1^3, \ldots, \Gamma_1^K, \Gamma_2^K, \Gamma_3^K, \ldots, \Gamma_N^K\}$.

- **Action**: The main action of the agent is to assign channels to the devices which belong to the action space A. At each episode for a set of S, the agent has to take $N \in A$ actions while maintaining one action per K elements from S. For $2,3,\ldots,n$-device NOMA, the agent can take one action $2,3,\ldots,n$-times, respectively.

- **Reward**: Whenever the agent completes taking N actions, the agent gets a reward r_t^l for each action. For each correct action, the agent gets a positive reward r_i and when the agent takes correct n actions, the agent gets the sum-rate of that channel as a reward for the taken actions. For example, let us assume a 3-device NOMA. The agent has to assign 3 devices per channel. In this case, when the agent successfully selects an appropriate channel based on priority for a device, the agent gets a positive reward r_i (i.e., 10). If the agent can select the same appropriate channel for 3 devices, the agent gets the sum-rate calculated by (2) as a reward for its 3 actions. The reward function can be defined as

$$r_t^l = \begin{cases} \sum_{i=1}^{n} R_i^k & \text{if } a_p^k = n \\ 0 < r_i < \sum_{i=1}^{n} R_i^k \text{ for each } a_t^l & \text{if } a_p^k < n \\ 0 & \text{if } a_p^k = 0 \end{cases}, \tag{14}$$

where a_p^k is the number of appropriate action a_t^l taken per channel k and $\forall l = 1,2,\ldots,N \in A$. Here, we consider maximizing the sum-rate for each channel which results in increased performance and fairness of the whole system.

With the state, action, and reward, we propose the deep neural network (DNN) structure shown in Fig. 12 as the policy controller for channel assignment. The DNN replaces the Q-table and estimates the Q-values for each state-action pair

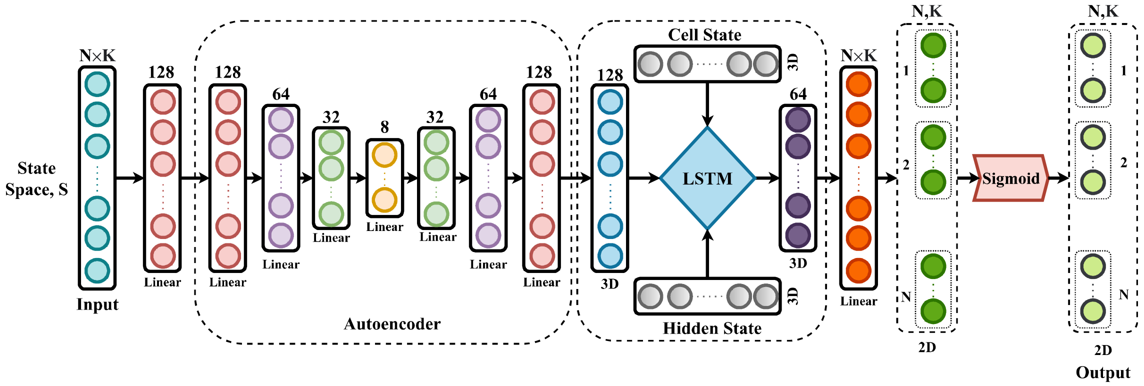


Figure 12: Proposed DNN structure.

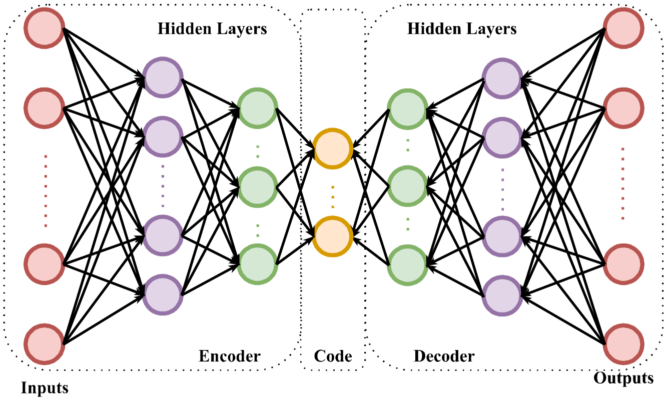


Figure 13: Autoencoder architecture.

of the environment. Eventually, the DNN approximates the optimal policy for channel assignment. The proposed DNN has two parts, an autoencoder model and an LSTM model. The main goal of the DNN is to derive probabilities for each device-channel pair for each state space, which can be expressed as $Q(S,A)$. These probabilities are the Q-values for DQL.

- **Autoencoder:** An autoencoder is a feed-forward neural network where the number of inputs is same as the number of output neurons. It compresses the input into a lower-dimensional code and then reconstructs the input data from the code at the output. The autoencoder can easily handle raw input data without any fancy processing or labeling. Therefore, the autoencoder is considered as a part of the unsupervised learning technique [64] and can generate their labels from the training data. The autoencoder has three main parts named an encoder, code, and decoder as shown in Fig. 13. Both the encoder and decoder are fully connected neural networks. The encoder starts with an input layer having 2^n neurons followed by multiple hidden layers having 2^{n-h} neurons, where h is the position of the layer. The number of neurons per hidden layer continues to decrease till the code part of the autoencoder. In this study, we use 2^3 neurons for the code layer. The decoder part is the mirror image of the encoder ending with an output layer. This type of structure is known as *stacked autoencoder* as the layers are stacked one after another, like a sandwich. Moreover, we use ReLU as an activation function for each layer in the autoencoder.

- **Long short-term memory:**Long short-term memory (LSTM) is an evolved form of recurrent neural network (RNN). LSTMs are a special type of RNN that can learn long-term dependencies and remember previous information for future usage. The LSTM network has a chain structure composed of multiple LSTM cells. We use three LSTM cells to build our LSTM network. The structure of a single LSTM cell is shown in Fig. 14 [65]. An LSTM cell has three input and two output parameters. The cell and hidden states are the common parameters between inputs and outputs. The other parameter is the current input. The LSTM cell also contains three sigmoid layers and two tanh layers involving some linear transformations as shown in Fig. 14. Initially, random cell and hidden states are given along

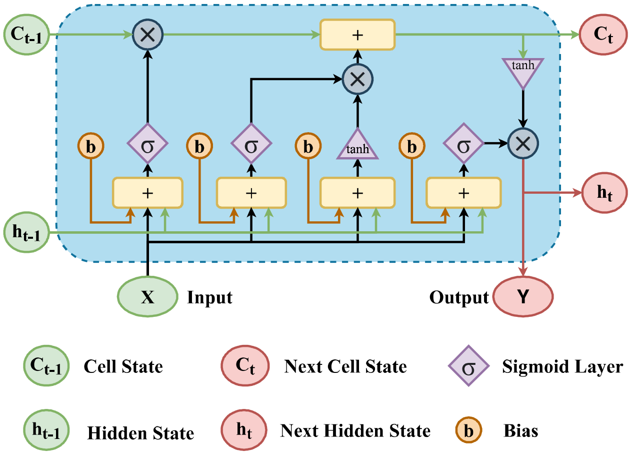


Figure 14: An LSTM cell.

with the input for the first LSTM cell. Then the two outputs (hidden state, cell state) become the three inputs of the next cell as shown in Fig. 14.

In this study, we use an autoencoder having input and output size of 128 and code size 8 followed by an LSTM network having 128 input size, 64 hidden state size, and 3 recurrent layers. Finally, the output of the LSTM is passed through a linear layer and a sigmoid layer to obtain the probabilities of the preferred channels for each device. The state space S is given as the input of our policy network. Initially, the input is first embedded with dimension 128. It then passes through the policy network to generate the channel assigning probabilities, as shown in Fig. 12.

3. Training

The proposed DNN is trained gradually with a set of training data $T_{data} = \{S^1, S^2, \ldots, S^{ins}\}$ per episode. For each state space S, the device-channel pairs are selected using ε-greedy policy according to the output probabilities from the

DNN. An episode terminates when all state spaces are passed through the DNN. The policy to take action for each device per state space can be expressed as

$$a_i^l = \begin{cases} \text{argmax } Q(S^i, A_i^l) & \text{if } \varepsilon < \ \varepsilon_{th}; \text{ where } \varepsilon_{th} \in (0,1] \\ \text{random action } [1, K] & \text{otherwise} \end{cases}, \qquad (15)$$
$$\forall l = 1, 2, \ldots, N \in A,$$
$$\forall i = 1, 2, \ldots, ins.$$

After taking the actions using (15), the agent gets the rewards according to (14) and the next state space S^{i+1}.

To train the DNN, we calculate the loss and optimize the parameters of the DNN performing back-propagation. To calculate the loss, we approximate the optimal Q^*-values for each device-channel pair of S^{i+1} from a different DNN called the target DNN [66]. The target DNN is identical to the policy DNN and initialized by the parameters of the policy DNN. The next state space S^{i+1} is given as an input to the target DNN and from the outputs the optimal Q^*-values are chosen greedily by the agent. Because assigning the channel is a classification problem, we use the categorical cross-entropy loss function to calculate the loss between the optimal Q^*-values and normal Q-values [67]. After calculating the loss, we optimize the policy DNN using the Adam optimizer [68]. To estimate the optimal Q^*-values correctly, we periodically update the target DNN with the parameters of the policy DNN after certain episodes.

For a more stable convergence of the optimal policy, we introduce the experience replay memory (ERM) to the DQL [69]. Initially, the agent explores the environment and saves current states, actions, rewards, and next states (S^i, A_i, r_i, S^{i+1}) as a tuple in the ERM. Subsequently, the agent takes a mini-batch of tuples from the ERM and trains the policy DNN. The ERM continues to be

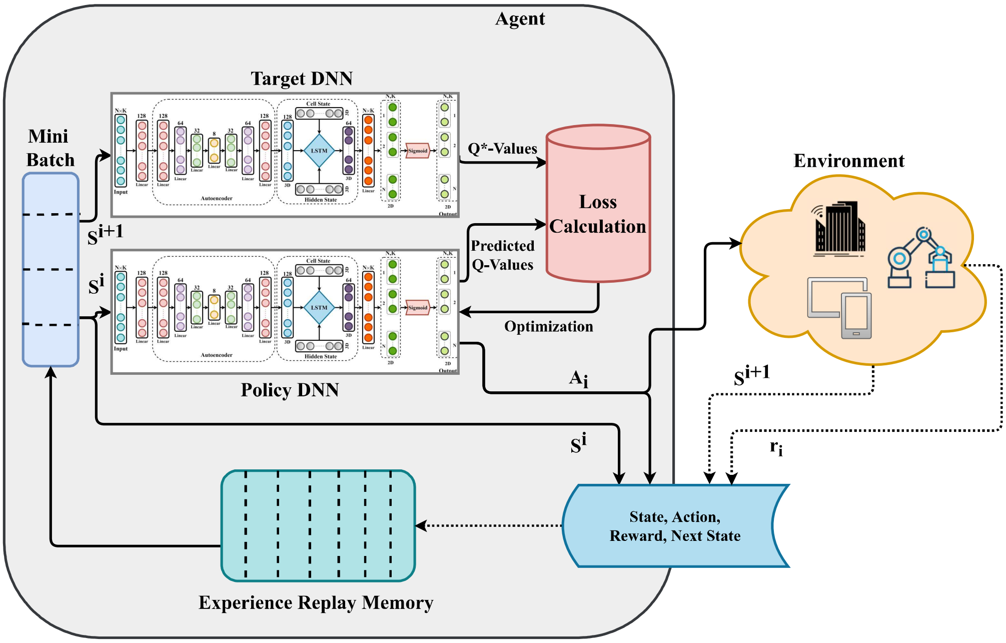


Figure 15: Proposed DQL framework.

updated for each training data. Fig. 15 and Algorithm 3 summarize the proposed DQL framework and the working flow.

D. Simulation Analysis

In this section, we perform multiple simulations to analyze the performance of the proposed DQL algorithm for priority-based channel assignment and compare the proposed priority-based joint resource allocation (priority-JRA) with the joint resource allocation (JRA) method and dynamic power allocation with fixed channels (DPA-FC) method proposed in [30] and [25], respectively. Moreover, we compare the priority-JRA NOMA system with the conventional OMA system. Finally, we also analyze the system complexity and system convergence varying different parameters.

Algorithm 3 Proposed DQL Algorithm

```
Initialize policy and target DNN with random parameters ($p$ and $p'$).
Initialize experience replay memory (ERM).
Initialize $\varepsilon$.
for each episode do
    for each instance do
        for each device do
            Select an channel and add to action space $A_i$ for present state space $S^i$ based on $\varepsilon$.
        end for
        Observe the immediate rewards $r_i$ and next state space $S^{i+1}$.
        Insert $(S^i, A_i, r_t, S^{i+1})$ in ERM.
        Create a mini-batch with random sample of $(S^i, A_i, r_t, S^{i+1})$ from ERM.
        for each tuple in mini-batch do
            Obtain $Q$-values using policy DNN.
            Approximate $Q^*$-values using target DNN.
            Calculate the loss using $Q$ an $Q^*$-values.
            Optimize the parameters $p$ of the policy DNN using Adam optimizer.
        end for
    end for
    $p' \leftarrow p$ after certain number of episodes.
end for=0
```

Table 1: Simulation parameters for DQL-based resource allocation

Parameter	Value	Parameter	Value
BW_t	5 MHz	BW_{ch}	180 kHz
P_{T}	$2-18$ W	Learning rate	0.01
N	24	Batch size	24
n	$2,3,4$	Circuit power	20 dBm
K	$12,8,6$	Number of episodes	200
T_{data}	5000 instances	R_0	2 bps/Hz

1. Simulation Environment

For the simulation environment, we consider a 5G micro-cell where 24 devices are randomly and uniformly distributed. We only consider three types of devices, URLLC, eMBB, and mMTC devices. We model the channel gain h_i^k of the k^{th} channel for each device based on the Rayleigh fading model, where the path loss exponent, $\eta = 3$. Then we calculate the CNR of each channel for each device using (5) where $\sigma_k = \frac{BW_t \times N_0}{k}$ for $\forall k = 1,2,\ldots,K$ with BW_t = 5MHz and N_0 = -172 dBm/Hz.

To analyze the performance, simulation parameters similar to [25], [30] are used as given in Table 1. The parameters of proposed DNN such as weights and biases are initialized randomly and uniformly. The input size of the DNN is $N \times K$ and the embedded size is 128. We generate 5000 instances for training and 1000 instances for validation data-set randomly for each episode. Each instance consists of $N \times K$ user-channel information.

2. Performance Analysis

In this section, we compare the proposed priority-JRA with JRA and DPA-FC in terms of system sum-rate, sum-rate per channel, and energy-efficiency varying power, number of users, and location.

Fig. 16 shows the sum-rate versus the BS power comparison among priority-JRA, JRA, DPA-FC 3-device NOMA system. It is also evident from the figure that the proposed scheme outperforms the other two methods. In the JRA method, the power allocation solution is derived first, and the channels are then assigned using a matching algorithm [30]. By contrast, in the DPA-FC method, power allocation is done dynamically based on the channel response between the device and the BS while assigning fixed channels to the devices [25]. Hence, we can conclude that the priority-based channel assignment technique is more efficient than the JRA, and DPA-FC methods. From Fig. 16, we can also observe that the sum-rate is shown in bps/Hz which also reinforces the spectral efficiency of the system. Moreover, due to the converging nature of (8), the graph saturates when the BS power is extremely large.

Sum-rate for each channel comparison among priority-JRA, JRA, and DPA-FC for 3-device NOMA is shown in Fig. 17. It is evident from the figure that the proposed priority-JRA achieves the highest sum-rate in most of the channels while maintaining the proposed priority scheme. In few channels, the sum-rate is low because of the trade-off between the priority scheme and the maximum sum-rate. Our main target is to fulfill the QoS requirements of the 5G services while achieving the maximum possible sum-rate.

Fig. 18 shows the sum-rate achieved by the three schemes for the $2,3,4$-device NOMA system. For every NOMA system, the proposed priority-JRA

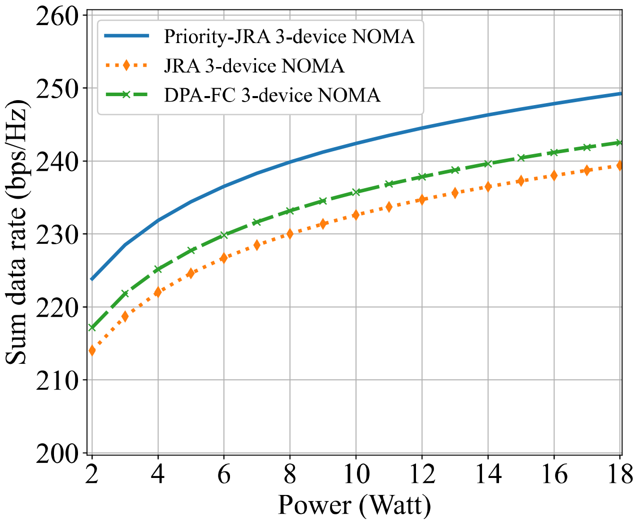


Figure 16: Sum-rate of 3-device NOMA system.

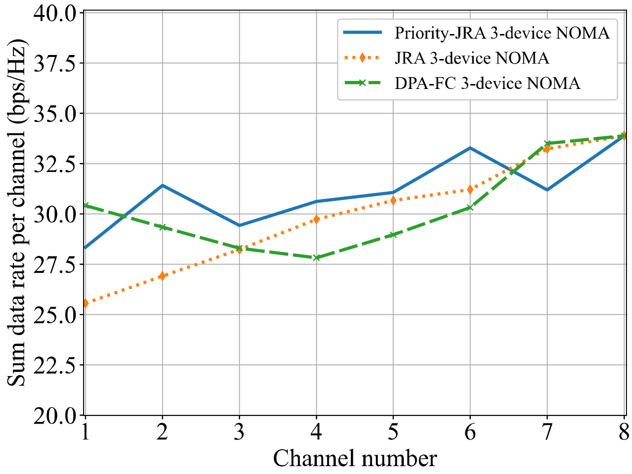


Figure 17: Sum-rate per channel of 3-device NOMA system where the channel number, $K = 8$.

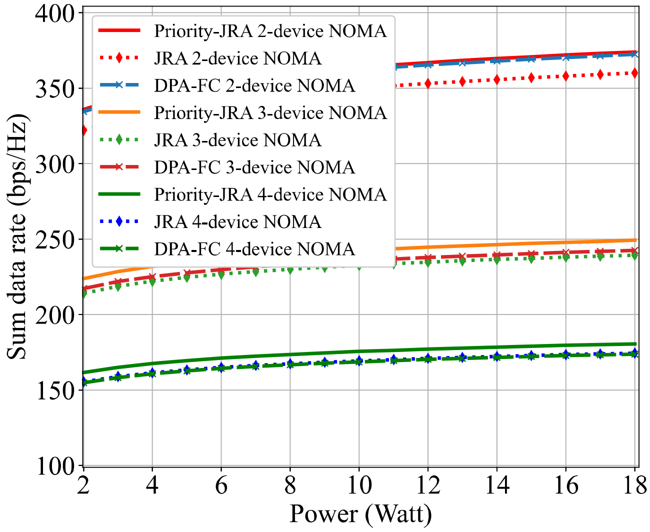
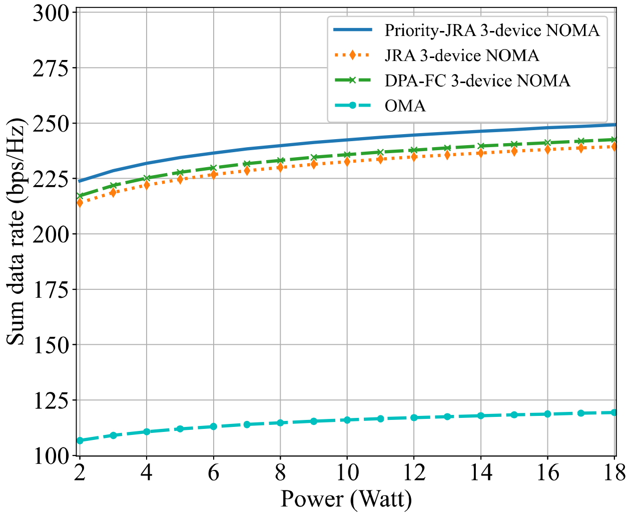


Figure 18: Sum-rate of 2, 3, 4-device NOMA systems.

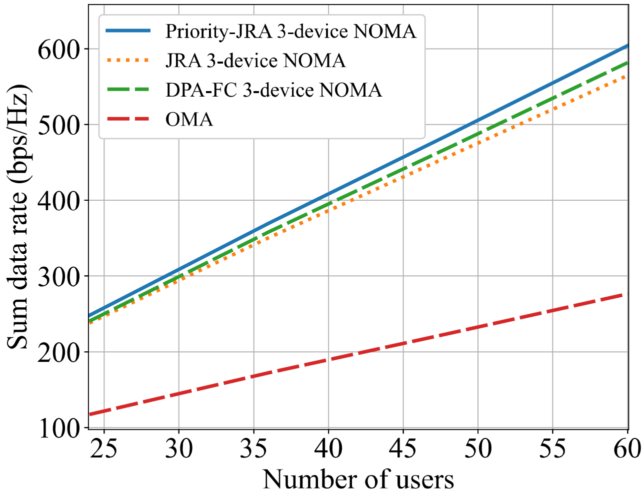
achieves the highest sum-rate compared to the other methods. Moreover, we can also observe that the sum-rate decreases when the number of devices per channel increases. This is due to the increase in system complexity and the division of the same amount of power into more devices.

In Fig. 19, we compare the conventional OMA system with priority-JRA along with JRA and DPA-FC NOMA systems in terms of the sum-rate with respect to power and number of users, respectively for the 3-device NOMA system. The sum-rate shown in the figure also represents the spectral efficiency of the system. It is clear that all NOMA systems outperform the traditional OMA system in terms of both the sum-rate and spectral efficiency. Moreover, we can also conclude from the Fig. 19 that the proposed priority-JRA outperforms all the other methods for any given power and number of users.

In Fig. 20, we compare the energy-efficiency of the OMA system with

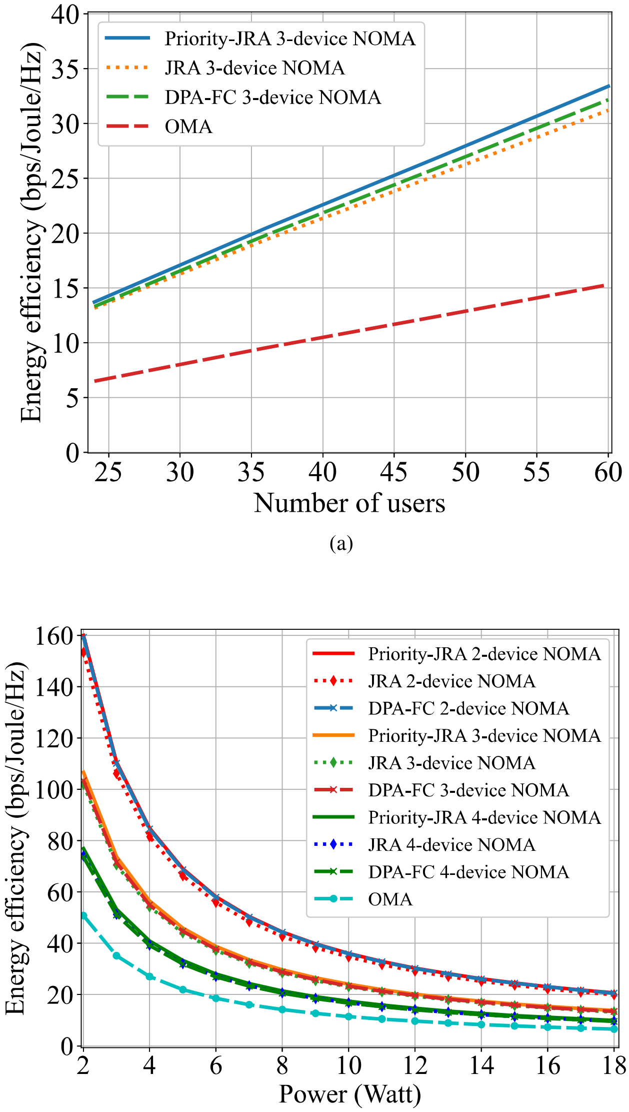


(a)



(b)

Figure 19: Sum-rate of 3-device NOMA system and OMA system with respect to (a) power and (b) number of users.



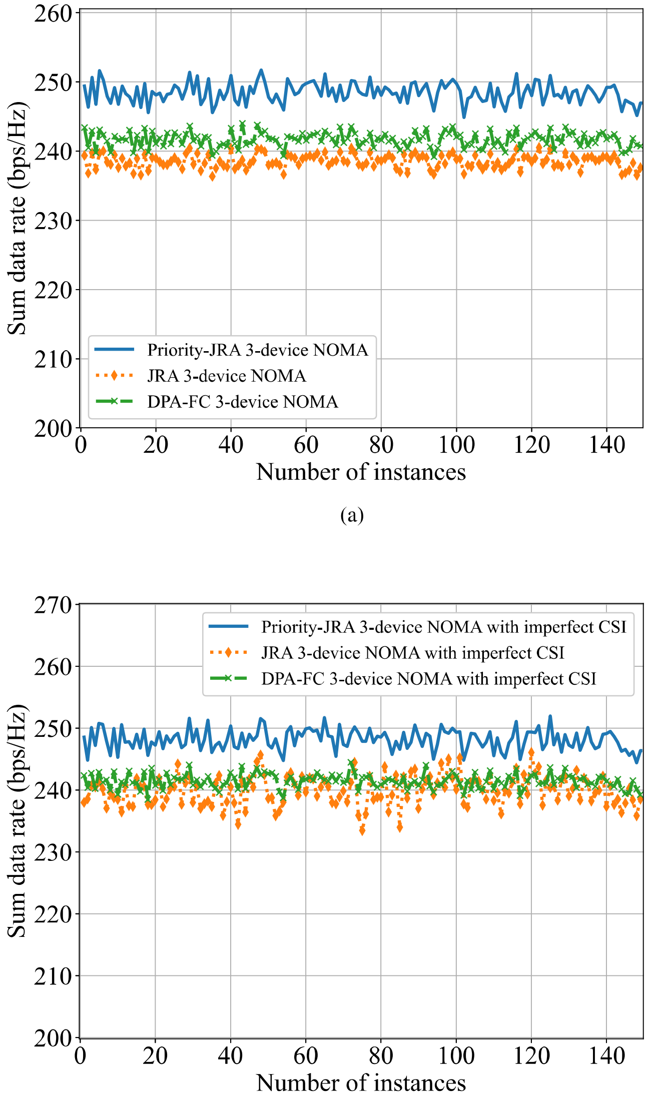
(a)

(b)

Figure 20: Energy-efficiency of (a) 3-device NOMA system and OMA system with respect to number of users and (b) 2, 3,and 4-device NOMA systems and OMA system with respect to power.

different methods of the NOMA system with respect to number of users and power, respectively. Energy-efficiency of a system represents the number of sent bits per joule of energy. The graph shows that the energy-efficiency decreases as the power increases because the energy efficiency is inversely proportional to power. We can conclude from the figure that the NOMA system is more energy-efficient than the conventional OMA system in any scenario. Moreover, from Fig. 20, we can also observe that the proposed priority-JRA is the most energy-efficient method for channel assignment among all for any given power and number of users. We calculated the energy efficiency graph using the BS power and circuit power for each method [30].

Moreover, Fig. 21 shows the sum-rate comparison among priority-JRA, JRA, DPA-FC 3-device NOMA system for different user-data instances considering perfect and imperfect CSI. As mentioned earlier, we generate 5000 and 1000 instances consisting of $N \times K$ user-channel information per instance for training and testing the proposed priority-JRA scheme, respectively. In every instance, the positions of the users are randomly and uniformly generated within the transmission range of the BS. From Fig. 21a, it is evident that the proposed priority-JRA achieves the highest sum-rate for any given positions of the users. By contrast, we consider $\pm 30\%$ CSI error to evaluate the performance of the aforementioned systems in Fig. 21b. It is noticeable from Fig. 21b that the performance of the proposed priority-JRA remains almost unchanged compared to the JRA, DPA-FC schemes.



(a)

(b)

Figure 21: Sum-rate of 3-device NOMA systems for multiple validating instances considering (a) perfect and (b) imperfect CSI.

3. Complexity and Parameter Analysis

The proposed priority-JRA scheme contains a DNN network. To visualize the efficiency of the proposed DNN network, we derive and analyze the time complexity. The proposed DNN can be divided into three main elements for complexity analysis, which are an auto-encoder, an LSTM, and two linear layers as shown in Fig. 12.

The proposed DNN has an input of (NK) and two linear layers of size $d_e = 128$. The time complexity can be written as $O(2Id_e^2(NK))$, where I refers to the kernel size. The auto-encoder has one code layer and two identical encoder and decoder layers. According to [70], the time complexity of the auto-encoder can be written as

$$\begin{aligned} & O(2Id_e^2(1+\frac{1}{2}+\frac{1}{4}+\frac{1}{32})(NK)) \\ & = O(\frac{55}{16}Id_e^2(NK)) \\ & \simeq O(3Id_e^2(NK)) \end{aligned} \tag{16}$$

For the LSTM the time complexity can be calculated as $O(I)$. Therefore, the overall time complexity of the proposed DNN can be written as

$$\begin{aligned} & O(3Id_e^2(NK)) + O(2Id_e^2(NK)) + O(I) \\ & = O(5Id_e^2(NK)) + O(I) \end{aligned} \tag{17}$$

By contrast, for the JRA scheme, the time complexity can be calculated as $O((\frac{I^2-I}{2})\binom{N}{n}^2)$, which includes all $\binom{N}{n}$ combinations for each channel k. Therefore, the complexity of the priority-JRA is much lower. However, DPA-FC scheme has the lowest complexity and it does not outperform the priority-JRA scheme.

To justify our proposed DNN structure, we compare it with multiple DNN structures such as standard fully-connected DNN, only LSTM, and only

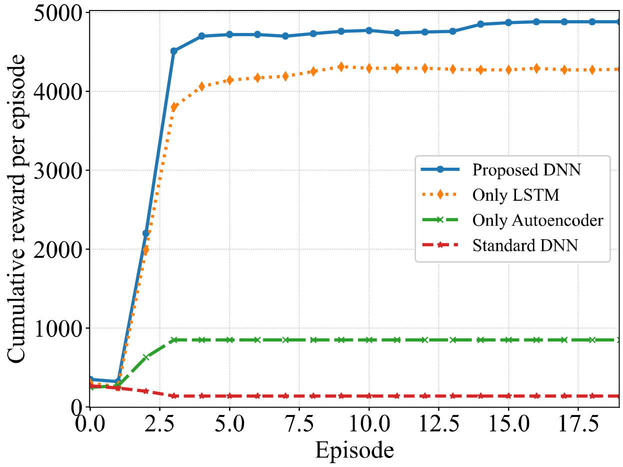
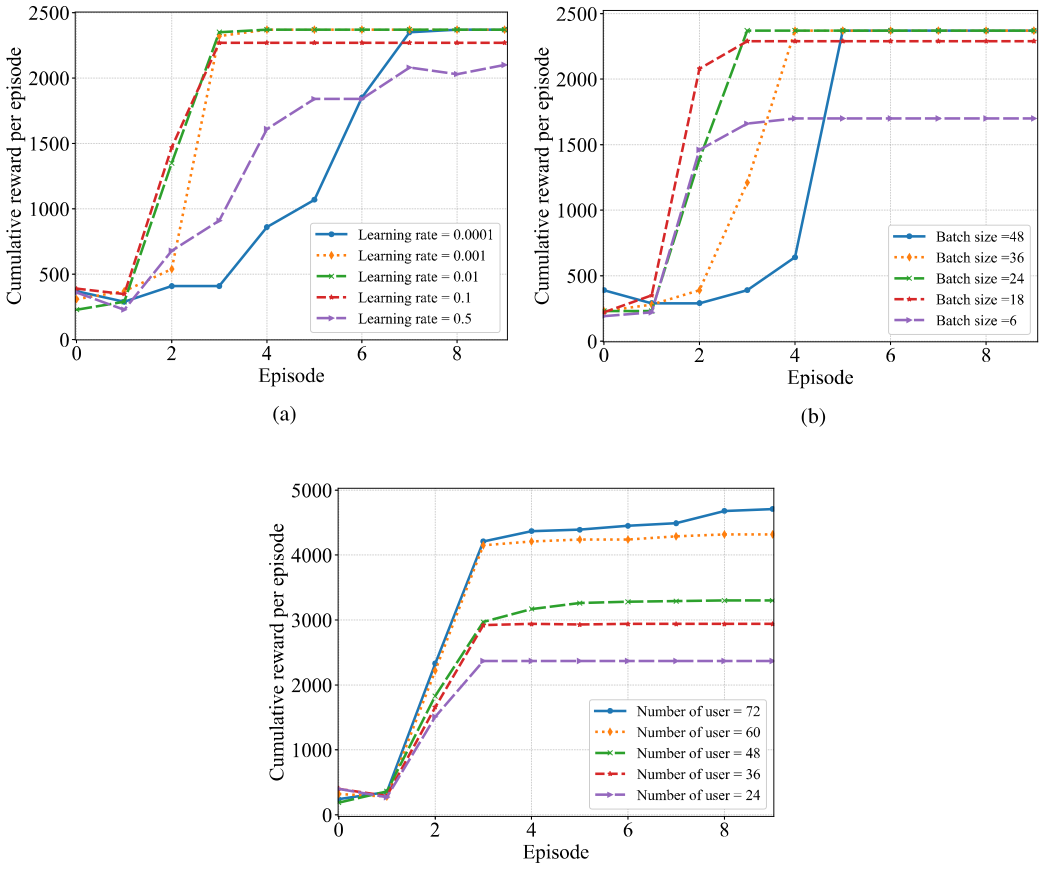


Figure 22: Channel assignment policy convergence for different DNN structures.

autoencoder in Fig. 22 for 72-devices and at a learning rate 0.01 and batch size of 24. It is evident from Fig. 22 that the proposed DNN structure achieves maximum cumulative reward and converges faster among all. Furthermore, Fig. 23a shows the effect of different learning rates on the proposed DNN for 24-devices and a batch size of 24. As shown in Fig. 23a, the proposed DNN cannot learn the optimal channel assignment policy for learning rates of $0.5, 0.1,$ and 0.001. However, for learning rates 0.01 and 0.001, the proposed DNN reached the optimal solution quickly in the same episode. Therefore, we can use any one of them. Fig. 23b shows the effect of different batch sizes on the proposed DNN for 24-devices and a learning rate of 0.01. As shown in Fig. 23b, the batch size should be greater than or equal to 24 to achieve optimality. However, a larger batch size refers to more room for exploration and slow convergence. Lastly, Fig. 23c represents the convergence of the proposed DNN for different number of users at a learning rate of 0.01 and batch size 24. The converging graphs of Fig. 23c



(a)

(b)

(c)

Figure 23: Channel assignment policy convergence for different (a) learning rate, (b) batch sizes, and (c) number of user.

signify the high scalability and stability of the proposed DNN for increasing number of users under the BS. Finally, we can ensure from the analysis that the proposed scheme can achieve a near-optimal performance with low complexity and high efficiency.

IV. FDRL-Based Multi-UAV Navigation

In this chapter, we describe the proposed scheme to improve the link quality and coverage of the 5G NOMA systems under multiple performance metrics. To describe the proposed solution, we derive the multiple UAV-assisted 5G NOMA system model and formulate the constraints of UAV navigation into a federated reinforcement learning problem in this chapter.

A. Problem Statement

In this section, we briefly describe the system model and derive different equations based on the objectives of our proposed solution.

1. System Model

We consider a macro-cell of a 5G network consisting of 5G enabled URLLC, mMTC, and eMBB devices with a base station (BS) in Fig. 24. We consider a simple downlink multi-carrier NOMA system where the BS serves the devices simultaneously over multiple wireless channels. We also consider $\{D_1, D_2, \ldots, D_m\}$ GDs far from the BS are suffering from packet loss, poor link quality, and low QoS fulfilling rate owing to the poor LOS, multi-path fading. Thus, the BS assigns $\{U_1, U_2, \ldots, U_n\}$ UAV-BSs to fly over that area and serve the m-GDs. Moreover, the BS assigns each UAV-BS to a cluster to serve the GDs and avoid interference and collision among UAV-BSs. The clusters are determined by the BS in advance, depending on the geographical locations. Thus, the number of the GDs and the total number of UAV-BSs depends on the number of clusters. The UAV-BSs are considered to be deployed at the center of the clusters by the BS. The clusters get updated every hour depending on the mobility of the GDs.

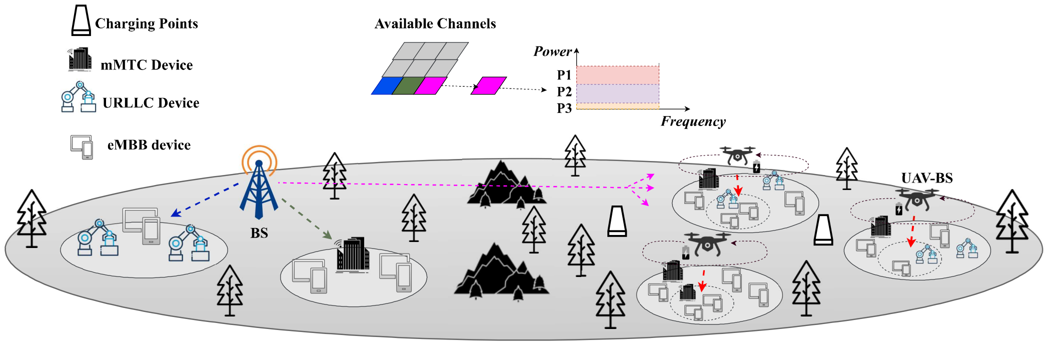


Figure 24: Multiple UAV-assisted heterogeneous 5G NOMA system scenario.

We consider that all UAV-BSs can move at any direction and with a certain distance in a $3D$ environment utilizing PID controller. We utilize spherical coordinate system where distance $d \in (0, d_{max}]$, azimuth $\phi \in (0, 2\pi]$, inclination $\theta \in (0, \pi]$. The UAV-BSs can also hover at a certain position where $d = 0$, $\phi = 0$, and $\theta = 0$. We also consider the energy budgets $\{E_1, E_2, \ldots, E_n\}$ of the UAV-BSs where the energy consumption e_n^t of n^{th} UAV-BS at every time-step gets deducted from its energy budget E_n. For UAV-BSs movement and hovering, the energy consumption $e_n^t = \beta d_n^t$ and $e_n^t = \beta$, respectively, where d_n^t is the flying distance and β can be found from the energy consumption model of a UAV. We also consider some charging points as shown in Fig. 24 where UAV-BSs visit autonomously utilizing PID controller if the energy level is below certain threshold e_{th}.

2. Problem Formulation

In this study, we focus on the channel-to-noise ratio (CNR), coverage score (CS), and residual energy (RE) as the key performance indicators for the optimization of UAV-BS navigation for heterogeneous 5G NOMA systems. In

NOMA systems, perfect SIC depends on the CNRs of the GDs as the power allocation is inversely proportional to the CNR of the GD. The GD with the highest CNR gets the lowest power and vice versa. Let us consider CNR of the m^{th} GD is

$$\Gamma_m = \frac{\mid h_m \mid^2}{\sigma}, \tag{18}$$

where σ is the thermal noise power variance and h_m is the channel gain of the m^{th} GD calculated using the Rayleigh fading model with a path loss exponent $\eta = 3$. The reason behind considering CNR as a performance parameter is to adjust the height of the UAV-BSs. We consider a high threshold value of CNR Γ_{th} that the UAV-BSs have to achieve for the GD residing at the edge of the communication range of the UAV-BSs.

We also consider the number of time-steps t_{cv} a GD was covered by a UAV-BS to calculate the coverage score. At each time-step t, we can calculate the coverage score for m^{th} GD under n^{th} UAV-BS using (19). After T time-step, we can calculate the CS for n^{th} UAV-BS using (20). We consider CS as a performance parameter is to ensure the maximum GD coverage by the UAV-BSs.

$$cd_n^m = \frac{t_{cv}^m}{t} \tag{19}$$

$$cs_n = \frac{(\sum_{i=1}^m cd_n^i)^2}{m \times \sum_{i=1}^m (cd_n^i)^2} \tag{20}$$

Finally, we consider the RE to ensure energy-efficient UAV navigation due to the UAVs' limited energy capacity. We calculate the RE at each time-step deducting the energy consumption $e_n^t = \beta d_n^t$ or $e_n^t = \beta$ from the energy level at previous energy (21). At time-step $t = 0$, starting energy E_n^0 is the remaining energy after UAV-BS deployment.

$$E_n^i = E_n^{i-1} - e_n^i, \quad i = 1, 2, \ldots, t \tag{21}$$

In this study, we derive an optimal multi-UAV navigation scheme using FDRL to serve URLLC, mMTC, and eMBB devices with LOS, better link quality maintaining the CNR threshold, maximum CS, and minimum energy consumption in 5G heterogeneous networks. First, we derive the DRL algorithm for a UAV-BS navigation and then utilize it for all the UAV-BSs using the FL framework.

B. Proposed FDRL-based Multi-UAV Navigation

In this section, we propose a FDRL-based multi-UAV navigation scheme using deep Q-learning. First, we formulate the UAV-BS navigation problem based as a reinforcement task and introduce an autoencoder neural network to create the DQL framework. Finally, we derive the FL framework incorporating the DQL frameworks for multiple UAV-BSs and train and aggregate the DNNs for validation.

1. Deep Q-Learning Framework

In this section, we propose a DQL framework to optimize the UAV-BS navigation problem. In DQL, an agent interacts with the surrounding area and takes actions to gather experiences to reach the optimal solution utilizing a deep neural network (DNN). The UAV-BS acts as an agent and interacts with the environment consisting of URLLC, eMBB, and mMTC devices. Initially, the agent explores the environment and collects the channel and coverage information for all GDs. Following the steps described in Section III-A-2, the agent explores and exploits the environment for multiple episodes to achieve an optimal policy π_n for navigation. One episode terminates when the UAV goes out of the service area, faces any obstacles, or suffers from low energy. We define the state, action, and reward for use in the proposed DNN as follows:

- **State**: For each UAV-BS at any hour τ the state S consists the current locations of the k GDs within the assigned cluster from the BS. Therefore, the state space can be represented as $S_n = \{x^{\tau}_{d1}, y^{\tau}_{d1}, x^{\tau}_{d2}, y^{\tau}_{d2}, \ldots, x^{\tau}_{dk}, y^{\tau}_{dk}\}$ for the n^{th} UAV-BS at hour τ.

- **Action**: The main action of the agent is to navigate in such way to cover as many GDs as possible for the maximum steps utilizing minimum energy. For navigation, UAV-BS selects distance $d^t \in [0, d_{max}]$, azimuth $\phi^t \in [0, 2\pi]$, inclination $\theta^t \in [0, \pi]$ and moves to that position using PID. Thus, action set for the n^{th} UAV-BS at time step t can be represented as $a^t_n = \{d^t_n, \phi^t_n, \theta^t_n\}$. Here, our main objective is to obtain optimal UAV-BS trajectory. Thus, we obtain the A^{τ}_n actions for T time steps for hour τ.

- **Reward**: Whenever a UAV-BS takes action at time step t, it gets a reward r^t. We model the reward function based on the coverage score, residual energy, and maximum CNR value. We can represent the reward r^t_n for the n^{th} UAV-BS at time step t as (22). We also consider a zero reward in case of the UAV goes out of the service area, faces any obstacles, provides poor CNR, or suffers from low energy. Here, (22) ensures the maximum coverage,

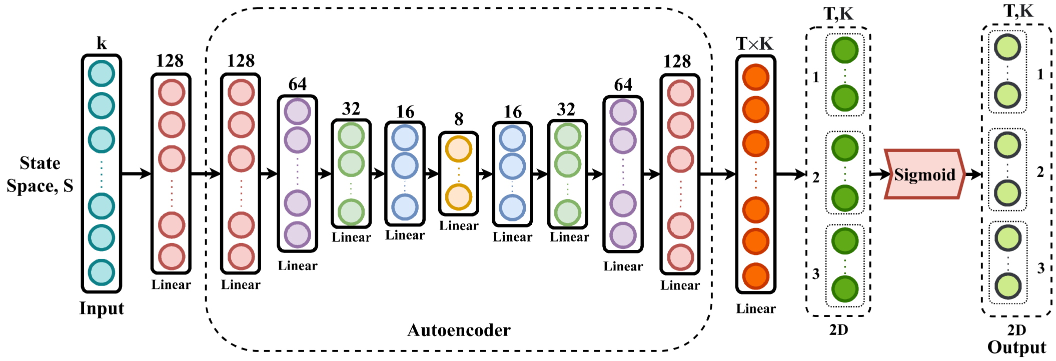


Figure 25: Proposed DNN structure for UAV-BS navigation.

residual energy, and minimum CNR requirement.

$$r_n^t = \begin{cases} 0 & \text{if } \mathit{Out\text{-}of\text{-}range},\ \mathit{obstacle},\ \mathit{low\ energy},\ \Gamma_n^t < \Gamma_{th} \\ cs_n^t \times E_n^t & \text{if } \Gamma_n^t \geq \Gamma_{th} \end{cases}, \quad (22)$$

With the state, action, and reward, we propose the deep neural network (DNN) structure shown in Fig. 25 as the policy controller for UAV-BS navigation. The main goal of the DNN is to derive probabilities (Q-values) for selecting the distance, azimuth, inclination for UAV-BS navigation. These probabilities are the Q-values for RL.

In this study, we use an autoencoder having an input and output size of 128 and code size 8. Finally, the autoencoder output is passed through a linear layer to reshape it and a Sigmoid layer to obtain the preferred distance, azimuth, and inclination probabilities. Here, K is the number of samples taken within the range of distance, azimuth, inclination for T time steps for an hour τ. The state space S consisting of k GD locations is given as the input of our policy network. Initially, the input is first embedded with dimension 128. It then passes through the policy network to generate the channel assigning probabilities, as shown in Fig. 25.

2. Federated Learning Framework

In this section, we utilize the proposed DQL framework for a UAV-BS navigation to derive the FL framework. We consider $\{U_1, U_2, \ldots, U_n\}$ UAV-BSs train their DNNs using the proposed DQL framework. After that, the BS collects the parameters of the DNNs, aggregates them, updates the global DNN (GDNN), and then updates the DNNs of the UAV-BSs with the GDNN at an interval of T_{update}. Here, we consider the HFDRL framework, where each agent interacts with different environments and performs the same task. Each UAV-BS train their DNNs using the local data of a specific area assigned by the BS as shown in Fig. 6 and Fig. 24. The main benefit of the FDRL is that any UAV-BSs can be replaced at any time without interrupting the network service in case of emergency such as low energy, equipment failure. The BS can quickly transfer the GDNN to the new UAV-BS, which replaced an old UAV-BS. Thus, we consider that if the energy level of a UAV-BS goes below e_{th}, another new UAV-BS will take over and update itself with the GDNN.

3. Training

We train the DNNs of the UAV-BSs gradually with their local training data $I_n^{data} = \{S_n^1, S_n^2, \ldots, S_n^I\}$ per episode. For T time steps, each UAV-BS selects T actions using ε-greedy policy according to the output probabilities from the DNN. An episode terminates when the UAV-BS goes out of the service area, faces any obstacles, suffers from low energy, or the number of data I runs out.

The policy to take actions for each UAV-BS per state space can be expressed as

$$A_l^\tau = \begin{cases} \text{argmax } Q(S_l^\tau, A_l^\tau) & \text{if } \varepsilon < \ \varepsilon_{th}; \text{ where } \varepsilon_{th} \in (0,1] \\ \text{random action } [T,K] & \text{otherwise} \end{cases}, \qquad (23)$$

$$\forall \tau = 1,2,\ldots,I,$$

$$\forall l = 1,2,\ldots,n.$$

After taking the actions using (23), the agent gets the rewards according to (22) and the next state space $S^{\tau+1}$.

To train the DNN at each UAV-BS, we calculate the loss and optimize the parameters of the DNN performing back-propagation. To calculate the loss, we approximate the optimal Q^*-values for each action of $S^{\tau+1}$ from a different DNN called the target DNN [66]. The target DNN is identical to the policy DNN and initialized by the parameters of the policy DNN at each UAV-BS. The next state space $S^{\tau+1}$ is given as an input to the target DNN, and from the outputs, the optimal Q^*-values are chosen greedily by each agent. We utilize the categorical cross-entropy loss function to compute the loss between the optimum Q^*-values and normal Q-values since selecting the distance, azimuth, and inclination is a classification problem [67]. We use the Adam optimizer [68] to optimize the policy DNN after computing the loss. After certain episodes, we update the target DNNs with the parameters of the policy DNNs to correctly estimate the optimal $Q*$-values for every UAV-BS.

For a more stable convergence of the optimal policy, we introduce the experience replay memory (ERM) to the DQL for each agent [69]. Initially, the agents explore the environment and save current states, actions, combined rewards, and next states $(S^\tau, A^\tau, R^\tau, S^{\tau+1})$ as a tuple in their individual ERM. Subsequently, the agents take a mini-batch of tuples from their ERMs and train their policy DNN. The ERMs continue to be updated for each training data.

To train the GDNN, we train the policy DNNs of the UAV-BSs in parallel utilizing a multiprocessing tool. They simultaneously train their policy DNNs and send the parameters of their policy DNNs to the BS periodically. Then, the BS updates the GDNN and sends the updated GDNN parameters to the UAV-BSs. Once the UAV-BSs receive the parameters, they update their policy DNNs. We consider at least 2 time steps to complete this back-and-forth parameter updating.

Algorithm 4 summarize the proposed FDRL framework and the working flow where we consider the UAV-BSs are serving the GDs $I = 24$ hours a day and taking $T = 60$ actions per hour. We also consider autonomous visit to nearest charging station in case of low energy.

C. Simulation Analysis

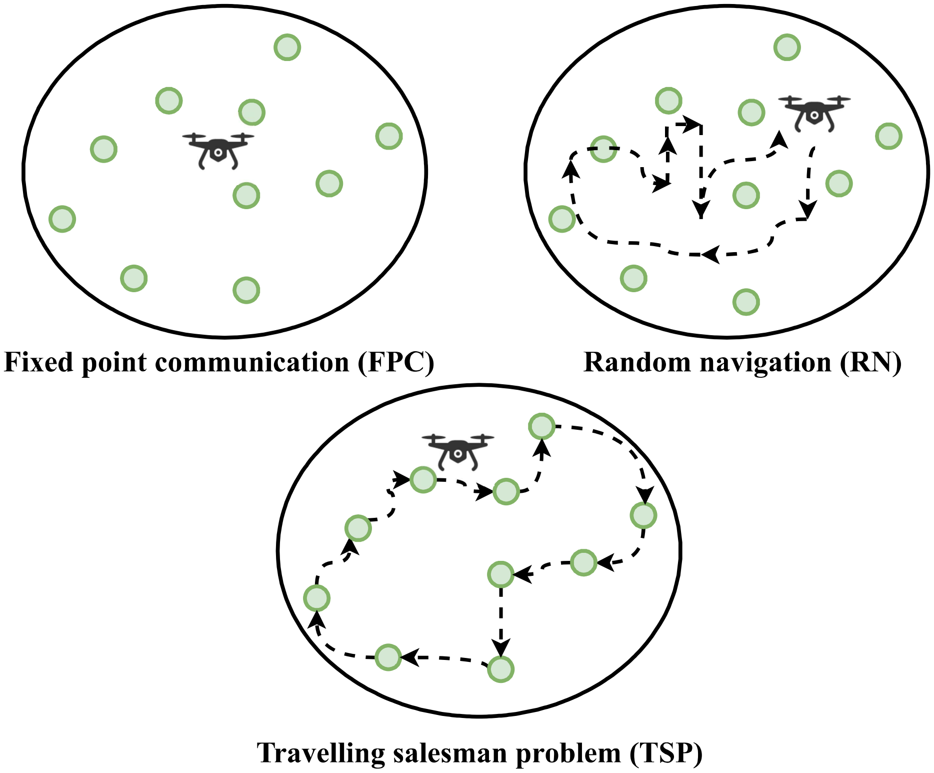
In this section, we perform multiple simulations to analyze the performance of the proposed FDRL algorithm for multiple UAV-BS navigation (FDRL-nav) and compare it with the conventional baselines such as fixed point communication (FPC), random navigation (RN), travelling salesman problem (TSP). In FPC. the UAV-BSs hover at a single point and serve the GDs. In contrast, UAV-BSs randomly move over the GDs and serve them in RN model. In TSP, UAV-BSs gather the locations of the GDs and fly to that locations at each time step to serve the GDs as shown in Fig. 26.

1. Simulation Environment

We consider a 5G macro-cell where 60 GDs are randomly distributed far from the BS for the simulation environment. The GDs suffer from packet loss, poor

Algorithm 4 Proposed FDRL Algorithm

```
Initialize global DNN with random parameters $g$ in BS.
for each UAV-BS do
    Initialize $n$ policy and target DNNs with random parameters ($p_i$ and $p'_i$).
    Initialize $n$ experience replay memory (ERM).
end for
for each episode in a UAV-BS do
    for each hour $\tau$ do
        Get the cluster location for deployment.
        Get the $k$ GD locations within the cluster.
        Select the distance, azimuth, inclination for action space $A^{\tau}$
            for present state space $S^{\tau}$ based on $\varepsilon$.
        for each time step $t$ do
            Check for UAV-BS flies beyond cluster or faces obstacles or
            suffers from low energy
            Observe the immediate rewards $r^t$ using (22).
            Obtain the combined rewards $R^{\tau}$ and next state space $S^{\tau+1}$.
            Check for episode termination conditions.
        end for
        Insert $(S^{\tau}, A^{\tau}, R^{\tau}, S^{\tau+1})$ in ERM.
        Create a mini-batch with random sample from ERM.
        for each tuple in mini-batch do
            Obtain $Q$-values and $Q^*$-values using policy and target DNN.
            Calculate the loss using $Q$ an $Q^*$-values.
            Optimize the policy DNN using Adam optimizer.
        end for
    end for
    $p'_i \leftarrow p_i$ after certain number of episodes.
    $g \leftarrow mean(p_1, p_2, \ldots, p_n)$ after certain number of episodes.
    $(p_1, p_2, \ldots, p_n) \leftarrow g$ immediately after updating $g$.
    Check for energy level for autonomous visit to nearby charging points.
end for
```



Fixed point communication (FPC)

Random navigation (RN)

Travelling salesman problem (TSP)

Figure 26: Baseline methods for UAV-BS navigation.

link quality, and low QoS fulfilling rate due to the poor LOS, multi-path fading. We only consider three types of GDs, URLLC, eMBB, and mMTC devices. We model the channel gain h_m of the m^{th} GD based on the Rayleigh fading model and calculate the CNR Γ_m using (18) where $\sigma_k = \frac{BW \times N_0}{k}$ for $\forall k = 1,2,\ldots,K$ with BW = 1MHz and N_0 = -174 dBm/Hz. To analyze the performance, we use the simulation parameters given in Table 2.

The BS utilizes a conventional K-means algorithm to divide the GDs into clusters for UAV-BS deployment and avoid mutual interference. We divide the 60 GDs into 5 clusters and determine the number of clusters using the elbow method of K-means for the simulation [71]. We also consider a UAV-BS agent per cluster

Table 2: Simulation parameters for FDRL-based UAV-BS navigation

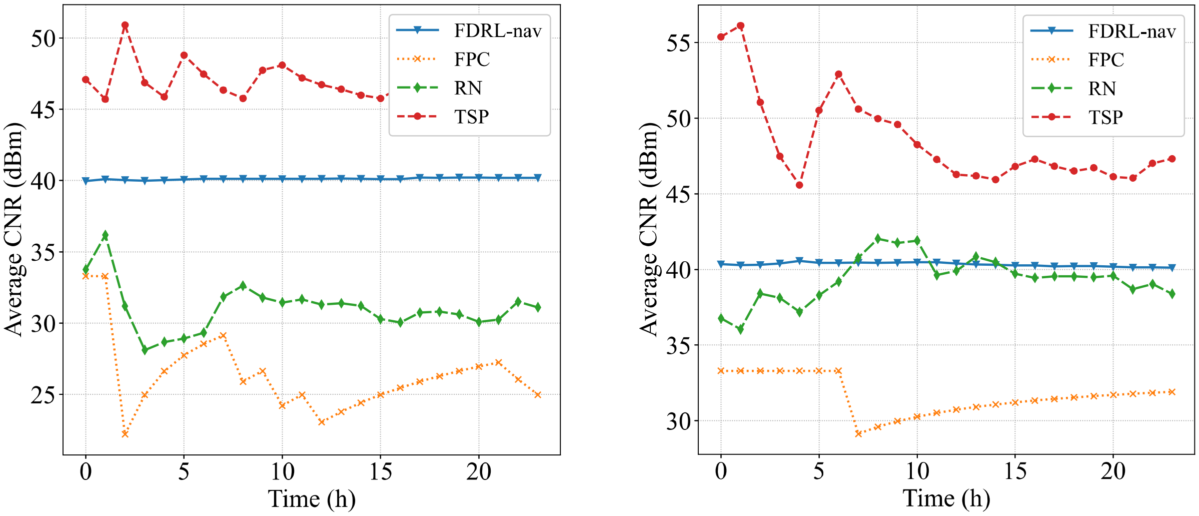
Parameter	Value	Parameter	Value
BW	1 MHz	Learning rate	1×10^{-4}
n	5	Batch size	8
m	60	e_{th}	15%
T	60	Number of episodes	200
I	24	Γ_{th}	25 dBm
Discount factor	0.999	K	15

serving the GDs. The parameters of DNNs of each UAV-BS and GDNN, such as weights and biases, are initialized randomly and uniformly. The input size of the DNNs is k, and the embedded size is 128.

For training data, we consider the locations of the GDs. We use $2D$ random walk mobility model [72] to generate the mobility data of the GDs for 24 hours. We generate 1000 instances for training and 500 instances for validation data-set randomly for each episode. Each instance consists of 60×24 location information. We also generate the $3D$ locations of obstacles randomly that are unknown to the UAV-BSs.

2. Performance Analysis

This section compares the proposed FDRL-nav with FPC, RN, and TSP in terms of system average CNR, average coverage time, coverage score, and residual energy for 24 hours in multiple agent scenarios without UAV-BS replacement.



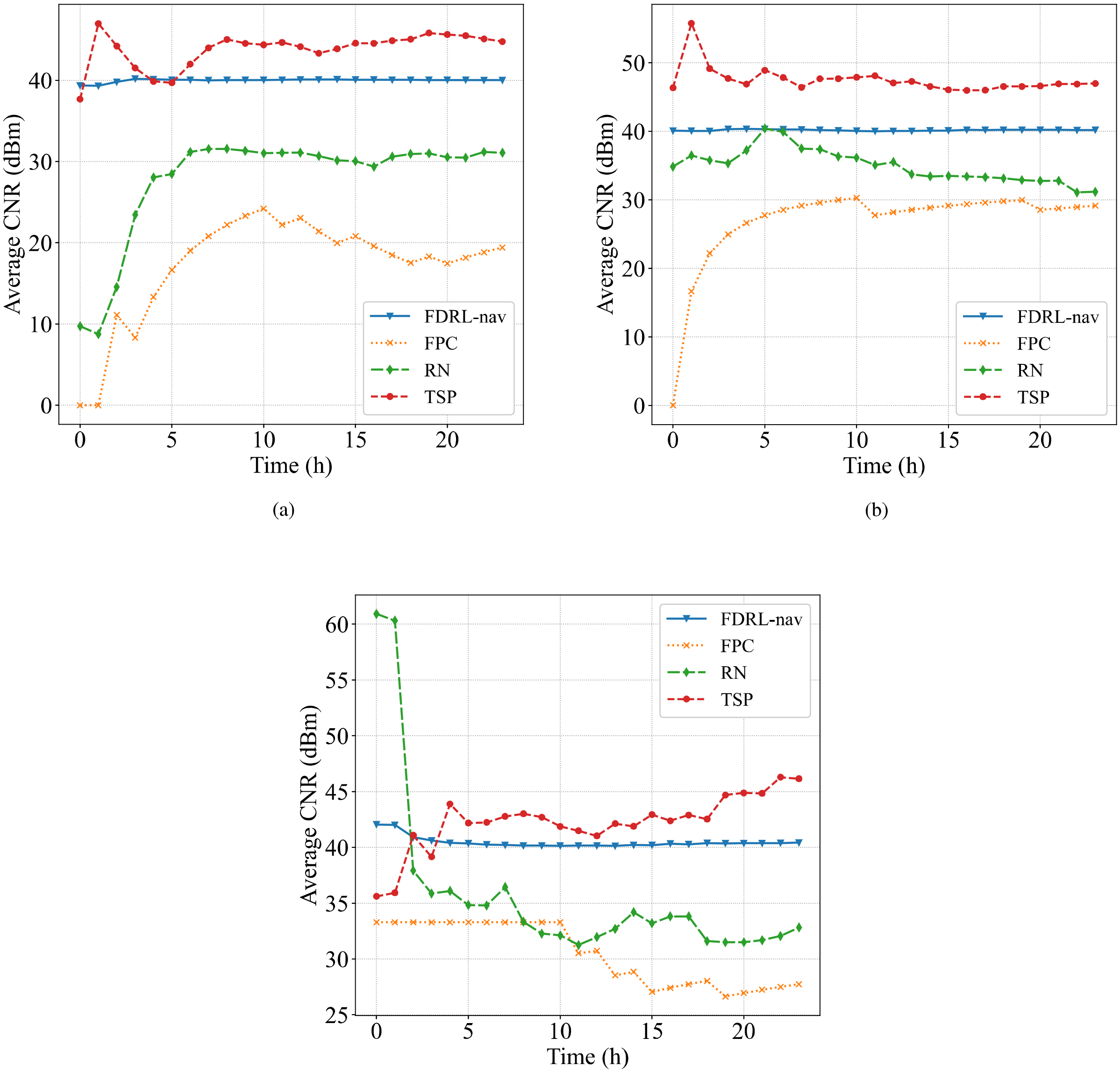
(a)

(b)

Figure 27: CNR comparison for (a) UAV-BS 1, (b) UAV-BS 2.

Fig. 27 and 28 shows the average CNR established by five UAV-BS agents throughout a day. It is evident from Fig. 27 and 28 that the TSP and the FDRL-nav scheme achieve the highest CNRs above the threshold. It is usual for the TSP method to achieve high CNR as the agents travel to each GD to serve them. By contrast, the CNR fluctuates and stays below the threshold for RN and FPC schemes, respectively. In the RN method, the UAV-BSs randomly move over the GDs, resulting in undesirable fluctuation in the CNR. The UAV-BSs hover at a fixed point to serve the GDs and can not maintain the CNR threshold in the FPC method. Fig. 27 and 28 also reinforces the optimality of the proposed FDRL-nav scheme as all the UAV-BSs maintain a steady CNR above the threshold.

Average coverage time comparison among FDRL-nav with FPC, RN, and TSP for each UAV-BS agent is shown in Fig. 29 and 30 . It is evident from



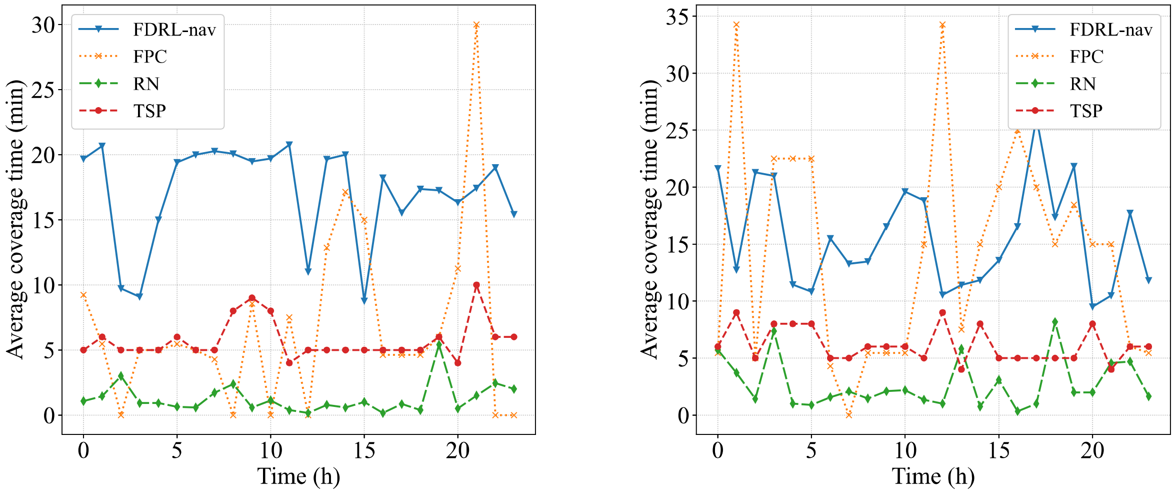
(a)

(b)

(c)

Figure 28: CNR comparison for (c) UAV-BS 3, (d) UAV-BS 4, and (e) UAV-BS 5.

the figure that the proposed FDRL-nav achieves the highest average coverage



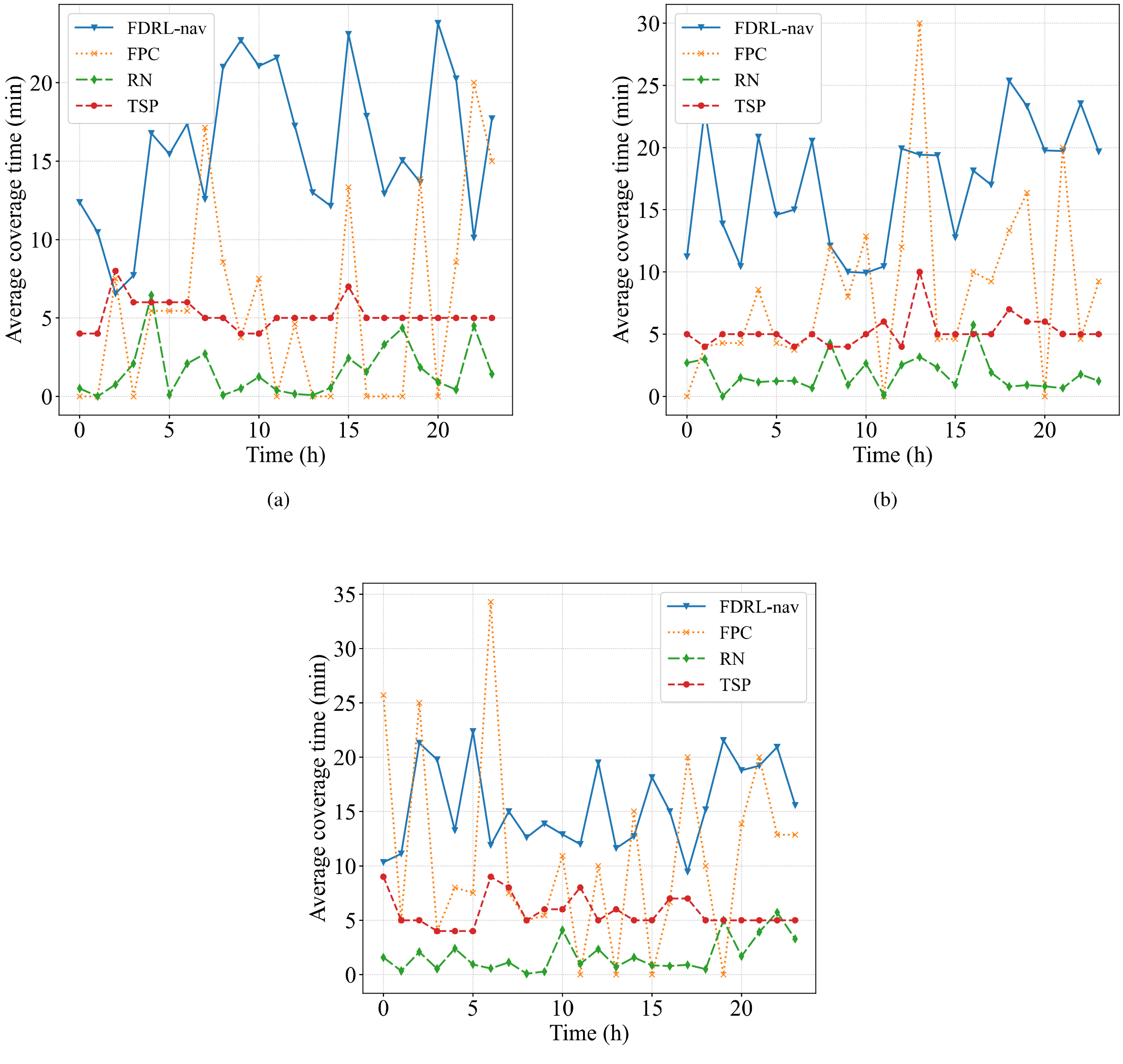
(a) (b)

Figure 29: Average coverage time comparison for (a) UAV-BS 1, (b) UAV-BS 2.

time for each agent. However, the average coverage time fluctuates due to GD mobility, unwanted obstacles, and LOS problems. The GDs were blocked by the obstacles in that hours, causing some uncertainty. In contrast, other schemes show low average coverage scores, which is not desirable.

Fig. 31 and 32 shows the CS achieved by the four schemes in each agent. In every UAV-BS agent, the proposed FDRL achieves the highest CS compared to the other methods. Moreover, we can also observe that the TSP scheme achieves similar CS owing to traveling and serving nature. However, it can ensure energy efficiency and higher average coverage time as shown in Fig. 33, 29, and 30, respectively.

In Fig. 33, we present the average residual energy of the overall system at each hour for FDRL-nav, FPC, RN, and TSP scheme. As in the FPC scheme, the UAV-BSs hover over a certain point that requires a meager percentage of energy



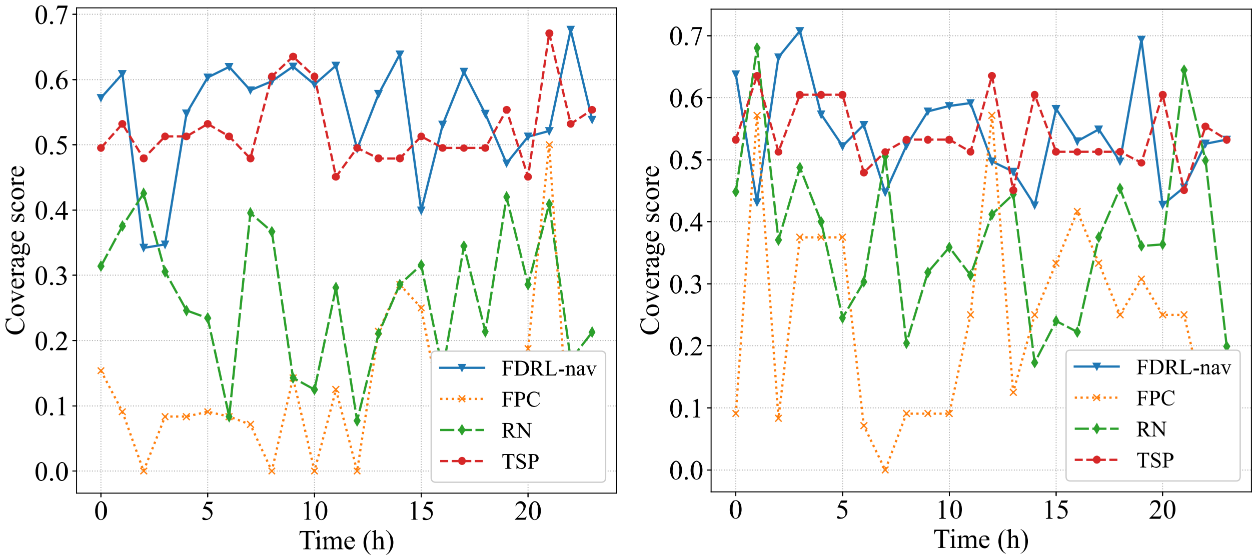
(a)

(b)

(c)

Figure 30: Average coverage time comparison for (c) UAV-BS 3, (d) UAV-BS 4, and (e) UAV-BS 5.

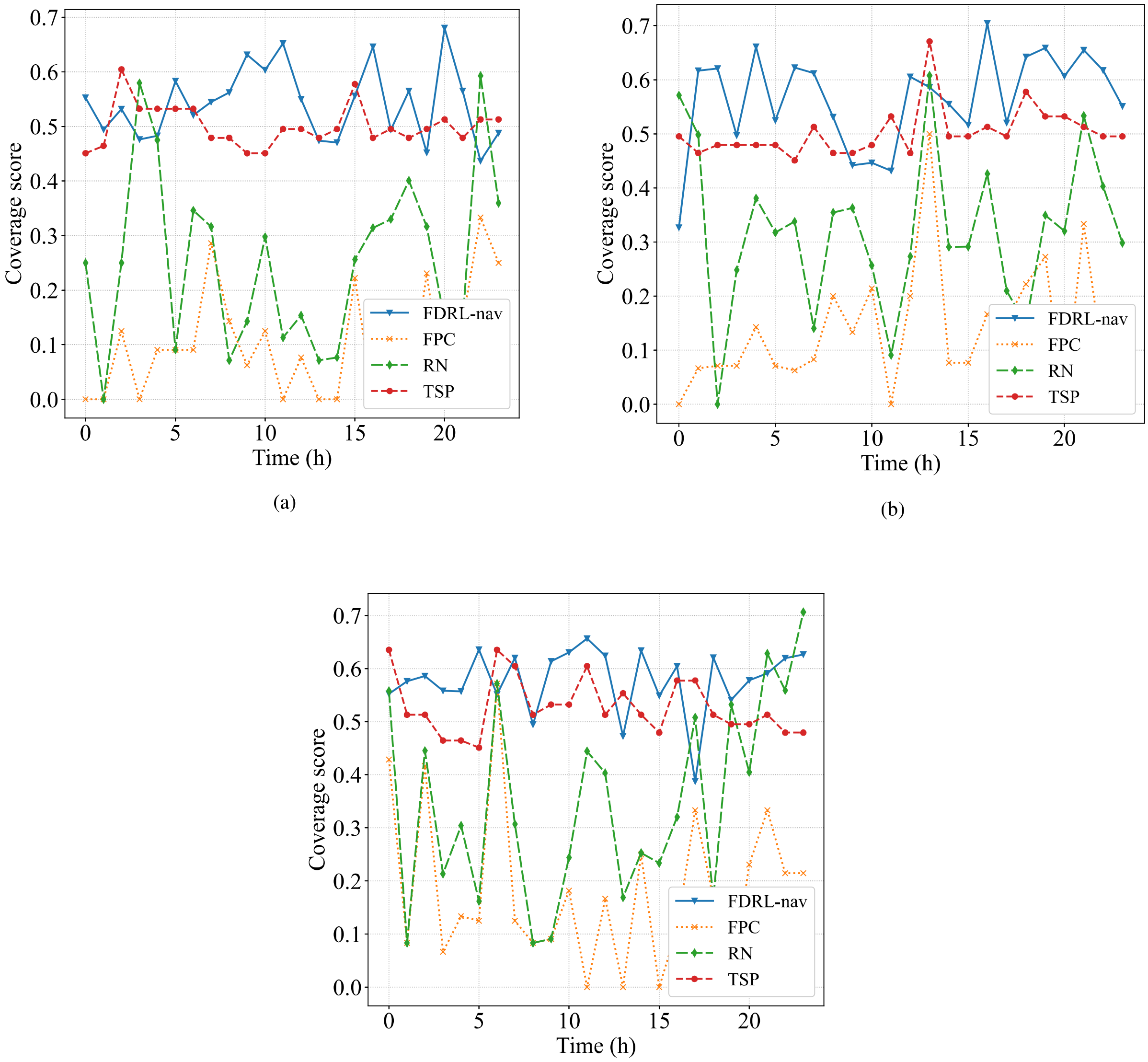
resulting in higher average residual energy. By contrast, the average residual



(a)

(b)

Figure 31: Coverage score comparison for (a) UAV-BS 1, (b) UAV-BS 2.



(a)

(b)

(c)

Figure 32: Coverage score comparison for (c) UAV-BS 3, (d) UAV-BS 4, and (e) UAV-BS 5.

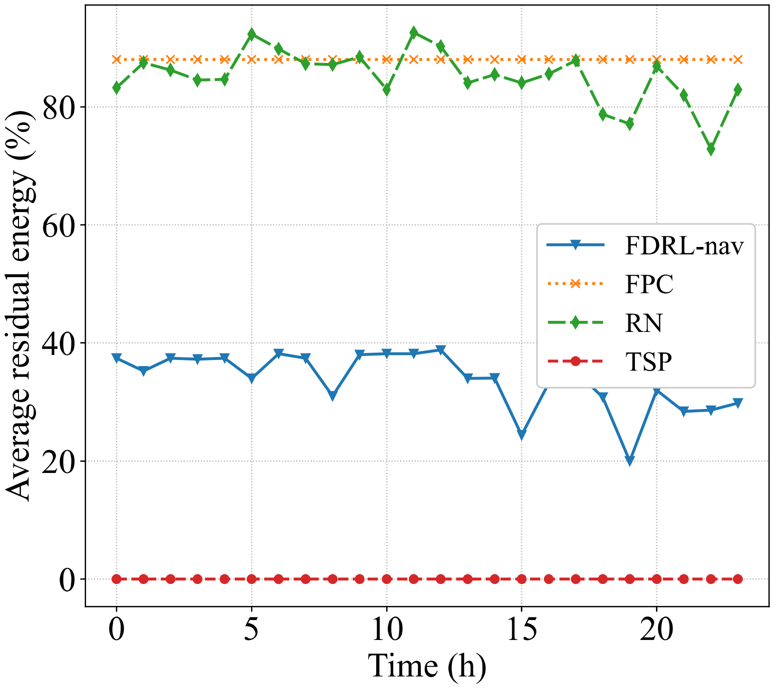


Figure 33: Residual energy comparison.

energy is zero, and the agents get replaced at every hour in the TSP scheme owning to its diverse movement. The RN scheme has high average residual energy owing to the low average coverage time. Although the proposed FDRL-nav uses a significant amount of energy, it outperforms all other schemes and leaves with average residual energy above the energy threshold e_{th}.

Furthermore, we also compute the time complexity of the proposed FDRL-nav. According to [70], the time complexity of the proposed DNN network for each UAV-BS can be represented as $O(5Id_e^2(k))$, where size of the two linear layers $d_e = 128$ and input size is k. Finally, we can conclude that the proposed FDRL-nav shows prominent performance with low complexity.

CONCLUSION

In this thesis, we propose a priority-based resource allocation scheme with deep Q-learning to fulfill the QoS requirements of the 5G services, such as URLLC, eMBB, and mMTC services, while maximizing the system performance and fairness of the multi-carrier NOMA system.

- It maximizes sum-rate (MSR), channel sum-rate (MCSR) while ensuring the 5G QoS requirements and channel distribution fairness.
- It addresses different constraints of the NOMA system, including the total power budget of the base station (BS), the minimum data rate requirement of each device, the QoS policies of different services of the 5G network, and the sum-rate maximization with channel fairness constraints
- The proposed scheme priority-JRA outperforms the JRA and DPA-FC schemes under different conditions.
- The proposed priority-JRA method is less complex than other optimal exhaustive search-based solutions while achieving a near-optimal solution.

We also propose a novel FDRL-based multiple UAV-BS navigation scheme to serve URLLC, mMTC, and eMBB devices suffering from NLOS, poor link quality, and multi-path fading in 5G heterogeneous networks.

- It ensures the LOS, better link quality for the suffered 5G ground devices.
- It addresses different constraints, including the power budget of the UAV-BS, obstacles, autonomous visit to charging points, the maximum device coverage, the minimum CNR threshold of the 5G network, and overall device coverage fairness.

- The proposed scheme FDRL-nav outperforms the baseline schemes, such as FPC, RN, TSP, achieving a near-optimal solution.

PUBLICATIONS

D. Journals

1. S. Rezwan **and** W. Choi, “Priority-based joint resource allocation with deep q-learning for heterogeneous noma systems,” *IEEE Access*, **jourvol** 9, **pages** 41 468–41 481, 2021. DOI: 10.1109/ACCESS.2021.3065314.

2. S. Rezwan **and** W. Choi, “A survey on applications of reinforcement learning in flying ad-hoc networks,” *Electronics*, **jourvol** 10, **number** 4, **page** 449, 2021.

E. Conferences

1. S. Rezwan, S. Shin **and** W. Choi, “Efficient user clustering and reinforcement learning based power allocation for NOMA systems,” **in** *Proceedings of the International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, Korea (South), 2020, **pages** 143–147. DOI: 10.1109/ICTC49870.2020.9289376.

2. S. Rezwan **and** W. Choi, “Q-learning-based resource allocation with priority-based clustering for heterogeneous noma systems,” **in** *Proceedings of the International Conference on Smart Media and Applications (SMA)*, Jeju, Korea (South), 2020.

3. S. Rezwan, S. Shin **and** W. Choi, “A study on implementation methods of reinforcement learning in noma systems,” *한국통신학회 학술대회논문집*, **pages** 189–190, 2020.

REFERENCES

[1] P. Popovski, K. F. Trillingsgaard, O. Simeone **and** G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access*, **jourvol** 6, **pages** 55 765–55 779, 2018.

[2] S. Hu, B. Yu, C. Qian, Y. Xiao, Q. Xiong, C. Sun **and** Y. Gao, "Nonorthogonal interleave-grid multiple access scheme for industrial internet of things in 5G network," *IEEE Transactions on Industrial Informatics*, **jourvol** 14, **number** 12, **pages** 5436–5446, 2018. DOI: 10.1109/TII.2018.2858142.

[3] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour **and** G. Wunder, "5G: A tutorial overview of standards, trials, challenges, deployment, and practice," *IEEE Journal on Selected Areas in Communications*, **jourvol** 35, **number** 6, **pages** 1201–1221, 2017.

[4] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong **and** J. C. Zhang, "What will 5G be?" *IEEE Journal on Selected Areas in Communications*, **jourvol** 32, **number** 6, **pages** 1065–1082, 2014. DOI: 10.1109/JSAC.2014.2328098.

[5] A. Shahini **and** N. Ansari, "NOMA aided narrowband IoT for machine type communications with user clustering," *IEEE Internet of Things Journal*, **jourvol** 6, **number** 4, **pages** 7183–7191, 2019.

[6] X. Liu, X. B. Zhai, W. Lu **and** C. Wu, "QoS-guarantee resource allocation for multibeam satellite industrial internet of things with NOMA,"

IEEE Transactions on Industrial Informatics, **jourvol** 17, **number** 3, **pages** 2052–2061, 2021. DOI: 10.1109/TII.2019.2951728.

[7] E. J. dos Santos, R. D. Souza, J. L. Rebelatto **and** H. Alves, "Network slicing for URLLC and eMBB with max-matching diversity channel allocation," *IEEE Communications Letters*, **jourvol** 24, **number** 3, **pages** 658–661, 2020.

[8] G. Gui, H. Sari **and** E. Biglieri, "A new definition of fairness for non-orthogonal multiple access," *IEEE Communications Letters*, **jourvol** 23, **number** 7, **pages** 1267–1271, 2019.

[9] X. Yan, K. An, T. Liang, G. Zheng, Z. Ding, S. Chatzinotas **and** Y. Liu, "The application of power-domain non-orthogonal multiple access in satellite communication networks," *IEEE Access*, **jourvol** 7, **pages** 63 531–63 539, 2019. DOI: 10.1109/ACCESS.2019.2917060.

[10] L. Dai, B. Wang, Y. Yuan, S. Han, C. I **and** Z. Wang, "Non-orthogonal multiple access for 5G: Solutions, challenges, opportunities, and future research trends," *IEEE Communications Magazine*, **jourvol** 53, **number** 9, **pages** 74–81, 2015.

[11] M. Liu, T. Song **and** G. Gui, "Deep cognitive perspective: Resource allocation for NOMA-based heterogeneous IoT with imperfect sic," *IEEE Internet of Things Journal*, **jourvol** 6, **number** 2, **pages** 2885–2894, 2019.

[12] Z. Ding, X. Lei, G. K. Karagiannidis, R. Schober, J. Yuan **and** V. K. Bhargava, "A survey on non-orthogonal multiple access for 5G networks: Research challenges and future trends," *IEEE Journal on Selected Areas in Communications*, **jourvol** 35, **number** 10, **pages** 2181–2195, 2017.

[13] M. Zeng, A. Yadav, O. A. Dobre, G. I. Tsiropoulos **and** H. V. Poor, "Capacity comparison between MIMO-NOMA and MIMO-OMA with multiple users in a cluster," *IEEE Journal on Selected Areas in Communications*, **jourvol** 35, **number** 10, **pages** 2413–2424, 2017. DOI: 10.1109/JSAC.2017.2725879.

[14] A. Celik, M. Tsai, R. M. Radaydeh, F. S. Al-Qahtani **and** M. Alouini, "Distributed user clustering and resource allocation for imperfect NOMA in heterogeneous networks," *IEEE Transactions on Communications*, **jourvol** 67, **number** 10, **pages** 7211–7227, 2019.

[16] Z. Ding, Y. Liu, J. Choi, Q. Sun, M. Elkashlan, I. Chih-Lin **and** H. V. Poor, "Application of non-orthogonal multiple access in LTE and 5G networks," *IEEE Communications Magazine*, **jourvol** 55, **number** 2, **pages** 185–191, 2017.

[17] Y. Saito, Y. Kishiyama, A. Benjebbour, T. Nakamura, A. Li **and** K. Higuchi, "Non-orthogonal multiple access (NOMA) for cellular future radio access," **in** *Proceedings of the 77th IEEE Vehicular Technology Conference (VTC Spring)*, Dresden, Germany, 2013, **pages** 1–5. DOI: 10.1109/VTCSpring.2013.6692652.

[18] Y. Liu **and** Y. Dai, "On the complexity of joint subcarrier and power allocation for multi-user OFDMA systems," *IEEE Transactions on Signal Processing*, **jourvol** 62, **number** 3, **pages** 583–596, 2014. DOI: 10.1109/TSP.2013.2293130.

[19] S. Zhang, B. Di, L. Song **and** Y. Li, "Radio resource allocation for non-orthogonal multiple access (NOMA) relay network using matching game," **in** *Proceedings of the IEEE International Conference on Communications*

(ICC), Kuala Lumpur, Malaysia, 2016, **pages** 1–6. DOI: 10.1109/ICC.2016.7510918.

[20] L. Lei, D. Yuan, C. K. Ho **and** S. Sun, "Joint optimization of power and channel allocation with non-orthogonal multiple access for 5G cellular systems," **in** *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, San Diego, CA, USA, 2015, **pages** 1–6. DOI: 10.1109/GLOCOM.2015.7417761.

[21] Y. Sun, D. W. K. Ng, Z. Ding **and** R. Schober, "Optimal joint power and subcarrier allocation for full-duplex multicarrier non-orthogonal multiple access systems," *IEEE Transactions on Communications*, **jourvol** 65, **number** 3, **pages** 1077–1091, 2017. DOI: 10.1109/TCOMM.2017.2650992.

[22] S. M. R. Islam, N. Avazov, O. A. Dobre **and** K. Kwak, "Power-domain non-orthogonal multiple access (NOMA) in 5G systems: Potentials and challenges," *IEEE Communications Surveys Tutorials*, **jourvol** 19, **number** 2, **pages** 721–742, 2017. DOI: 10.1109/COMST.2016.2621116.

[23] S. M. R. Islam, M. Zeng, O. A. Dobre **and** K. Kwak, "Resource allocation for downlink NOMA systems: Key techniques and open issues," *IEEE Wireless Communications*, **jourvol** 25, **number** 2, **pages** 40–47, 2018. DOI: 10.1109/MWC.2018.1700099.

[24] J. G. Andrews **and** T. H. Meng, "Optimum power control for successive interference cancellation with imperfect channel estimation," *IEEE Transactions on Wireless Communications*, **jourvol** 2, **number** 2, **pages** 375–383, 2003.

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

[26] J. Choi, “Power allocation for max-sum rate and max-min rate proportional fairness in NOMA,” *IEEE Communications Letters*, **jourvol** 20, **number** 10, **pages** 2055–2058, 2016. DOI: 10.1109/LCOMM.2016.2596760.

[27] X. Shao, C. Yang, D. Chen, N. Zhao **and** F. R. Yu, “Dynamic IoT device clustering and energy management with hybrid NOMA systems,” *IEEE Transactions on Industrial Informatics*, **jourvol** 14, **number** 10, **pages** 4622–4630, 2018. DOI: 10.1109/TII.2018.2856776.

[28] P. Parida **and** S. S. Das, “Power allocation in ofdm based NOMA systems: A dc programming approach,” **in** *Proceedings of the IEEE Globecom Workshops (GC Wkshps)*, 2014, **pages** 1026–1031. DOI: 10.1109/GLOCOMW.2014.7063568.

[29] M. Hojeij, J. Farah, C. A. Nour **and** C. Douillard, “Resource allocation in downlink non-orthogonal multiple access (NOMA) for future radio access,” **in** *Proceedings of the IEEE 81st Vehicular Technology Conference (VTC Spring)*, 2015, **pages** 1–6. DOI: 10.1109/VTCSpring.2015.7146056.

[30] J. Zhu, J. Wang, Y. Huang, S. He, X. You **and** L. Yang, “On optimal power allocation for downlink non-orthogonal multiple access systems,” *IEEE Journal on Selected Areas in Communications*, **jourvol** 35, **number** 12, **pages** 2744–2757, 2017. DOI: 10.1109/JSAC.2017.2725618.

[31] Z. Ning, X. Wang, J. J. P. C. Rodrigues **and** F. Xia, “Joint computation offloading, power allocation, and channel assignment for 5G-enabled traffic management systems,” *IEEE Transactions on Industrial Informatics*, **jourvol** 15, **number** 5, **pages** 3058–3067, 2019. DOI: 10.1109/TII.2019.2892767.

[32] L. Xiao, Y. Li, C. Dai, H. Dai **and** H. V. Poor, “Reinforcement learning-based NOMA power allocation in the presence of smart jamming,” *IEEE Transactions on Vehicular Technology*, **jourvol** 67, **number** 4, **pages** 3377–3389, 2018.

[33] C. He, Y. Hu, Y. Chen **and** B. Zeng, “Joint power allocation and channel assignment for NOMA with deep reinforcement learning,” *IEEE Journal on Selected Areas in Communications*, **jourvol** 37, **number** 10, **pages** 2200–2210, 2019. DOI: 10.1109/JSAC.2019.2933762.

[34] Y. Wei, F. R. Yu, M. Song **and** Z. Han, “User scheduling and resource allocation in hetnets with hybrid energy supply: An actor-critic reinforcement learning approach,” *IEEE Transactions on Wireless Communications*, **jourvol** 17, **number** 1, **pages** 680–692, 2018. DOI: 10.1109/TWC.2017.2769644.

[35] W. Shi, J. Li, W. Xu, H. Zhou, N. Zhang **and** X. Shen, “3d drone-cell deployment optimization for drone assisted radio access networks,” **in** *2017 IEEE/CIC International Conference on Communications in China (ICCC)*, IEEE, 2017, **pages** 1–6.

[36] N. Nomikos, E. T. Michailidis, P. Trakadas, D. Vouyioukas, H. Karl, J. Martrat, T. Zahariadis, K. Papadopoulos **and** S. Voliotis, “A uav-based

moving 5g ran for massive connectivity of mobile users and iot devices," *Vehicular Communications*, **jourvol** 25, **page** 100 250, 2020.

[37] I. Bor-Yaliniz **and** H. Yanikomeroglu, "The new frontier in ran heterogeneity: Multi-tier drone-cells," *IEEE Communications Magazine*, **jourvol** 54, **number** 11, **pages** 48–55, 2016.

[38] S. Sekander, H. Tabassum **and** E. Hossain, "Multi-tier drone architecture for 5g/b5g cellular networks: Challenges, trends, and prospects," *IEEE Communications Magazine*, **jourvol** 56, **number** 3, **pages** 96–103, 2018. DOI: `10.1109/MCOM.2018.1700666`.

[39] A. Al-Hourani, S. Kandeepan **and** S. Lardner, "Optimal lap altitude for maximum coverage," *IEEE Wireless Communications Letters*, **jourvol** 3, **number** 6, **pages** 569–572, 2014.

[40] V. Sharma, K. Srinivasan, H.-C. Chao, K.-L. Hua **and** W.-H. Cheng, "Intelligent deployment of uavs in 5g heterogeneous communication environment for improved coverage," *Journal of Network and Computer Applications*, **jourvol** 85, **pages** 94–105, 2017.

[41] A. Fotouhi, M. Ding **and** M. Hassan, "Dynamic base station repositioning to improve spectral efficiency of drone small cells," **in** *2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, IEEE, 2017, **pages** 1–9.

[42] ——, "Service on demand: Drone base stations cruising in the cellular network," **in** *2017 IEEE Globecom Workshops (GC Wkshps)*, IEEE, 2017, **pages** 1–6.

[43] J. Lyu, Y. Zeng, R. Zhang **and** T. J. Lim, "Placement optimization of uav-mounted mobile base stations," *IEEE Communications Letters*, **jourvol** 21, **number** 3, **pages** 604–607, 2016.

[44] U. Challita, W. Saad **and** C. Bettstetter, "Deep reinforcement learning for interference-aware path planning of cellular-connected uavs," **in** *2018 IEEE International Conference on Communications (ICC)*, IEEE, 2018, **pages** 1–7.

[45] S. F. Abedin, M. S. Munir, N. H. Tran, Z. Han **and** C. S. Hong, "Data freshness and energy-efficient uav navigation optimization: A deep reinforcement learning approach," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[46] C. H. Liu, X. Ma, X. Gao **and** J. Tang, "Distributed energy-efficient multi-uav navigation for long-term communication coverage by deep reinforcement learning," *IEEE Transactions on Mobile Computing*, **jourvol** 19, **number** 6, **pages** 1274–1285, 2019.

[47] M. Ponsen, M. E. Taylor **and** K. Tuyls, "Abstraction and generalization in reinforcement learning: A summary and framework," **in** *International Workshop on Adaptive and Learning Agents*, Springer, 2009, **pages** 1–32.

[48] K. Tuyls **and** A. Nowé, "Evolutionary game theory and multi-agent reinforcement learning," *The Knowledge Engineering Review*, **jourvol** 20, **number** 1, **pages** 63–90, 2005.

[49] Z. Xiong, Y. Zhang, D. Niyato, R. Deng, P. Wang **and** L.-C. Wang, "Deep reinforcement learning for mobile 5g and beyond: Fundamentals, applications, and challenges," *IEEE Vehicular Technology Magazine*, **jourvol** 14, **number** 2, **pages** 44–52, 2019.

[50] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare **and** J. Pineau, "An introduction to deep reinforcement learning," *arXiv preprint arXiv:1811.12560*, 2018.

[51] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu **and** F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, **jourvol** 234, **pages** 11–26, 2017.

[52] D. Bau, J.-Y. Zhu, H. Strobelt, A. Lapedriza, B. Zhou **and** A. Torralba, "Understanding the role of individual units in a deep neural network," *Proceedings of the National Academy of Sciences*, **jourvol** 117, **number** 48, **pages** 30 071–30 078, 2020.

[53] Y. Li, *Deep reinforcement learning: An overview*, 2018. arXiv: `1701.07274 [cs.LG]`.

[54] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver **and** K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," *CoRR*, **jourvol** abs/1602.01783, 2016. arXiv: `1602.01783`. [Online]. Available: `http://arxiv.org/abs/1602.01783`.

[55] K. Arulkumaran, M. P. Deisenroth, M. Brundage **and** A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, **jourvol** 34, **number** 6, **pages** 26–38, 2017. DOI: `10.1109/MSP.2017.2743240`.

[56] T. Lillicrap, J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver **and** D. Wierstra, "Continuous control with deep reinforcement learning," *CoRR*, Sep. 2015.

[57] Q. Yang, Y. Liu, T. Chen **and** Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, **jourvol** 10, **number** 2, **pages** 1–19, 2019.

[58] J. Konečnỳ, H. B. McMahan, D. Ramage **and** P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.

[59] H. B. McMahan, E. Moore, D. Ramage **and** B. A. y Arcas, "Federated learning of deep networks using model averaging," *arXiv preprint arXiv:1602.05629*, 2016.

[60] V. Smith, C.-K. Chiang, M. Sanjabi **and** A. Talwalkar, "Federated multi-task learning," *arXiv preprint arXiv:1705.10467*, 2017.

[61] Q. Yang, Y. Liu, Y. Cheng, Y. Kang, T. Chen **and** H. Yu, "Federated learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, **jourvol** 13, **number** 3, **pages** 1–207, 2019.

[62] Z. Wei, D. W. K. Ng, J. Yuan **and** H. Wang, "Optimal resource allocation for power-efficient MC-NOMA with imperfect channel state information," *IEEE Transactions on Communications*, **jourvol** 65, **number** 9, **pages** 3944–3961, 2017. DOI: 10.1109/TCOMM.2017.2709301.

[63] E. K. P. Chong **and** S. H. Zak, *An Introduction to Optimization*, 3 **edition**. Wiley, 2008.

[64] D. P. Kingma **and** M. Welling, "Auto-Encoding Variational Bayes," **in** *Proceedings of the 2nd International Conference on Learning Representations, ICLR*, Banff, AB, Canada, 2014, **pages** 1–14.

[65] X.-H. Le, H. V. Ho, G. Lee **and** S. Jung, “Application of long short-term memory (LSTM) neural network for flood forecasting,” *Water*, **jourvol** 11, **number** 7, 2019, ISSN: 2073-4441. DOI: `10.3390/w11071387`.

[66] R. S. Sutton, D. McAllester, S. Singh **and** Y. Mansour, “Policy gradient methods for reinforcement learning with function approximation,” **in** *Proceedings of the 12th International Conference on Neural Information Processing Systems*, Denver, CO: MIT Press, 1999, 1057–1063.

[67] Z. Zhang **and** M. R. Sabuncu, “Generalized cross entropy loss for training deep neural networks with noisy labels,” **in** *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montréal, Canada: Curran Associates Inc., 2018, 8792–8802.

[68] D. P. Kingma **and** J. Ba, “Adam: A method for stochastic optimization,” **in** *Proceedings of the 3rd International Conference on Learning Representations, ICLR*, San Diego, CA, USA, 2015. [Online]. Available: `http://arxiv.org/abs/1412.6980`.

[69] S. Zhang **and** R. S. Sutton, “A deeper look at experience replay,” *Computing Research Repository (CoRR)*, **jourvol** abs/1712.01275, 2017. arXiv: `1712.01275`. [Online]. Available: `http://arxiv.org/abs/1712.01275`.

[70] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser **and** I. Polosukhin, “Attention is all you need,” *Computing Research Repository (CoRR)*, **jourvol** abs/1706.03762, 2017. arXiv: `1706.03762`. [Online]. Available: `https://arxiv.org/pdf/1706.03762`.

[71] P. Bholowalia **and** A. Kumar, “Ebk-means: A clustering technique based on elbow method and k-means in wsn,” *International Journal of Computer Applications*, **jourvol** 105, **number** 9, 2014.

[72] G. Papikyan, E. Mokrov **and** K. Samouylov, “Interaction between user and uav with unreliable location information,” **in** *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*, Springer, 2020, **pages** 439–449.

ACKNOWLEDGEMENTS

I want to express my gratefulness to all the individuals who have supported me in the process of completing my Master's degree and research. Firstly, I really would like to take this opportunity to express my gratitude to Prof. Wooyeol Choi, my supervisor, for allowing me to pursue my Master's degree at Chosun University. His constant inspiration, support, and insightful recommendations have led and pushed me throughout my studies and research. His continuous supervision and direction have aided me in producing high-quality research. I will be eternally grateful to him for instilling in me the values of professionalism, organizational skills, and concentration. I also would like to convey my heartfelt gratitude to Prof. Seok Joo Shin and Prof. Moon Soo Kang, members of the thesis committee, for their constructive remarks and helpful ideas. Furthermore, I am glad for the opportunity to work in the Department of Computer Engineering at Chosun University with such a diversified batch of students, teachers, and staff. I want to thank Smart Networking Lab for giving me such an excellent opportunity and an environment to develop academically. My lab colleagues have been a source of moral and intellectual support for me. In addition, I want to express my appreciation to all of my Bangladeshi seniors and friends at Chosun University for their compassion and cooperation in making my life in South Korea easy and joyful. Lastly, I want to express my gratitude to my parents, relatives, and friends for their constant and unwavering support throughout my difficult times. It would have been difficult for me to do anything without their motivation and direction.